



TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND
Virumaa kolledž

Siemens tööstuskontrolleri avatud koodiga platvormiga ühenduse loomine

**Connecting to an open-source platform for a Siemens industrial
controller**

TELEMAATIKA JA ARUKAD SÜSTEEMID ÕPPEKAVA LÕPUTÖÖ

Üliõpilane: Darja Slobodjanova

Üliõpilaskood: 183582EDTR

Juhendaja: Sergei Ponomar, lektor

AUTORIDEKLARATSIOON

Olen koostanud lõputöö iseseisvalt.

Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneridiplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

"...." 20.....

Autor:

/ allkiri /

Töö vastab rakenduskõrgharidusõppe lõputööle/magistritööle esitatud nõuetele

"...." 20.....

Juhendaja:

/ allkiri /

Kaitsmisele lubatud

"...." 20.....

Kaitsmiskomisjoni esimeeson Mare Roosileht.....

/ nimi ja allkiri /

LIHTLITSENTS LÕPUTÖÖ ÜLDSUSELE KÄTTESAADAVAKS TEGEMISEKS JA REPRODUTSEERIMISEKS

Mina Darja Slobodjanova (sünnikuupäev: 19.11.1998)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose Siemens tööstuskontrolleri avatud koodiga platvormiga ühenduse loomine mille juhendaja on Sergei Ponomar,
 - 1.1. reprodutseerimiseks säilitamise ja elektroonilise avaldamise eesmärgil, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta kolmandate isikute intellektuaalomandi ega isikuandmete kaitse seadusest ja teistest õigusaktidest tulenevaid õigusi.

TalTech Inseneriteaduskond Virumaa kolledž

LÕPUTÖÖ ÜLESANNE

Üliõpilane: Darja Slobodjanova, 183582EDTR

Õppekava, peeriala: EDTR17/18 - Telemaatika ja arukad süsteemid, Protsesside automatiseerimine

Juhendaja(d): TalTech Virumaa kolledz, Sergei Ponomar, sergei.ponomar@taltech.ee

Lõputöö teema:

(eesti keeles) Prototüübi loomine kasutades tööstuskontroller Siemens ja arendus platvorm Arduino ja Raspberry Pi

(inglise keeles) Prototyping using the industrial controller Siemens and the development platform Arduino and Raspberry Pi

Lõputöö põhieesmärgid:

1. Koostamine õppematerjale, kuidas Arduino, Raspberry Pi ja Siemens`i kontrolleri ühendama
2. Uurida, kuidas kasutada Arduino ja Raspberry Pi koos tööstusliku kontrolleriga Siemens s7-1200 Uurida kuidas Arduino ja Siemens`i kontrolleri ühendama

Lõputöö etapid ja ajakava:

Nr	Ülesande kirjeldus	Tähtaeg
1.	Arduino, Raspberry Pi ja Siemens`i kontrolleri uurimine	oktoober
2.	Lõõputöö struktuuri koostamine	oktoober
3.	Uurida kuidas Paspberry Pi ja Siemens`i kontrolleri ühendama	november
4.	Uurida kuidas Arduino ja Siemens`i kontrolleri ühendama	november
5.	Andurite andmete edastamine	detsember
6.	Õppematerjale koostamine	detsember
7.	Lõõputöö esitamine	detsember
8.	Osalimine eelkaitsmine	detsember

Töö keel: Eesti keel

Lõputöö esitamise tähtaeg: "12" jaanuar 2021a

Üliõpilane: Darja Slobodjanova

"....."..... 20.....a

	/allkiri/	
Juhendaja: Sergei Ponomar		"....."..... 20.....a
	/allkiri/	
Programmijuht: Žanna Gratšjova		"....."..... 20.....a
	/allkiri/	

SISUKORD

EESSÖNA	7
LÜHENDITE JA TÄHISTE LOETELU.....	8
SISSEJUHATUS.....	9
1 KOMPONENDID.....	10
1.1 Arduino MKR1010 WIFI Rev2.....	10
1.2 Ethernet Shield.....	10
1.3 Arduino MKR Connector Carrier	11
1.4 Raspberry Pi	12
1.5 Siemens SIMATIC S7-1200	12
1.6 Andurid	13
2 KASUTATUD TARKVARA	15
2.1 TIA Portal.....	15
2.2 Arduino IDE	15
2.3 Thonny IDE.....	15
3 KASUTATUD TEEGID JA PROTOKOOLID.....	16
3.1 Simensi mikrokontrolleri andmeside protokollid	16
3.2 MODBUS TCP	16
3.3 PROFIBUS	17
3.4 Ethernet S7 Protokoll	18
3.5 Settimino.....	19
3.6 Snap7.....	19
4 ARDUINO JA SIEMENS'I ÜHENDAMINE	21
4.1 Ühendamine skeem	21
4.2 Andmete edastamine TIA portaalist Arduinosse	23
4.3 Andmete edastamine Arduinoga ühendatud anduritelt TIA portaali.....	25
5 RASPBERRY PI JA SIEMENS'I ÜHENDAMINE	27
5.1 Ühendamine skeem	27
5.2 Andmete saatmine TIA portaalist Raspberry Pi-le.....	28
KOKKUVÖTTE	31
SUMMARY	32
KASUTATUD KIRJANDUSE LOETELU	33
LISAD	35

EESSÕNA

Selle lõputöö antud teema pakkus välja Virumaa kolledzi õppejõud Sergei Ponomar. Käesoleva lõputöö kirjutamiseks andis TalTech Virumaa kolledz autorile tasuta kogu varustuse Samuti soovin tänada lõputöö juhti Sergei Ponomari abi ja nõuannete eest lõputöö kirjutamisel.

LÜHENDITE JA TÄHISTE LOETELU

Snap7 – Nimi suhtlusteeg

Settimino – Nimi suhtlusteeg

C++ - Programmeerimis keel

Arduino IDE – programmeerimiskeskkond Arduino jaoks

Linux – Operatsioonisüsteem

TIA Portal – (Totally Integrated Automation Portal) arenduskeskkond ja ühendus Siemensi kontrolleritega

PLC – (Programmable Logic Controller) Programmeeriv kontroller

Breadboard – Makettplaat

MPI – (Message Passing Interface) Asünkroonse liidese füüsilise tase standard

PROFINET – Kommunikatsiooni protokoll

OPC SERVER – (Open Platform Communications) tarkvaratehnoloogia, mis pakub ühtset liidest automatiseerimisobjektide haldamiseks

CP – (Central Processor) Keskseade

Grove Connector – Ühenduse tüüp

Liideseplaat – seade, mis võimaldab lisada emaplaadile porte ja ka olemasolevat liidesele.

LAD – (Ladder Diagram) Programmeerimis keel

SCL – (Structured Control Language) Programmeerimis keel

SISSEJUHATUS

Diplomitöö oli koostatud Raspberry Pi ja Arduino kiire ühendamiseks Siemensi mikrokontrollerile, mis aitab suurendada nende võimalust ja luua kiiremini erinevaid prototüüpe.

Käesoleva lõputöö põhieesmärk on seadistada Arduino ja Raspberry Pi ühendus Siemensi tööstusliku mikrokontrolleriga, mis võimaldab tulevikus kiiresti luua erinevaid prototüüpe.

Töö eesmärgi saavutamiseks on vaja:

1. Selgeks õppima kuidas Arduino plaat koostöötab Siemensi mikrokontrolleriga
2. Saada selgeks kuidas töötab üheplaadiline Raspberry Pi arvuti koos Siemensi mikrokontrolleriga
3. Õppematerjalide koostamine Arduino plaadi ühendamiseks Siemensi mikrokontrollerile
4. Õppematerjalide koostamine üheplaadiliste Raspberry Pi Siemensi mikrokontrollile
5. Andurite ühendamine ja andmete kättesaamine.
6. Kirjutada üles kättesaadud andmed

Käesolev diplomitöö on kiiresti moodustata prototüüpi, mis hakkab töötama koos Siemens S7-1200 ja-või Arduino, Raspberry Pi- plaadiga.

See aitab Siemensi S7-1200 mikrokontrolleril rakendada uusi võimalusi. Näiteks tootmises, on võimalik kontrollida selle aktuaalsust ja vajadust enne kallimate komponentide ostmisel ja uue idee realiseerimisel, kasutades odavamaid ja kättesaadavamaid hinna poolest.

1 KOMPONENDID

Käesolevas töö peatükis on lühikirjeldus ja tehniline teave komponentide kohta, mida lõputöö tegemisel kasutati. See peatükk sisaldab teavet Siemensi tööstusliku mikrokontrolleri, riist- ja tarkvaraplatvormide, laiendusplaatide ja kasutatud andurite kohta.

1.1 Arduino MKR1010 WIFI Rev2

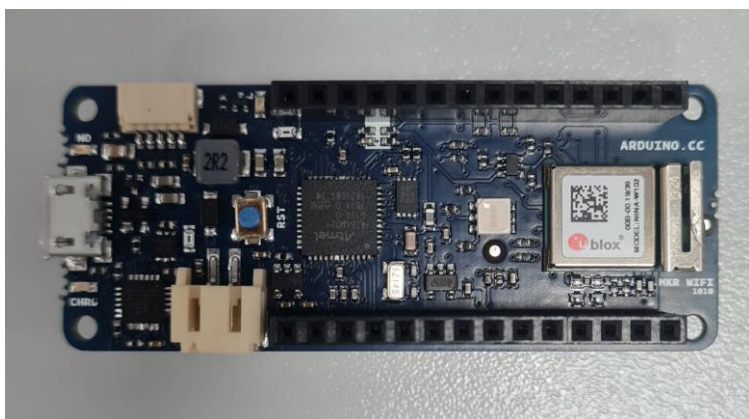
Arduino on miniatüürne plaat, millel on oma protsessor ja mälu. Plaadil on ka kontaktid mida saab ühendada laiendamise plaate, erinevaid andureid jne.

Arduino saab programmeerida nii, et see võimaldaks teha erinevaid töid, alates tule sisselülitamises, lõppedes kasvuhooone taimede kastmisega. Ühesõnaga Arduino annab võimalust looma lõpmatult erinevaid vidinaid ja mõtteid. [1]

Arduino programmeerimiseks kasutatakse Arduino IDE, peamine programmeerimis keel on C++. [5]

Arduino juurde saab ühendada Breadboardi, mil järgi saab ühendada veelgi rohkem erinevaid komponente, ja see ei nõua keevitamist.

Arduino MKR1010 WIFI Rev2 plaadil on 7 analoogsisendit ja üks analoogväljund, samuti 8 digitaalset sisendit/väljundit ja USB-pesa [11]



Joonis 1 Arduino MKR1010 WIFI Rev2

1.2 Ethernet Shield

Ethernet Shield – on laiendusplaat, mis võimaldab Arudinol töötada lokaalsetel võrgustikus, kasutades Etherneti kaabli. Antud plaat töötab «Arduino Ethernet library» abil, mis on juba ettenähtud seal. Ethernet Shieldil on mälukaardi pesa, mis võimaldab üleslaadida suurema andmete koguse, ning laadida ka erinevaid web-saite kuid sel juhul läheb vaja teist teegid. [10]. MKR Ethernet Shieldi laiendusplaadil on Rj45 Female

konnektor, mälukaardi pesa ja 7 analoogset sisse-välja, üks analoogväljund ja 8 digitaalset sisse/välja pesa. [12]

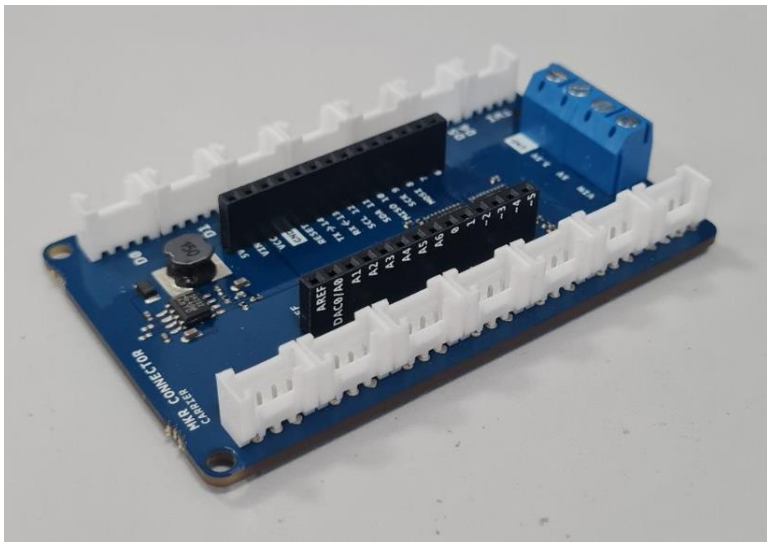


Joonis 2 Ethernet Shield

1.3 Arduino MKR Connector Carrier

Arduino MKR Connector Carrier on Arduino laiendplaat, mis annab Arduino MKR plaadil ühendada erineva koguse anduritel ja mooduleid, mis kasutavad Grove connector.

Laiendplaadil on 5 analoogset üksikpesa, 1 kahekordne digitaalne pesa, 5 digitaalsete ühendusega üksikpesa ja 1 kahekordne digitaalne pesa. Olemas ka eraldi väljapääs Serial ja TWI. Iga pesa sisse/välja toitepinge on 5B.



Joonis 3 Arduino MKR Connector Carrier

1.4 Raspberry Pi

Raspberry Pi on mini üheplaadiline arvuti, suurus nagu pangakaardil. Raspberry Pi peab ühendama monitoriga, hiirega ja klaviatuuriga. Plaadil on 4 pesa USB, JACK port, HDMI, Ethernet pistikühendus, mirco-SD pesa ja MicroUSB ühenduspesa. Raspberry Pi plaadil on GPIO mitmeotstarbeline sisend/väljund, mis annab võimalust kasutada Raspberry Pi robotehnika projektides ja «Smart Home» seadmetes. [2]

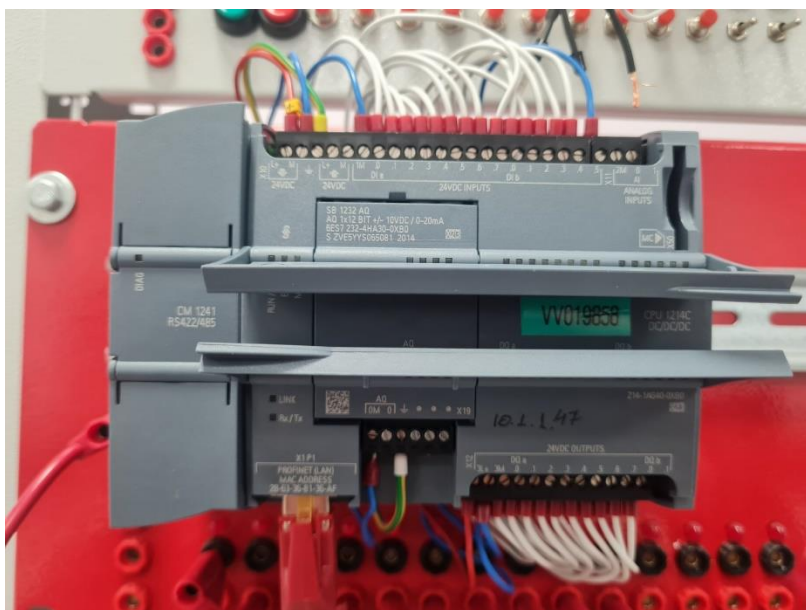


Joonis 4 Raspberry Pi

1.5 Siemens SIMATIC S7-1200

Siemens on tootmis mikrokontroller, mis saab lahendada erinevaid automatiseerimise ülesandeid. Antud kontrolleril on mooduline konstruktsioon ja universaalne otstarve. Kontroller saab teenindada 10 kuni 248 diskreetset sisse-välja pesa ja 2 kuni 51 analoogset sisse-välja pesa. [3] Siemens SIMATIC S7-1200 mikrokontrolleril on 14 digitaalset sisse ja 10 välja pesa ning 2 analoogilist väljumispesat. Bitimälu on 8192 baiti.

Siemensil on oma IP-aadress. Kõrvalkaanel on kirjas kontrolleri põhiline info.



Joonis 5 Siemens SIMATIC S7-1200

1.6 Andurid

DHT22 on kõrgpingeline andur, mis mõõdab temperatuuri diapsoonis -40 kuni 80 kraadi tselsiuse järgi, ja niiskust alates 5 kuni 99 protsenti. Antud andur töötab 8-Bit kontrolleri abil, mis omamoodi moodustab analoogset digitaalväljumist ja näitab temperatuuri ja niiskuse andmeid. Tootja sõnadest anduri täpsus on 2% niiskust ja $0,5$ kraadi tselsiuse järgi.

Siseminepinge varieerub $3,3$ kuni 6 V.



Joonis 6 Temperature&Humidity Sensor Pro

PIR on liikumisandur, mis leiab infrapuna signaali, mis on moodustatud liikumisega. Otsingu diapsoon ja kiiruse vastust saab reguleerida 2 potentsiomeetrite abil, mis asuvad plaadil. Vastuse kiirus on $0,3$ sekundit kuni 25 sekundit. Maksimaalne ulatus on 6 meetrit, ja nurk 120 kraadi

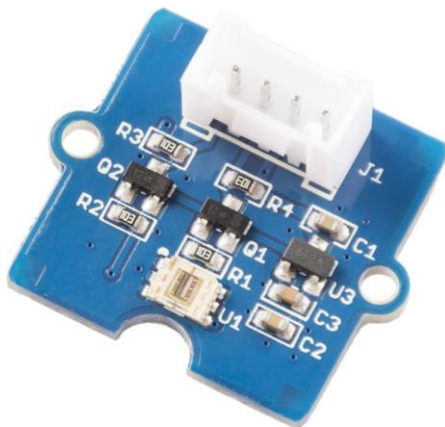
Siseminepinge varieerub 3 kuni 5 V



Joonis 7 PIR Motion Sensor v1.2

Light Sensor on andur mis saab moodustada valguse intensiivsust digitaalsignaalsiks. Antud andurit saab kasutada valguse intensiivsuse mõõtmiseks. Mõõtmis diapasoone on 0,1 kuni 40000 luks. -40 kuni 85 kraadi tselsiuse järgi on temperatuur, mil andur saab töötada.

Siseminepinge varieerub 3 kuni 5 B.



Joonis 8 Digital Light Sensor v1.1

2 KASUTATUD TARKVARA

Selles peatükis kirjeldatakse tarkvara, mida vajate Arduino ja Raspberry Pi seadistamiseks Siemensi mikrokontrolleriga ühenduse loomiseks.

2.1 TIA Portal

TIA portaal on Siemensi automatiseerimiskomponentidega erinevate projektide tarkvaraarenduskeskkond. Tia portaal võimaldab seadmeid konfigureraida ja programmeerida. Lõputöö kirjutamisel kasutab autor peamiselt globaalset andmeplokki (tähistega DB). Globaalsed andmeplokid võivad sisaldada erinevaid kasutajaandmeid, mida saab kasutada terves programmis, ning kõik kasutajaprogrammi plokid pääsevad ligi globaalsele andmeplokile. Globaalsetes andmeplokkides saab salvestada erinevat tüüpi andmeid. Arduino ühendamisel Siemensi mikrokontrolleriga kasutab autor peamiselt globaalset andmeplokki. Raspberry Pi ja Arduino ühendamisel Siemensi PLC-ga töötab autor erinevate andmetüüpidega nagu Bool, int, real, char. String, data, time, timer, array. Seda tüüpi andmetega töötades õppis autor, kuidas neid ühest seadmest teise üle kanda ja vastu võtta. Samas on TIA portaalil ka teisi andmetüüpe, näiteks: USint, Sint, UInt, UDIInt, Real, LReal jne.

2.2 Arduino IDE

Arduino IDE on programmeerimiskeskond, mille saab ametlikult veebisaidilt tasuta alla laadida ja kus saab kirjutada koodi C ja C ++ keeles. Põhikeel on aga C ++. Arduino IDE võimaldab luua programme Arduino plaatidele ja teiste tootjate plaatidele.

Arduino IDE-s töötamiseks peab määrama kasutatava pordi ja plaadi. Samuti Arduino IDE-s lisada erinevaid teke - seda saab teha «Manage Libraries» kaudu.

2.3 Thonny IDE

Thonny IDE on programmeerimiskeskond, mis on Raspbiani operatsioonisüsteemi eelinstallitud. Programmeerimiskeskond kasutab Pythoni programmeerimiskeelt, mis on üks kõige populaarsemateist programmeerimiskeeli maailmas.

Thonny IDE-sse saab laadida erinevate raamatukoguid, mis laiendab oluliselt võimalusi.

3 KASUTATUD TEEGID JA PROTOKOOLID

Selles peatükis kirjeldatakse selle lõputöös kasutatud protokolle ja teeke.

3.1 Simensi mikrokontrolleri andmeside protokollid

Siemensni mikrokontrolleri andmeside jagamisel kasutatakse järgmisi protokolle: Ethernet S7 Protokool, Modbus, PROFIBUS jms.

Antud diplomitöös autor kasutas Ethernet S7 protokooli, kuna töö oli tehtud ainult Siemensi mikrokontrolleritega.

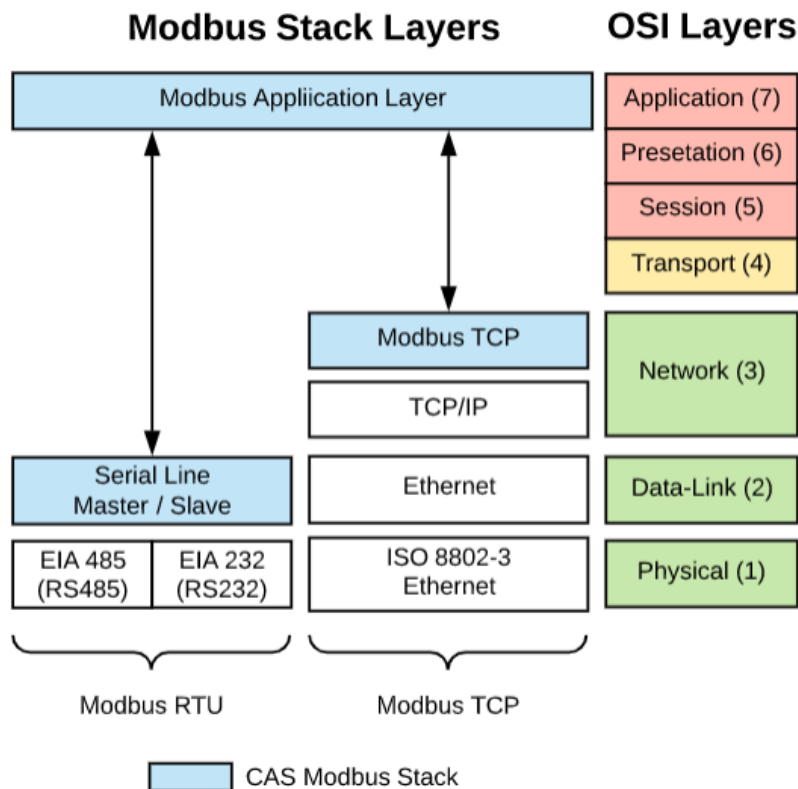
3.2 MODBUS TCP

Modbus - on kommunikatsiooni protokool, mida kasutatakse elektroonikaseade ühendustes. Modbus TCP protokooli ühenduses kasutatakse PROFINETi liidet keskseadet. Modbus TCP kasutamiseks läheb vaja STEP 7 basic tarkvara, alates V10.5, kuhu on lisatud sideplokkide teegi.

Modbusi TCP-protokolli kasutavas võrgus suhtlevad seadmed «ülem-alluv» põhimõttel. Modbus TCP saab suhelda ainult kahe seadme vahel.

Pildil 9 on näidatud Modbusi jaoks mõeldud OSI võrgumudel. Antud töös autor kasutab ainult Modbus TCP. Pildil on ka näidatud, et Modbus TCP kasutab: füüsilist taset, andmesite taset ja rakendustaset.

Võrgumudeli esimesel ja teisel tasel kasutatakse Ethernet protokooli, kolmandal ja neljandal tasel kasutatakse aga TCP/IP protokooli. Viimasel tasel kasutatakse Modbus protokooli.



Joonis 9 OSI võrgumudel MODBUSi jaoks

3.3 PROFIBUS

PROFIBUS on protokoll, mis on ühenduslülits kontrolleri ja juhtimissüsteemi vahel, samuti andurite ja täiturmehhanismidega. PROFIBUS võimaldab samaaegset andmevahetust kõrgema taseme süsteemidega.

Joonisel 10 on näidatud OSI võrgumudel PROFIBUSi jaoks. Andmete edastamisel kasutab PROFIBUS kolme taset: füüsilist taset, andmesidetase ja rakendustase. Füüsiline tase on andmeedastustehnoloogia, sellel tasel edastatakse teavet bittidena, kasutades näiteks keerdpaari, kiudoptilist või infrapunavõrku (Wi-Fi).

Andmeside tase vastutab ülemate ja alamseadmete vahelise ühenduse eest (mis peaks olema piisavalt kiire). Andmesidekihti PROFIBUSis nimetatakse tööstuslikuks sidekanaliks (FDL), sellel tasemel kasutatakse «ülem-alluv» põhimõtet. Rakenduskiht vastutab andmete edastamise eest kasutajale talle arusaadavas vormis. Rakenduskiht saab kasutada PROFIBUS DB protokoll, mis rakendab funktsioone DB-V0 (kasutatakse tsükliliseks andmevahetuseks ja diagnostikaks), DP-V1 (kasutatakse atsükliliseks ja tsükliliseks andmevahetuseks ja häirete käsitlemiseks), DP-V2 (kasutatakse leviedastuseks sideandmed või isokroonrežiim) [6]

	User program		Application profiles
7	Application Layer		PROFIBUS DP Protocol (DP-V0, DP-V1, DP-V2)
6	Presentation Layer		Not used
5	Session Layer		
4	Transport Layer		
3	Network Layer		
2	Data link Layer		Fieldbus Data Link (FDL): Master Slave principle Token principle
1	Physical Layer		Transmission technology
	OSI Layer Model		OSI implementation at PROFIBUS

Joonis 10 võrgumudel OSI PROFIBUSi jaoks

3.4 Ethernet S7 Protokoll

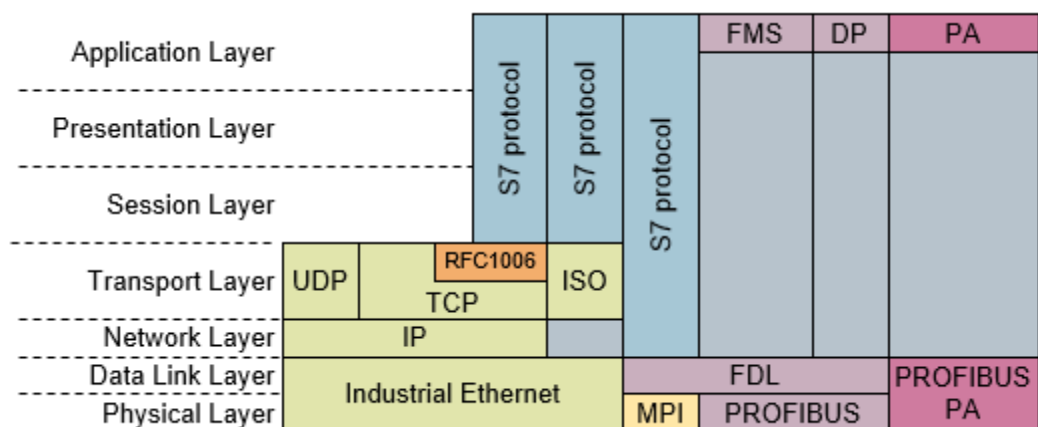
S7 protokoll on rakendustase protokoll, mis võimaldab andmevahetust SCADA süsteemides kontrolleri ja teiste S7 protokolliga toetavate seadmete vahel andmevahetust.

See protokoll võib töötada kas tavalises PROFIBUS-võrgus või MPI-võrgus.

Kui kasutada ühenduse loomiseks MPI-võrku, ei pea ühendamiseks kontrolleri jaoks täiendavat CP-kaarti installima, kuid mõned parameetrid seda tüüpi ühenduse korral on piiratud (näiteks andmevahetuskirgus).

Protokoll S7 käivitamisel võrgus PROFIBUS nõuab, et kontrolleri süsteemidesse oleks paigaldatud lisa CP, mis võib kindlustada võrgu täielikku paindlikkust ja pakkuda palju eeliseid, näiteks kiirust.

Arvuti ühendamiseks MPI- või PROFIBUS-võrku tuleb arvutisse installida Siemensi tarkvara ja liideseplaat. [6]



Joonis 11 Ethernet S7 Protocoll – OSI võrgumudel

3.5 Settimino

Settimino on avatud lähtekoodiga Etherneti teek, mida saab kasutada Arduino liidestamiseks Siemens S7 PLC-ga. Settimino toetab ka PLC 1200/1500, LOGO 0BA7 ja S7200. Settimino on Snap7, mis on kohandatud töötama koos Arduinoga.

Settimino koosneb kolmest failist:

1. Settimino.h – vastutab Arduino ja Siemens S7 PLC vahelise ühenduse eest
 2. Settimino.cpp – vastutab juhtmega või traadita ühenduse automaatse loomise eest
 3. Keywords.txt – vastutab võimaluse eest lisada märksõnu süntaksi esiletõstmiseks
- Settiminoga töötamiseks tuleb see programmeerimiskeskonda importida.

Peamine eksporditav klass on S7Client (antud juhul Siemens), mis suudab ühendada ja edastada andmeid Arduinost Siemensi kontrolleri ja vastupidi. S7Helper on globaalne objekt, mis võimaldab teil baitpuhvrist soovitud vormingus väärtusi hankida.

Settiminol on täielik juurdepääs PLC-mälule ja kolmele mälumudeli optimeerimiseks. Settimino kasutab Arduino Etherneti standardraamatukogu, mis tähendab, et Settimino saab koos eksisteerida teiste klientide ja serveritega. PDU lugemiseks sisemisse puhvri kuulub 3 ms ja 24 ms 1024 baiti välisse puhvrise.

Settiminoga töötamiseks läks vaja Siemensi PLC-ga Etherneti kaabli kaudu ühenduse loomiseks Siemens S1200 kontrolleri, Arduino plaati (antud juhul Arduino MKR WiFi 1010) ja Ethernet Shieldi laiendusplaati. Samuti laadida alla Settimino teek, mis on ametlikul veebisaidil vabalt saadaval.

3.6 Snap7

Snap7 on avatud lähtekoodiga 32- või 64-bitine mitmeplatvormiline Etherneti teek, mis võimaldab koostalitlusvõimet Siemens S7 kontrolleri. Samuti teek toetab osaliselt teisi protsessoreid.

Snap7 on teistest teegist sõltumatu, sellel pole ka kohandamist ja seda pole vaja installida. Andmeedastuseks on kaks võimalust: sünkroonne ja asünkroonne ning andmevoo jaoks kaks võimalust: küsitletud ja soovimatu.

Snap7 loodi algselt selleks, et ületada OPC-serverite piirangud suurte ja kiirete andmete edastamisel tööstuslikes keskkondades. Kuid see töötab hästi ka väikeste Linuxipõhiste plaatidega (nagu Raspberry Pi) või Arduinoga. [8]

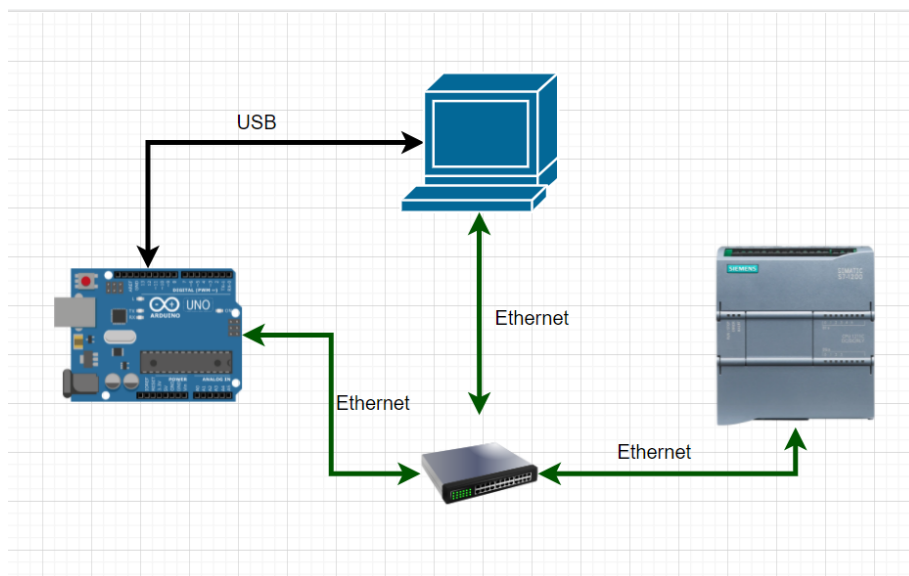
Snap7-ga töötades tuleb märkida, et Snap7 jaoks on olemas üksikasjalik dokumentatsioon (vabalt saadaval), mis aitab seda teeki mõista. Teine eelis on suhteline kasutusmugavus ja võimalus töötada erinevate platvormidega.

4 ARDUINO JA SIEMENS'I ÜHENDAMINE

See peatükk sisaldab juhtmestiku skeemi koos kirjeldusega, kasutatavate seadmetega, teabega andmete saatmise ja vastuvõtmise kohta Siemensi mikrokontrollerist Arduino platvormile

4.1 Ühendamine skeem

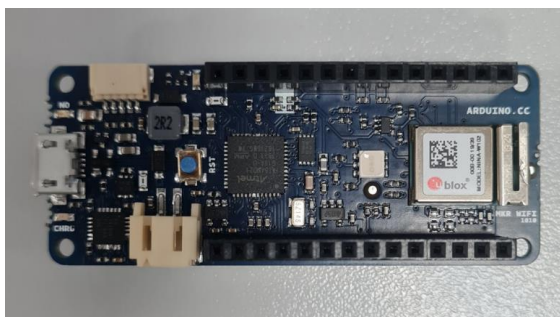
Joonisel 12 on näidatud Arduino ühendusskeem Siemens S7-1200 tööstusliku kontrolleriiga. Ühendamiseks ühendage Arduino plaat ja Ethernet Shield, seejärel ühendage Etherneti kaabel Etherneti Shieldi ja Switchiga. Järgmiseks ühenda arvuti Etherneti kaabli kaudu Switchiga ning USB kaabli kaudu ka Arduino arvutiga ning lõpuks ühenda Siemensi mikrokontroller ja Switchi Ethernet kaabli kaudu.



Joonis 12 Arduino ühendusskeem Siemens S7-1200 tööstusliku kontrolleriiga

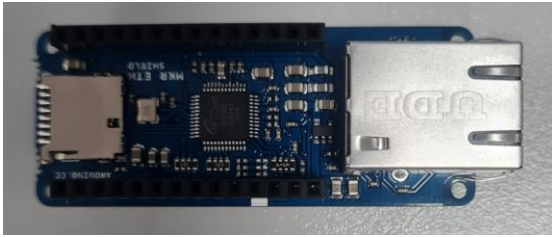
Arduino ühendamiseks Siemensi mikrokontrolleriga läheb vaja:

- 1) Arduino plaat (lõputöö jaoks kasutati Arduino MKR1010 WIFI Rev2 plati)



Joonis 13 Arduino MKR1010 WIFI Rev2

- 2) Laiendusplaat MKR Ethernet Shield



Joonis 14 MKR Ethernet Shield

3) Switch Netgear 8 ports



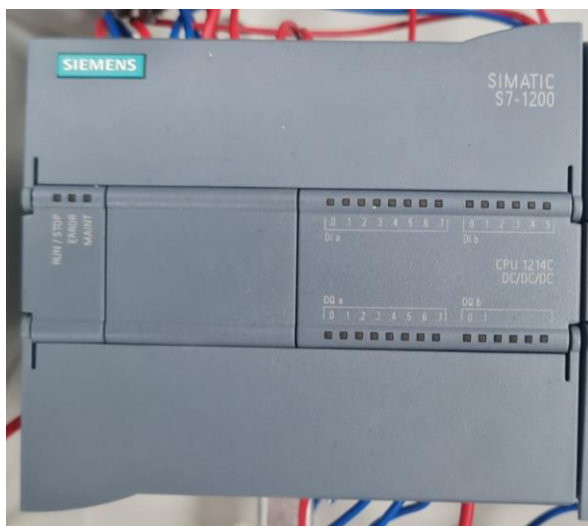
Joonis 15 Switch Netgear 8 ports

- 4) Arvuti
- 5) USB kaabel
- 6) Arduino MKR Connector Carrier
- 7) Ethernet kaabel



Joonis 16 Kaabel Ethernet

8) Tootmis mikrokontroller Siemens SIMATIC S7-1200

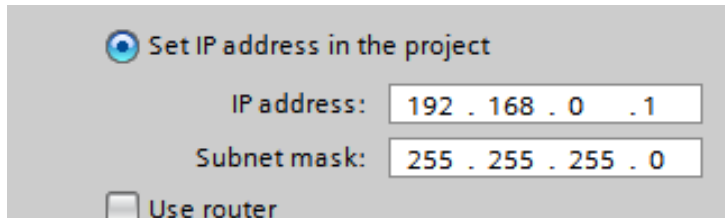


Joonis 17 Siemens SIMATIC S7-1200

Arvutisse peab installima Arduino IDE programmeerimiskeskonda. Seejärel ühendada Arduino ülaltoodud skeebi abil TIA portaaliga (joonis 12) ja nüüd saab koodi kirjutada.

4.2 Andmete edastamine TIA portaalist Arduinosse

Arduino ja Siemens mikrokontrolleri ühendamiseks peab seadistama võrgu. Selleks peab siseneda TIA portaali sätetesse ja määrata seal IP-aadress 192.168.0.1 Siemens PLC ja alamvõrgu mask 255.255.255.0.



Joonis 18 IP-aadress ja alamvõrgu mask TIA Portalis

Samuti peab seadetes lubama funktsiooni PUT / GET, mis võimaldab võrgus andmeid saata ja lugeda. Ka Arduinos peab määrama kohaliku IP-aadressi 192.168.0.2, Siemensi kontrolleri IP-aadressi - 192.168.0.1 ja MAC-aadressi 0x90, 0xA2, 0xDA, 0x0F, 0x08, 0xE11. Selleks, et Arduino saaks kontrolleriga ühenduse luua, peavad need asuma samas alamvõrgus.

```
byte mac[] = {
  0x90, 0xA2, 0xDA, 0x0F, 0x08, 0xE11 // on vaja kirjutada MAC-aadress
};

IPAddress Local(192, 168, 0, 2); // on vaja kirjutada kohalik IP-aadress
IPAddress PLC(192, 168, 0, 1); // Mikrokontrolleri IP-aadress
```

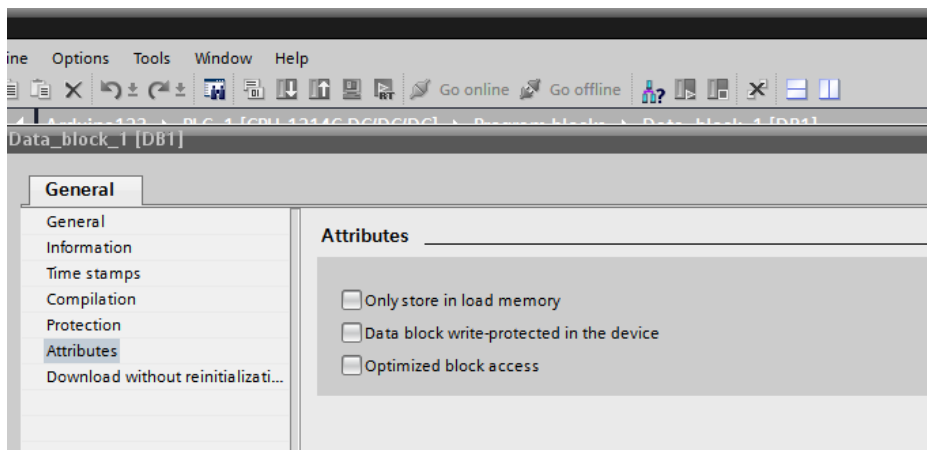
Joonis 19 Võrgu seadistamine Arduino

Andmete saatmiseks TIA portaalist Arduinosse lisati TIA portaali Data_Block, kuhu sisestati erinevat tüüpi andmed: Bool, int, real, char. String, datetime, data, time, counter, timer, array. Igal muutujal on oma address.

Data_block_1							
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writ...
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Bool	Bool	0.0	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Int	Int	2.0	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	String	String	4.0	'hello'	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Char	Char	260.0	'A'	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Real	Real	262.0	12.56	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Array	Array[0..1] of Bool	266.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Joonis 20 Data_Block TIA Portaalis

Samuti peate Data_Block seadetes blokeerima optimeeritud juurdepääsu plokkidele, et sellel plokil oleks n.ö "nihe", mida saab kasutada adresseerimiseks.



Joonis 21 Data_Block häälestamine

Arduino ja Siemensi PLC koostöökäsitamiseks laaditi Settimino ja Platform.h teek Arduinosse ning andmeedastuse kood kirjutati Arduino IDE-sse, milles kasutati erinevaid funktsioone. Arduino IDE-le määrati Data_Block number, mida kasutatakse TIA portaalis, määrati ühendus PLC ja Internetiga, määrati kirjutamiseks eraldatud mälu maht ja muutujad, mis sobisid muutujate tüüpidega TIA portaalis (massiivi kuvamiseks kasutati tsükli "for".)

Pärast võrgu seadistamist, koodi seadistamist ja käivitamist Arduino IDE-s, programmi seadistamist ja käivitamist TIA portaalis ning monitori avamist, teabeväljundeid Arduino IDE-s, näete andmeid, mida sisestati TIA portaalis. Ja kui muuta väärtusi TIA portaalis, muutuvad need ka Arduino IDE-s. TIA portaalis ja Arduino IDE-s võib andmetüübi nimi olla erinev, numbri 1 all olevast tabelist on näha, milline andmetüüp TIA portaalis vastab Arduino IDE-s olevale tüübile.

Andmetüüp TIA Portal	Andmetüüp Arduino IDE	Suurus bittides TIA Portal	Suurus baitides Arduino IDE
Bool	Bool	1	1
Int	Int	16	2
Real	Float	32	4
Char	Char	8	1
String	String	Alates 1 kuni SRAM lõpuni	Alates 1 kuni SRAM lõpuni
Array of ... [Int], [Bool] ja n.e	Array	Sõltub andmetüübist	Sõltub andmetüübist

Tabel 1 Andmetüüp TIA Portaalis ja Arduino IDE

4.3 Andmete edastamine Arduinoga ühendatud anduriteelt TIA portaali

Andmete edastamiseks Arduinost TIA portaali kasutati järgmisi andureid: Temperature & Humidity Sensor PRO (DHT22) – temperatuuri- ja niiskusandur, PIR Motion Sensor – liikumisandur ja Digital Light Sensor – valgusandur.

Kõik andurid kasutasid Grove'i ühendust.

Arduino IDE programmeerimiskeskonda lisati erinevaid teeke, mis on vajalikud andurite töötamiseks.

```

tia_portal_andurid____
#include <Platform.h>
#include "Settimino.h"
#include <Wire.h>
#include <Digital_Light_TSL2561.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>

```

Joonis 22 Lisamine teek

Igal anduril oli oma PIN (port, kuhu on ühendatud Arduino MKR Connector Carrier'il olev andur), mis oli koodis märgitud.

Anduri DHT22 temperatuuri ja niiskuse mõõtmiseks laaditi Arduino IDE-s järgmised teegid: <Adafruit_Sensor.h>, <DHT.h> ja <DHT_U.h>. Valgusanduri töötamiseks laaditi teegid <Wire.h> ja <Digital_Light_TSL2561.h>. Täiendavaid teeke polnud liikumisanduri töötamiseks vaja.

Lisaks kirjutati iga anduri tööks funktsioon Arduino IDE programmeerimiskeskonda.

Kui arvestada temperatuuri mõõtmise funktsiooni, kasutatakse temperatuuri registreerimiseks muutujatüüpi "float". Ja andurilt andmete salvestamiseks kasutatakse "sensors_event_t" tüüpi, mida on vaja andurilt saadud andmete "Encapsulation"

```
float TemperatureSensor() {  
    float temperature = 0;  
    sensors_event_t event;  
    dht.temperature().getEvent(&event);  
    temperature = event.temperature;  
    return temperature;  
}
```

Joonis 23 Koodi osa

Andmete ülekandmiseks andurilt Arduino IDE-sse kasutati eri tüüpi andmete salvestamiseks ühte lahtrisse spetsiaalset andmetüüpi "union", samuti kasutati "Reverse", kuna Arduinol ja Siemensil on erinev baitide järjestus.

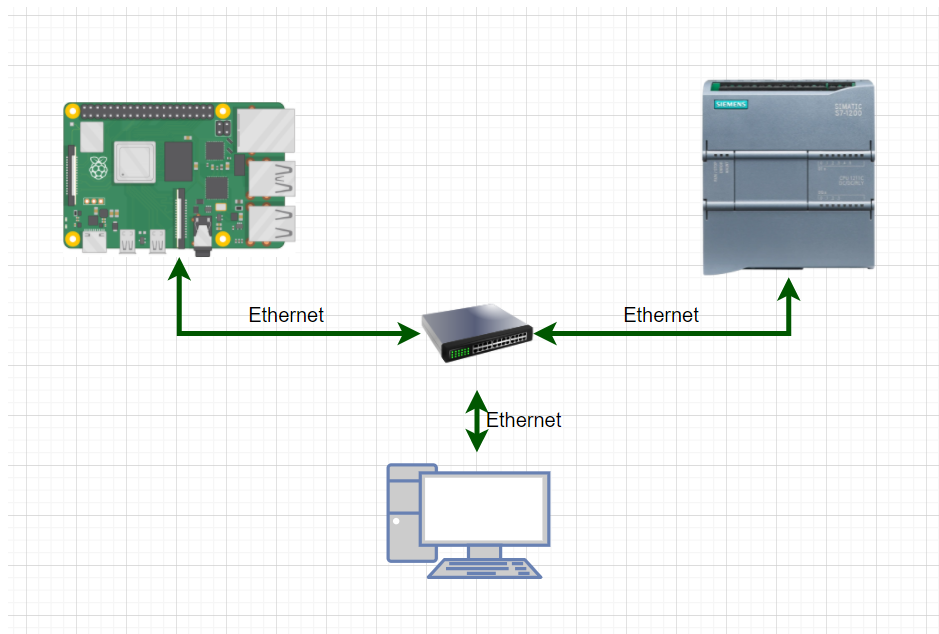
Arduino loeb teavet suuremast väiksemale baidini, see tähendab "vasakult paremale", Siemens aga väiksemalt suuremale. Seega võimaldab "Reverse" lugeda teavet Siemensi PLC jaoks sobival kujul.

5 RASPBERRY PI JA SIEMENS'I ÜHENDAMINE

Selles peatükis kirjeldatakse, kuidas ühendada Raspberry Pi Siemensi tööstusliku mikrokontrolleriga. See peatükk sisaldab juhtmestiku skeemi koos kirjelduse, kasutatavate seadmete, andmete saatmise ja vastuvõtmise kohta Siemensi mikrokontrollerist Raspberry Pi platvormile.

5.1 Ühendamine skeem

Joonisel 20 on kujutatud Raspberry Pi ühendusskeem Siemensi tootmismikrokontrolleriga. Ühenduse loomiseks peab Raspberry Pi ühendama Etherneti kaabli kaudu Switchiga. Järgmiseks tuleb Etherneti kaabli kaudu Switchiga ühendada arvuti ja Siemensi mikrokontroller. Seejärel ühendada HDMI-kaabli abil monitor ja Raspberry Pi ning seejärel ühendame Raspberry Pi toiteallika.



Joonis 24 Raspberry Pi ühendusskeem Siemensi tootmismikrokontrolleriga

Raspberry Pi ühendamiseks Siemensi mikrokontrolleriga läheb vaja:

- 1) Monitor Raspberry Pi jaoks
- 2) Üheplaadiline arvuti Raspberry Pi



Joonis 25 Raspberry Pi

- 3) Arvuti
- 4) Tootmis mikrokontroller Siemens SIMATIC S7-1200
- 5) HDMI kaabel



Joonis 26 HDMI kaabel

- 6) Switch Netgear 8 ports
- 7) Mälukaart
- 8) Arvutihir
- 9) Klaviatuur
- 10) Ethernet kaabel

Alguses peab installima mälukaardile Raspberry Pi operatsioonisüsteemi - Raspbian, mis põhineb Debianil. Järgmiseks tuleb Raspberry Pi üheplaadiarvutisse paigaldada mälukaart ning ühendada see monitoriga (HDMI kaabli kaudu), hiire ja klaviatuuriga. Seejärel ühendage Raspberry Pi ja Siemens lülitiga.

5.2 Andmete saatmine TIA portaalist Raspberry Pi-le

Raspberry Pi ühendamisel Siemens mikrokontrolleriga töötas autor järgmistega andmetüüpidega: Bool, Int ja String.

Data_block_1							
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writ...
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Bool	Bool	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Int	Int	2.0	10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	String	String	4.0	'hi'	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Joonis 27 Data_Block TIA Portaalis

Võrgu seadistamine ja TIA portaaliga töötamine Raspberry Pi-ga ühenduse loomine on sama, mis Arduinoga ühenduse loomisel. Kuid Thonny IDE-s on võrgu konfigureerimiseks vaja ainult mikrokontrolleri IP-aadressi, kuna Raspberry Pi kohalik IP-aadress on vaikimisi juba määratud.

```
4
5 c = snap7.client.Client()
6
7 c.connect("192.168.0.1", 0, 1) # Mikrokontrolleri IP-aadress
8 c.get_connected()
9
```

Joonis 28 Koodi fragment

Raspberry Pi ja Siemens PLC koostöötamiseks lisati Thonnyle teegid: "Snap7", "ctypes", Snap 7 teegist imporditi "util".

```
siemenss7.py ×
1 import snap7 # Teegiühendus
2 import ctypes # Teegiühendus
3 from snap7 import util # Teegiühendus
```

Joonis 29 Koodi osa

Thonnys kirjutati andmete edastamiseks kood, millele lisati muutujad ja aadressid, kust andmeid võtta, ning info kuvamise funktsioon. TIA portaalist andmete saatmiseks kasutati "get" funktsiooni, kus on nähs, milliselt aadressilt tuleb andmeid lugeda.

```
16 print(data)
17 print("\n")
18 d = util.get_bool(data,0,0)
19 print(d)
```

Joonis 30 Koodi osa

Get-funktsiooni asemel kasutati andmete saatmiseks Raspberry Pi-st mikrokontrollerisse seatud funktsiooni «set». Loodi puhver, kuhu kirjutati andmed. Ka funktsioonis "set" oli vaja märkida edastatavate andmete tüüp, lahtri aadress salvestamiseks ja millist teavet on vaja salvestada. Tuleb märkida, et Raspberry Pi-st Siemensi mikrokontrollerisse saadetud andmetüüp String kuvati kuueteistkümnendsüsteemis ja normaalseks kuvamiseks peate kasutama TIA portaalis SCL programmeerimiskeelt (autor kasutas LAD programmeerimist keel TIA portaalis lõputöö kirjutamisel).

```
30 buffer = bytearray([0b00000000])
31 util.set_bool(buffer, 0, 0, False)
32
33 bu1 = bytearray(256)
34 util.set_bool(bu1, 0, 0, False)
35 util.set_int(bu1, 2, 55)
36 util.set_string(bu1, 0, 'qwdw', 256)
```

Joonis 31 Koodi fragment

KOKKUVÕTTE

Lõputöö püstitatud eesmärk sai täidetud - õnnestus Arduino ja Raspberry Pi ühendada Siemensi tööstusliku mikrokontrolleriga

Lõputöö kirjutades oli loetud palju informatsiooni erinevate kontrolleritest kasutatavate andmeedastusprotokollide kohta, Arduinost, Raspberry Pi ja Siemensi mikrokontrolleri kohta. Uuriti, kuidas ühendada Arduino ja Raspberry Pi plaate Siemens S7-1200 tööstusliku mikrokontrolleriga. Lisaks sellele, kuidas saata andmeid Siemensi mikrokontrollerist Arduino ja Raspberry Pi plaatidele ning kuidas saata andmeid Arduino ja Raspberry Pi plaatidega ühendatud anduritelt Siemensi mikrokontrollerile, tehti TIA portaalis Thonny IDE erinevat tüüpi andmetega tööd. ja Arduino IDE ning kirjutatud 5 dokumentatsioonietappi Arduino ja Raspberry Pi ühendamiseks Siemensi mikrokontrolleriga

Selle lõputöö tegemisel puutus autor kokku erinevate probleemidega, näiteks vajaliku info puudumine kiires ligipääsus ning erinevad probleemid tekkisid Arduino ja Raspberry Pi seadistamine.

Selle lõputöö abil saab kiiresti ühendada Arduino ja Raspberry Pi Siemensi mikrokontrolleriga, mis aitab luua erinevaid prototüüpe.

SUMMARY

In this thesis is described how to connect Arduino and Raspberry Pi boards to Siemens microcontroller

The thesis contains information and technical data of the components used when setting up the connection. Expansion boards, sensors and etc. are described.

To set up the connection, the Settimino and Snap7 libraries were studied. Also studied various protocols, for example, PROFIBUS, Ethernet S7 and Modbus, the description of which is in the thesis. Also in the thesis there is a description of the OSI network model of these protocols.

The thesis contains diagrams describing how to connect Arduino and Raspberry Pi boards to a Siemens microcontroller with code fragments.

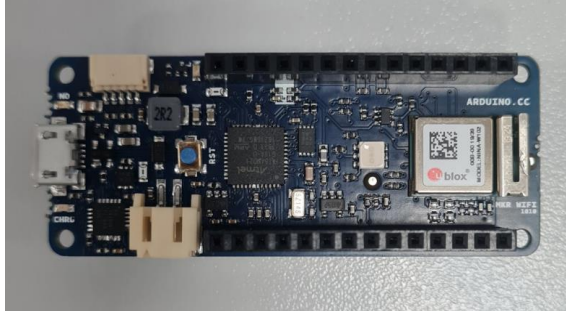

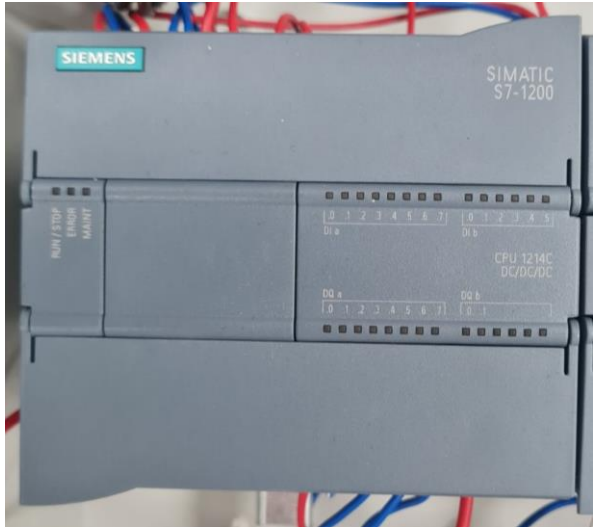

KASUTATUD KIRJANDUSE LOETELU

1. Что такое Arduino и зачем это нужно
<https://amperka.ru/page/what-is-arduino> (17.10.21) [Online]
2. Ben Nuttall, Getting started with Raspberry Pi
<https://opensource.com/article/16/12/getting-started-raspberry-pi> (17.10.21) [Online]
3. Siemens S7-1200 Программируемый контроллер
<https://www.siemens-pro.ru/components/s7-1200.htm> (15.10.21) [Online]
4. Snap7
<https://github.com/gijzelaerr/python-snap7> (21.10.21) [Online]
5. What is an Arduino?
<https://learn.sparkfun.com/tutorials/what-is-an-arduino/all> (17.10.21) [Online]
6. Siemens PROFIBUS/MPI S7 Protocol & Siemens Industrial Ethernet S7 protocol ID:8
<https://igss.schneider-electric.com/plc-scada-driver-8/> (25.10.21) [Online]
7. Communication with PLC devices SIMATIC by the Siemens company
<https://www.promotic.eu/en/pmdoc/Subsystems/Comm/PLC/Simatic/Simatic.htm> (30.10.21) [Online]
8. Step7 Open Source Ethernet Communication Suite
<http://snap7.sourceforge.net/> (2.11.21) [Online]
9. Руководство по программированию S7-1200/S7-1500
<https://assets.new.siemens.com/siemens/assets/api/uuid:6f74436a53942bf8f1c7b6ed34d2ae72e92ed8a8/programming-guideline-v14-rus.pdf> (15.11.21) [Online]
10. Ethernet Shield и Arduino – основы
<https://arduino-diy.com/arduino-ethernet-shield-osnovi> (5.11.21) [Online]
11. Arduino MKR Wifi 1010
<https://store.arduino.cc/products/arduino-mkr-wifi-1010> (5.11.21) [Online]
12. Arduino MKR ETH Shield
<https://store.arduino.cc/products/arduino-mkr-eth-shield?selectedStore=eu> (5.11.21) [Online]
13. What properties, advantages and special features does the S7 protocol offer?
<https://support.industry.siemens.com/cs/document/26483647/what-properties-advantages-and-special-features-does-the-s7-protocol-offer-?dti=0&lc=en-US> (01.12.21) [Online]
14. CAS MODBUS STACK
<https://store.chipkin.com/services/stacks/modbus-stack> (13.11.21) [Online]
15. Виктор Денисенко, Протоколы и сети Modbus и Modbus TCP
<https://www.cta.ru/cms/f/435973.pdf> (20.11.21) [Online]

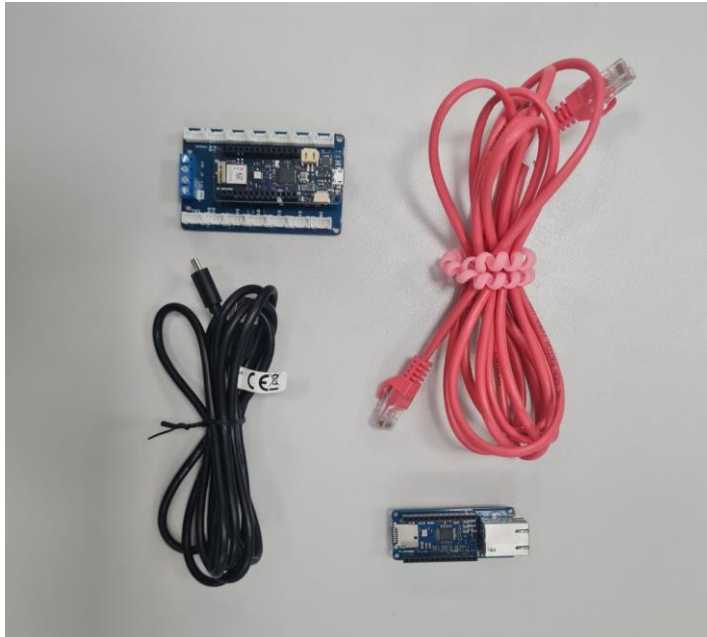
16. Grove - Temperature&Humidity Sensor Pro(DHT22)
https://wiki.seeedstudio.com/Grove-Temperature_and_Humidity_Sensor_Pro/
(02.12.21) [Online]
17. Grove - Digital Light Sensor - TSL2561
<https://www.seeedstudio.com/Grove-Digital-Light-Sensor-TSL2561.html>
(02.12.21) [Online]
18. Grove - PIR Motion Sensor
<https://www.seeedstudio.com/Grove-PIR-Motion-Sensor.html> (02.12.21) [Online]

LISAD

Siemens PLC ühendamine Arduino arendusplatvormiga

Komponendid	Kogus
Arduino (MKR1010) 	1 tk.
Ethernet Shield 	1 tk.
Siemens PLC 	1 tk.
Kaabel Ethernet	3 tk.
Kaabel USB 2.0	1 tk.
Switch 	1 tk.
Arvuti, kus on Arduino IDE	1 tk.

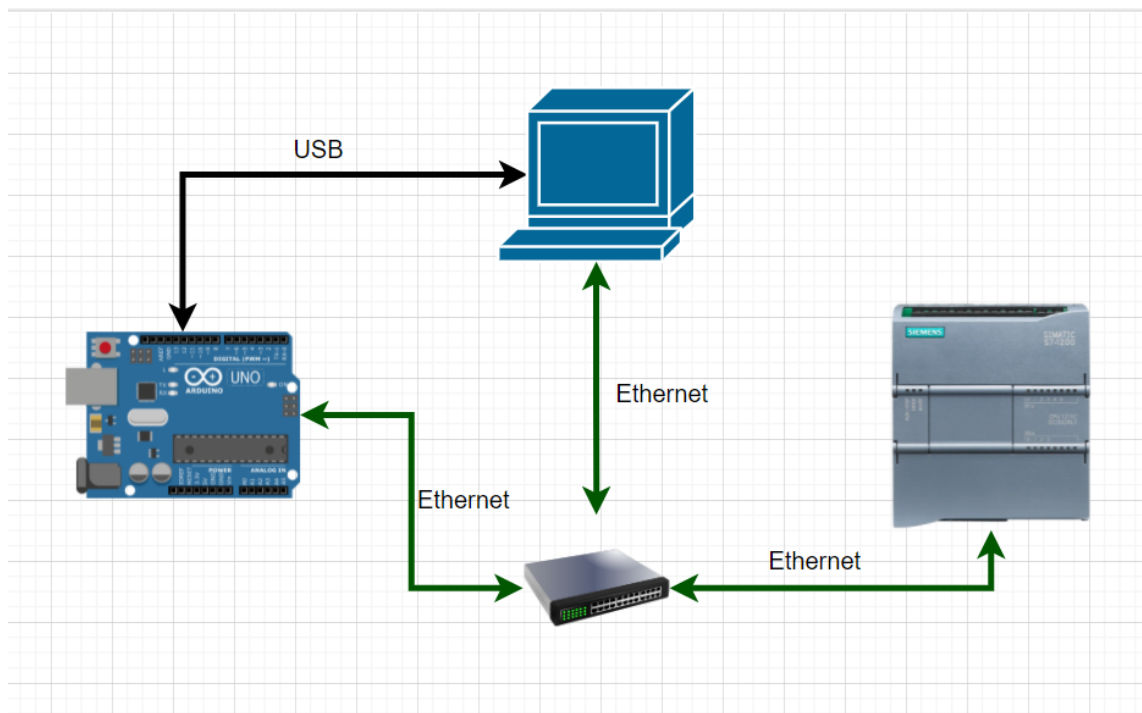
See on komponendid, mis on vaja



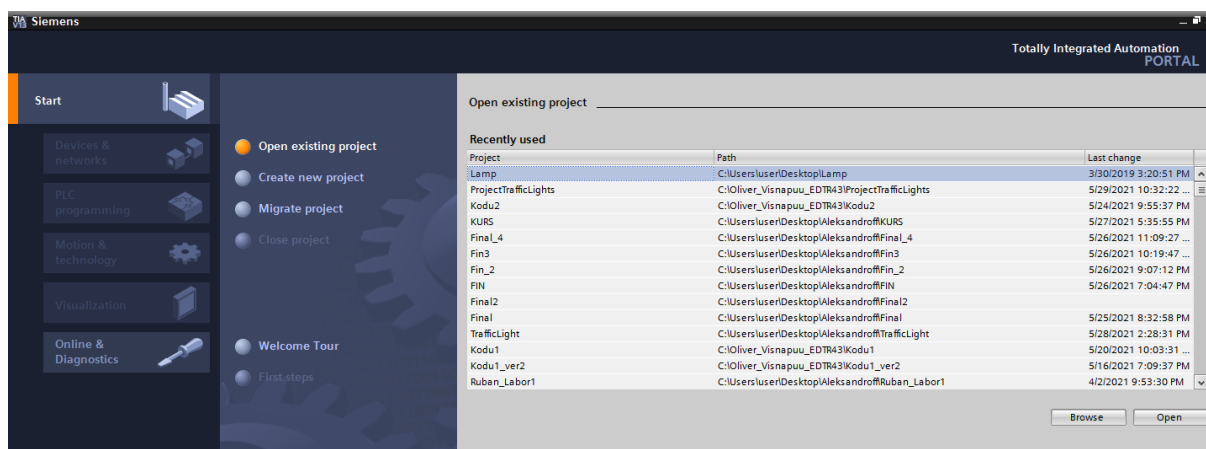
Joonis 1 Komponentid, mis on vaja

- On vaja ühendada Arduino ja Ethernet Shield
- Ühendada Ethernet Shield ja Switch (kasutada kabeel Ethernet)
- Edasi on vaja ühendada arvuti ja Switch, kasutades kabeel Ethernet
- Ühendada Arduino ja arvuti USB-kaabli kaudu
- Ethernet Ühendada Siemens PLC ja Switch, kasutades Ethernet kabeel

Samuti on olemas ühendamineskeem, kus on võimalus kontrollida ühendamine.

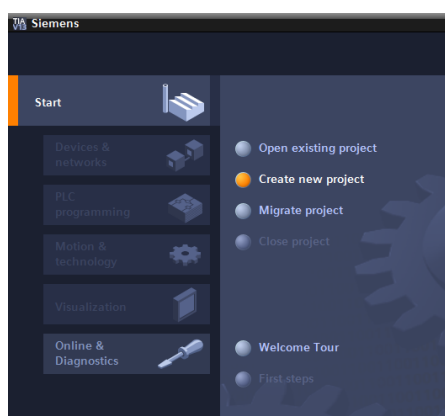


Nüüd saate jätkata kontrolleri konfigureerimist. Selleks avage Tia portaal V15.1



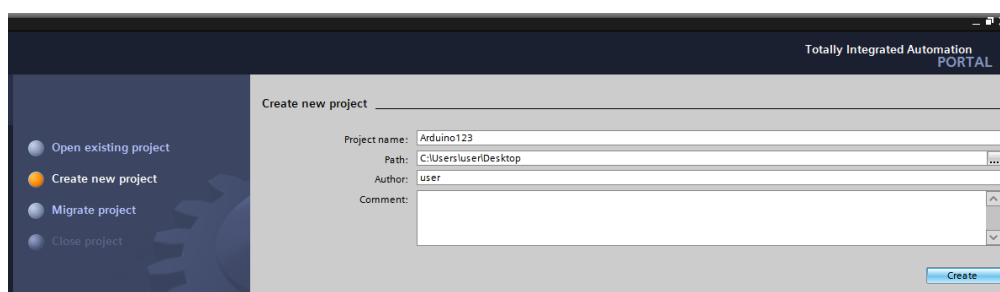
Joonis 2 Avaleht TIA Portaalis

Vajuta nupp «Create new project»



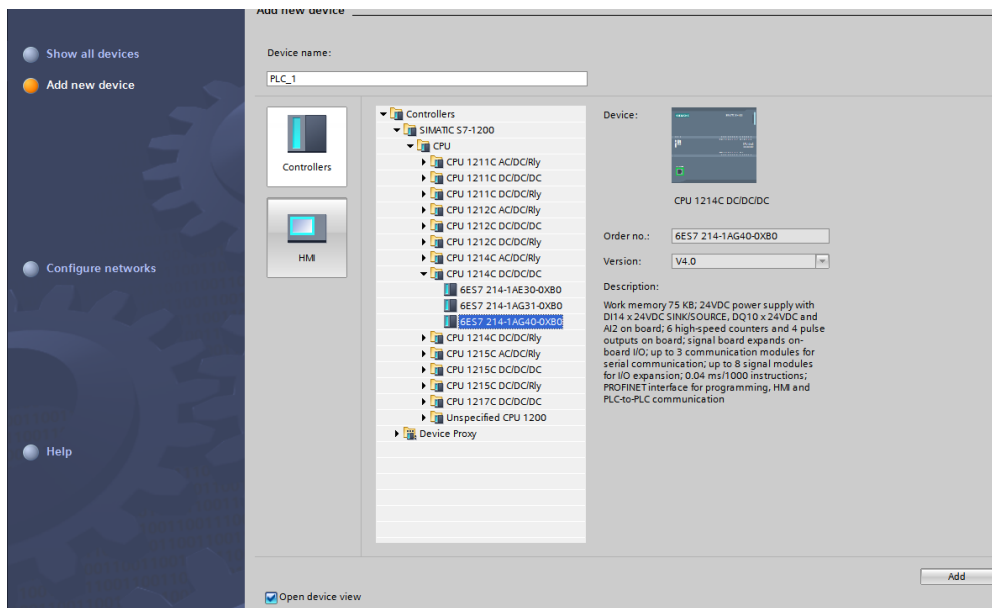
Joonis 3 Uue projekti loomine

Edasi on vaja luua uue projekti ja anda nimi



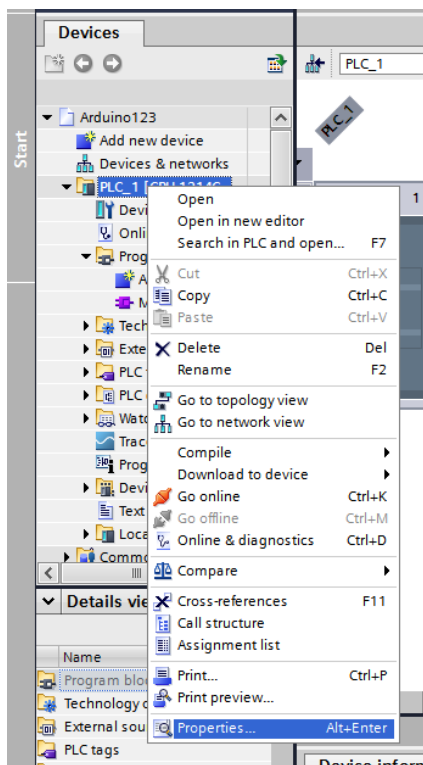
Joonis 4 Projekti nimi

Vajuta nupp «Devices & networks», edasi «Add new device» ja valida kontrolleri, mis te kasutate (kontrolleri teave on loetletud kontrolleri küljel) – vajutama «Add»



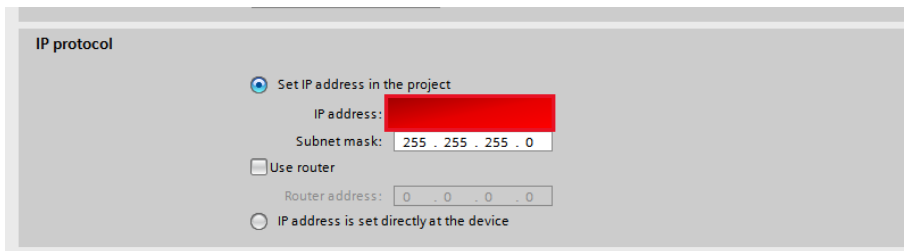
Joonis 5 PLC lisamine

Paremklõpsake PLC_1 ja valida «Properties»



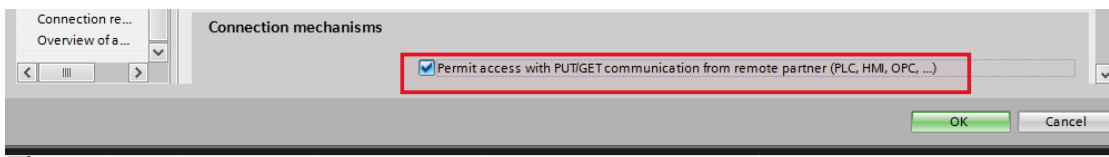
Joonis 6 Seadistamine

Siin on vaja leida «PROFINET interface» – «Ethernet addresses» edasi on vaja vahetada IP address (IP märgitud kontrolleri)



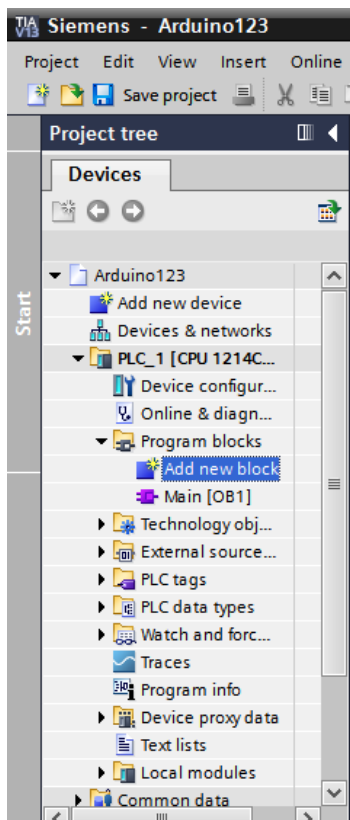
Joonis 7 IP-adressi lisamine

Edasi on vaja avatada «Protection» ja pane linnuke, seeläbi võimaldada funktsiooni PUT/GET



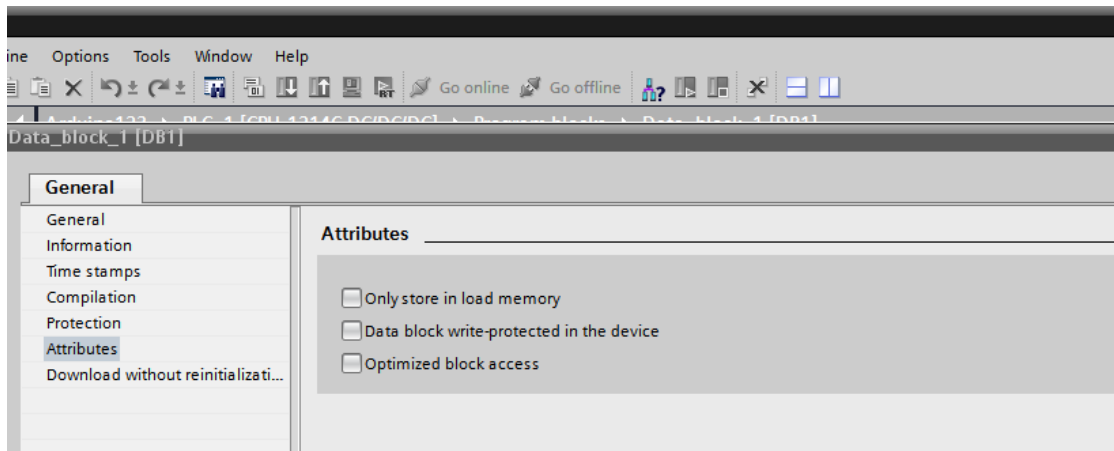
Joonis 8 PUT/GET funktsioon

Siis saate sulgeda «Properties» ja nuud on vaja lisada «Data_Block». Selleks klõpsake nuppu «Program blocks» – «Add new blocks» – «Data Block»



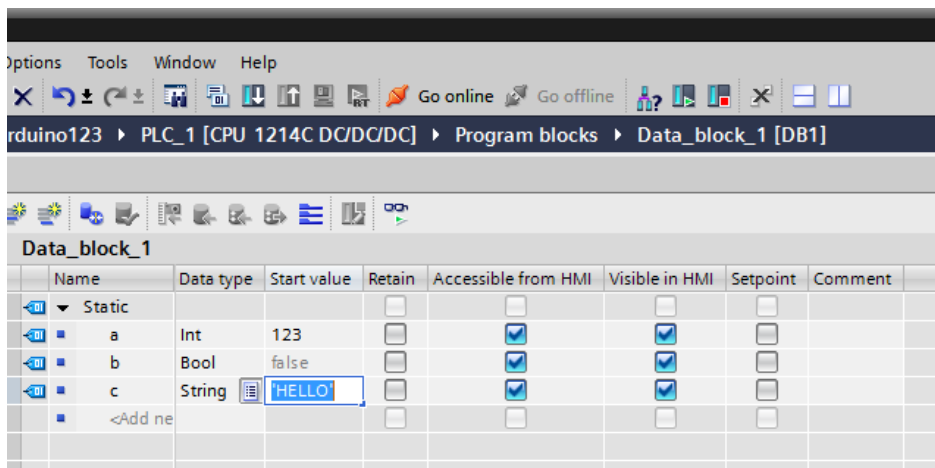
Joonis 9 Data_Block lisamine

Paremklõpsake «Data_Block», mis on vasakpoolses veerus ja valida «Properties». Edasi «Attributes» tühista linnike «Optimized block access» (see on vaja adresseerimiseks) ja klõpsake «OK»



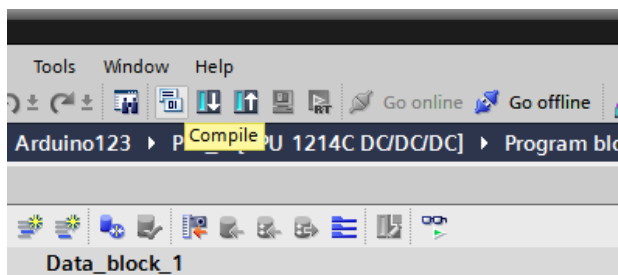
Joonis 10 «Optimized block access»

Lisage siltid



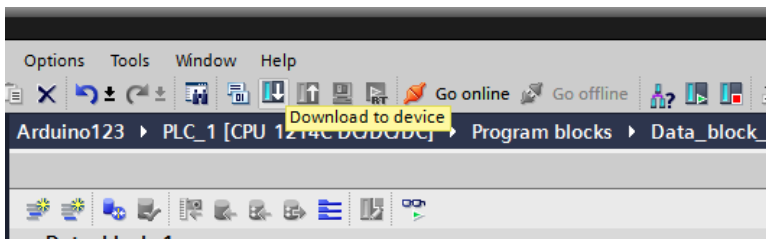
Joonis 11 Lisamine siltid

klõpsake nuppu «Compile»



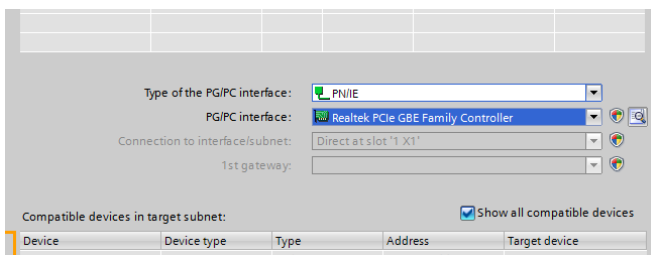
Joonis 12 «Compile»

Klõpsake «Download to device»



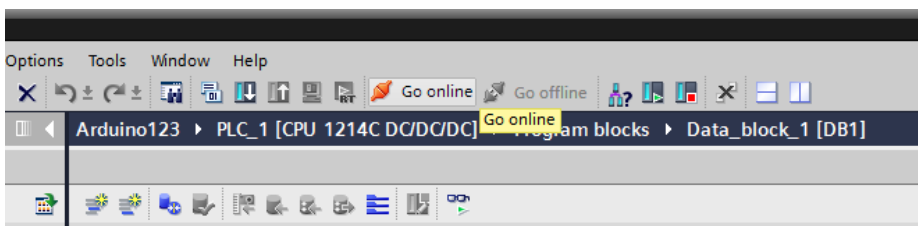
Joonis 13 «Download to device»

Siin on vaja tähistada (PN/IE) ja liides (Realtek PCIe GBE Family Controller). Edasi klõpsage «Start search» - «Load»



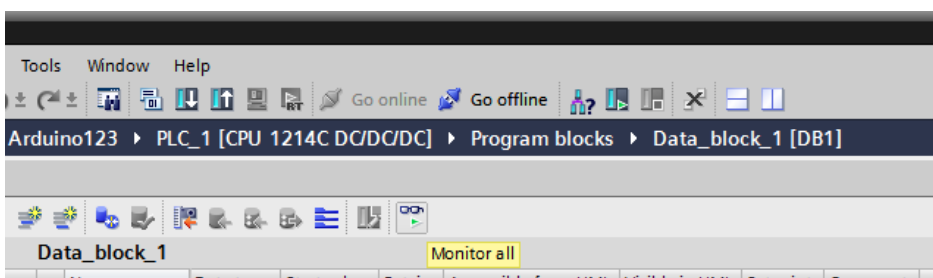
Joonis 14 PLC leidmine

Alustamiseks on vaja vajutada «Go online»



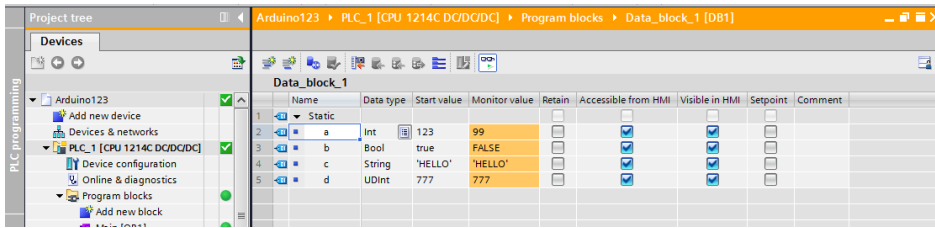
Joonis 15 «Go online»

Ja «Monitor all»



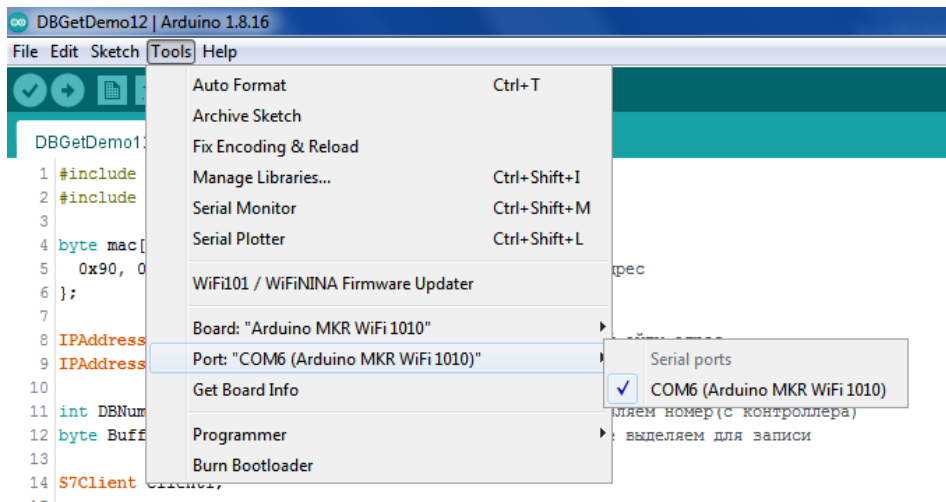
Joonis 16 «Monitor all»

Praegu programm TIA Portaal is töötab



Joonis 17 Data_Block TIA Portaalis

Edasi on vaja avatada Arduino IDE ja määrake port, millega ühendatud Arduino. Nüüd on vaja kirjutada koodi



Joonis 18 Pordi mäрге, millega Arduino ühendatud

Lisamine teigid

```
1. #include <Platform.h>
2. #include "Settimino.h"
```

Määrake MAC-aadress

```
1. byte mac[] = {
2.   0x90, 0xA2, 0xDA, 0x0F, 0x08, 0xE11
3. };
```

Järgmisena peate määrama kohaliku IP-aadressi ja kontrolleri IP-aadressi (mis määrati Tia portaalis)

```
1. IPAddress Local(192, 168, 6, 135); // on vaja kirjutada kohalik IP-aadress
2. IPAddress PLC(192, 168, 6, 134); // Mikrokontrolleri IP-aadress
```

Lisage silt Data_Block ja lisage number (kontrollerist) (antud juhul DB1)

```
1. int DBNum = 1;
```

Märgime salvestamiseks eraldatava mälumahu

```
1. byte Buffer[512];
```

Loome S7Client klassi objekti

```
1. S7Client Client1;
2.
3. unsigned long Elapsed;
4. void setup() {
5.   Serial.begin(9600); // uuenduse kiirust
6.   while(!Serial){}
```

Etherneti ühendamise

```
1. EthernetInit(mac, Local); // interenetiühendus
2. delay(2000); // peetus
3. Serial.println("");
4. Serial.println("Cable connected"); // info väljunud
5. Serial.print("Local IP address : ");
6. Serial.println(Ethernet.localIP());
7.
8. }
```

Registreerime ühenduse kontrolleringa

```
1. bool Connect()
2. {
3.   int Result = Client1.ConnectTo(PLC,
4.                                   0, // Rack
5.                                   1); // Slot
6.   Serial.print("Connecting to "); Serial.println(PLC); // ühendamise PLC
7.   if (Result == 0)
8.   {
9.     Serial.print("Connected ! PDU Length = "); Serial.println(Client1.GetPDULength());
10.  } // edukas ühendus
11.  else
12.    Serial.println("Connection error");
13.  return Result == 0; // ühenduse viga
14. }
```

Veakontrolli lisamine

```
1. void CheckError(int ErrNo)
2. {
3.   Serial.print("Error No. 0x");
4.   Serial.println(ErrNo, HEX); //viga number
5.
6.   if (ErrNo & 0x00FF) // kui viga on tõsine, lülitage välja
7.   {
8.     Serial.println("SEVERE ERROR, disconnecting.");
9.     Client1.Disconnect();
10.  }
11. }
```

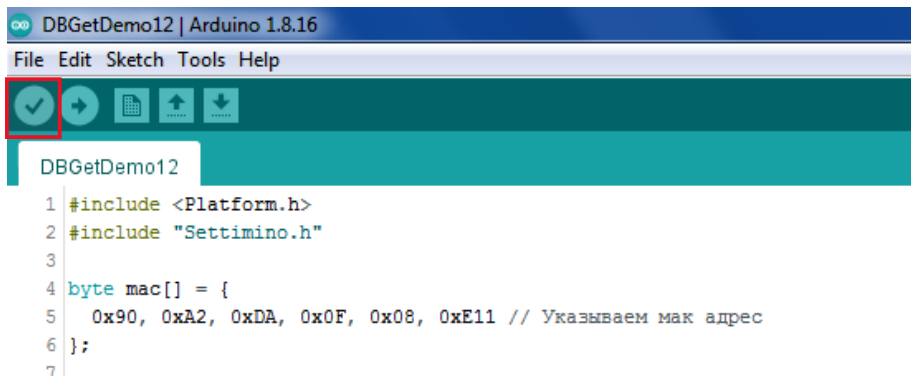
Siltid lisamine ja andmete väljastamine

```

1. void loop()
2. {
3.
4.     uint16_t Size;
5.     int Result;
6.     Size = sizeof(Buffer);
7.     void *userdata = NULL;
8.     // Connection
9.     while (!Client1.Connected)
10.    {
11.        if (!Connect())
12.            delay(500);
13.    }
14.
15.    Serial.print("Uploading DB"); Serial.println(DBNum);
16.    MarkTime();
17.    byte adadw = 0x0;
18.    Result = Client1.ReadArea(0x81, 1, 0, 1, &adadw);
19.
20.    Serial.print("my byte");
21.    Serial.println(adadw);
22.    byte mydb[512];
23.
24.
25.    Result = Client1.ReadArea(0x84, 1, 0, 260, &mydb); // Siin on vaja kirjutada andmeala,
    DB number, start, kogus ja mälu
26.    Serial.print("my byte arr is:");
27.
28.    int a = S7.IntegerAt(&mydb, 0); // Lisamine silt
29.    bool b = S7.BitAt(&mydb, 2, 0);
30.    String c = S7.StringAt(&mydb, 4);
31.    Serial.print("a= "); // andmeväljund
32.    Serial.println(a);
33.    Serial.print("b= ");
34.    Serial.println(b);
35.    Serial.print("c= ");
36.    Serial.println(c);
37.    Serial.println("----- ");
38.    for (int i = 0; i < 2; i++) {
39.        for (int j = 0; j < 8; j++) {
40.            Serial.print(S7.BitAt(&mydb, i, j));
41.        }
42.    } // massivväljund (kahendsüsteem)
43.
44.    Serial.println("----- ");
45.    Serial.println("-----");
46.
47.    delay(5000); // peetus
48. }

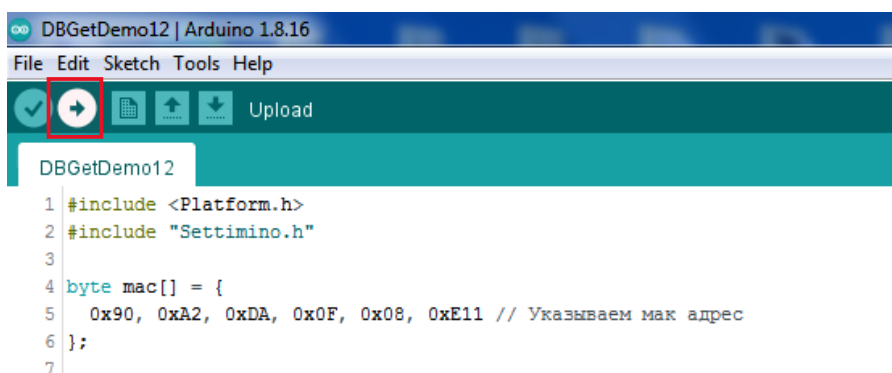
```

Pärast koodi kirjutamist peate klõpsama vasakus ülanurgas olevat linnukest. See kontrollib koodi süntaksit.



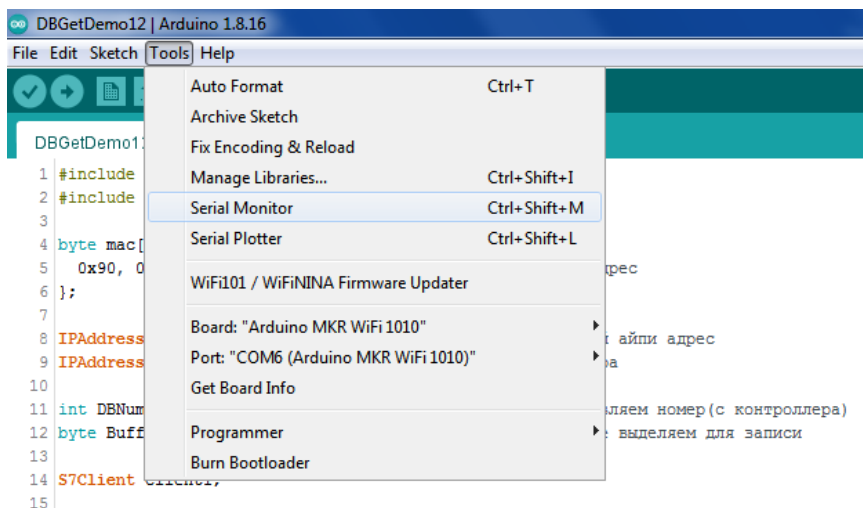
Joonis 19 Koodi süntaksis kontrollomine

Järgmisena laadige kood, klõpsates linnukesele



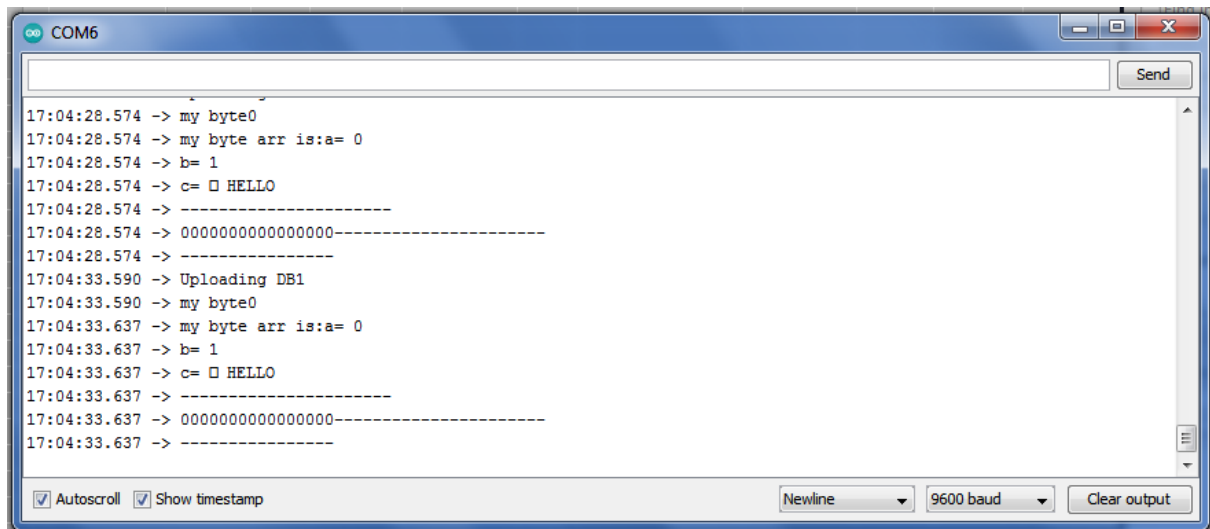
Joonis 20 Koodi laadimine

Pärast edukat allalaadimist, avage «Tools» – «Serial Monitor» (või Ctrl+Shift+M)



Joonis 21 "Serial Monitor"

Ilmub andmeväljund



Joonis 22 Andmeväljund