



**UNIVERSITY
OF OULU**

TIETO- JA SÄHKÖTEKNIIKAN TIEDEKUNTA

**Eliud Keino
Jonne Taipale
Joonas Päivärinta**

**Puristusvoima-anturi osana robottitarttujan
toimintaa**

Kandidaatintyö
Tietotekniikan tutkinto-ohjelma
Toukokuu 2019

Keino E., Taipale J., Päivärinta J. (2019) Puristusvoima-anturi osana robottitarttujan toimintaa. Oulun yliopisto, tietotekniikan tutkinto-ohjelma. Kandidaatintyö, 43 s.

TIIVISTELMÄ

Robotti on jo noin 80 vuotta vanha käsite, mutta nykyistä muotoaan robotiikka on alkanut saavuttaa vasta 1900-luvun loppupuolella teollisuuden myötä. Robotiikka onkin nopeassa kasvussa oleva teknologian ala. Sille pyritään jatkuvasti etsimään uusia sovelluskohteita ja se on korvannut ihmisiä jo monilla aloilla, kuten teollisuuden kokoonpanolinjoilla. Lisäksi robotiikka on tullut osaksi lääketiedettä ja palvelualoja, ja autonomiset autot tulevat yleistymään liikenteessä lähivuosina. Robottien yleistyessä on otettava huomioon myös turvallisuusnäkökulma, jotta niiden kanssa toimiminen olisi ihmisille luontevaa ja riskitöntä. Kun robotit kykenevät paremmin havainnoimaan ympäristöään esimerkiksi erilaisten sensorien avulla, tulee niiden käytöstä turvallisempaa.

Tässä työssä esitellään puristusvoima-anturin käyttöä osana robottitarttujaa, jolla pyritään tarttumaan erilaisiin kappaleisiin. Työssä käytetyn robottitarttujan toiminta perustuu servomoottorin ja anturin yhteistoimintaan, jota ohjataan Arduino Uno -kehitysalustan avulla. Työssä tultiin tulokseen, että puristusvoima-anturi on hyödyllinen apuväline robottitarttujan toiminnassa. Tämä korostuu varsinkin silloin, kun tarttujan välissä on kova periksiantamaton kappale, mikä voi johtaa servon tai laitteiston rikkoontumiseen. Työssä käytetty anturi todettiin kuitenkin liian pieneksi, sillä se ei tunnista kappaleita, jotka ovat hieman suurikokoisempia ja joissa on tasainen pinta.

Avainsanat: robotti, tarttuja, FSR, Arduino, servo

Keino E., Taipale J., Päivärinta J. (2019) Force sensing resistor as a part of robot gripper. University of Oulu, Degree Programme in Computer Science and Engineering. Bachelor's Thesis, 43 p.

ABSTRACT

As a concept, robot is about 80 years old already, but the robotics has started to reach its current form in the late 20th century by becoming common with industry. The robotics is a quickly growing field of technology. New applications for robotics are constantly being striven to find, and the robotics has already replaced people in many fields, for example in assembly lines in industry. In addition, the robotics has become a part of medicine and service industry, and autonomous cars will become more general in traffic in the near future. While robots are getting more common, the security perspective of robotics should be also taken into account. This is necessary for the interaction with robots to be natural and risk free for human beings. When robots are able to detect their environment better, for example by using different sensors, it becomes safer to use them.

In this project force sensing resistor is introduced as a part of a robot gripper, which is trying to catch different objects. The action of the robot gripper is based on the cooperation of servomotor and FSR, that are being controlled by Arduino Uno -development board. In conclusion, the FSR is a useful tool for the robot gripper. This appears especially when there is a tough object that won't bend inside the gripper. This can break the servo or the setup. The sensor used in this project appeared to be too small, because it does not recognize objects that are slightly larger and have a flat surface.

Keywords: robot, gripper, FSR, Arduino, servo

SISÄLLYSLUETTELO

TIIVISTELMÄ

ABSTRACT

SISÄLLYSLUETTELO

ALKULAUSE

LYHENTEIDEN JA MERKKIEN SELITYKSET

1.	JOHDANTO.....	7
1.1.	Mikä on robotti?	8
2.	ROBOTIIKAN KEHITYS JA TEKNOLOGIAT	9
2.1.	Robottiikan alku ja kehitys.....	9
2.2.	Nykyajan robotiikka	9
2.2.1.	Robottiautot ja mobiilirobotit.....	10
2.2.2.	Robottiikka teollisuudessa.....	13
2.2.3.	Robottiikka terveydenhuollossa	14
2.2.4.	Robottiikka koulutuksessa.....	15
2.2.5.	Robottien käytön turvallisuus.....	16
3.	ROBOTITARTTUJAN KOSKETUS JA OHJAUS.....	18
3.1.	Kosketusaisti ja kosketussensorit	18
3.2.	Laitteisto	19
3.2.1.	Arduino.....	19
3.2.2.	Puristusvoima-anturi	20
3.2.3.	Servomootorit.....	22
3.3.	ROS	25
3.4.	Anturi-servo -yhdistelmä.....	25
4.	LAITTEISTON TESTAUS JA TULOKSET.....	29
4.1.	Testaus.....	30
4.2.	Testauksen yhteenveto ja tulokset.....	33
4.3.	Jatkokehitys	34
5.	PROJEKTIN KUVAUS	35
6.	YHTEENVETO	36
7.	LÄHTEET	37
8.	LIITTEET.....	40

ALKULAUSE

Kandidaatintyö tehtiin Oulun Yliopiston tietotekniikan osastolle kurssilla 521275A - Sulautettujen ohjelmistojen projekti. Kiitokset Teemu Tokolalle ja Juha Röningille, jotka toimivat työn ohjaajana ja tarkastajana. Kiitokset myös Niklas Saarelle, joka oli valmis antamaan neuvoja aina ongelmatilanteissa. Suurin osa työn toteutuksessa käytetyistä laitteista saatiin lainaksi yliopiston tietotekniikan osastolta sekä Fab Lab -työskentelylaboratoriosta.

Oulu, toukokuu 2019

Eliud Keino
Jonne Taipale
Joonas Päivärinta

LYHENTEIDEN JA MERKKIEN SELITYKSET

LED	valoa säteilevä puolijohdekomponentti (Light-Emitting Diode)
LiDAR	optinen tutka (Light Detection and Ranging)
RADAR	radioaaltoihin perustuva tutka (Radio Detection and Ranging)
GPS	satelliittipaikannusjärjestelmä (Global Positioning System)
USB	sarjaväyläarkkitehtuuri (Universal Serial Bus)
I/O	siirräntä (Input/Output)
SRAM	staattinen RAM-muisti (Static Random Access Memory)
FSR	puristusvoima-anturi (Force Sensing Resistor)
ROS	robotiikan käyttöjärjestelmä (Robot Operating System)

1. JOHDANTO

Lähes kaikille meille syntyy edes jonkinlainen mielikuva sanoista robotti ja robotiikka. Olemme nähneet tieteiselokuvia tai vastaavaa materiaalia, joista on käynyt ilmi robottien olemus tai muu niille ominainen toiminta. Robotiikka ja tekoäly herättävät ihmisissä usein tunteita. Jotkut pelkäävät koneen syrjäyttävän ihmisen - ovathan robotit korvanneet ihmisiä työläisinä jo monilla aloilla, kuten tehtaiden kokoonpanolinjoilla ja varastotöissä, mutta onko tämä syrjäyttäminen kuitenkaan todellisuutta nyt tai tulevaisuudessa? Onko olemassa esimerkiksi aloja, joilla robotiikka ja tekoäly ei riitä korvaamaan ihmisen paikkaa, tai muodostuuko esteeksi muita tekijöitä, kuten robotiikan kehittämisen kalleus tai esimerkiksi eettiset tekijät?

Nämä metallin ja älyn yhdistelmät jäljittelevät ihmisen toimintaa tavalla tai toisella. Ne suorittavat esimerkiksi tiettyjä niille annettuja yksittäisiä tehtäviä ilman sen suurempaa toiminnallisuutta. Jotkut voivat mieltää robotit kuvan 1 ihmismäisiksi olennoiksi, jotka kykenevät samoihin asioihin kuin ihmiset; itsenäiseen ajatteluun, sosiaaliseen ja fyysiseen vuorovaikutukseen ympäristönsä kanssa - vain sillä erotuksella, että energianlähteenä toimii ruoan sijasta pistorasia. Lähempänä jokapäiväistä todellisuutta tämän hetken maailmassa ovat kuitenkin erilaiset teollisuuden kokoonpanorobotit sekä itsenäiset pölynimurit ja ruohonleikkurit, joita näkee jo arjessa helpottamassa useiden kotitalouksien taakkaa.



Kuva 1. "Ihmisrobotti" Sophia (Hanson Robotics).¹

¹ Lähde: Stephen McCarthy/Web Summit, "Sophia the Robot". (CC BY 2.0)

1.1. Mikä on robotti?

Robotti ei siis olekaan käsitteenä niin yksinkertainen kuin nopeasti ajateltuna voisi luulla. Isaac Asimovin vuonna 1942 julkaisemassa teoksessa nimeltä Runaround määriteltiin kolme robotiikan lakia, jotka tunnetaan myös Asimovin lakeina [1, 2]:

1. Robotti ei saa vahingoittaa ihmistä toiminnallaan tai toiminnasta pidättäytymällä.
2. Robotin täytyy totella ihmisen käskyjä silloin, kun ne eivät ole ristiriidassa ensimmäisen lain kanssa.
3. Robotin täytyy suojella omaa olemassaoloaan niin pitkälle kuin se ei ole ristiriidassa ensimmäisen tai toisen lain kanssa.

Nämä lait ovat siis perua ajoilta, jolloin robotiikka oli vielä teknisesti kehittämätöntä, ja ne koskevat lähinnä robotteihin liittyviä eettisiä kysymyksiä ottamatta kantaa itse robotiikassa käytettäviin teknologioihin. Kuitenkin niiden voidaan ajatella toimivan ohjenuorana robotiikan kehitykselle myös tulevaisuudessa.

Käytännössä robotti kuitenkin mielletään laitteeksi, joka voi suorittaa monimutkaisia tehtäviä joko suoraan ihmisen käskyttämänä, osittain ihmisen käskyttämänä, ihmisen valvottavana tai täysin itsenäisesti (autonomisesti) eli tietokoneen ohjaamana. Robotteihin liittyviä elementtejä ovat tietotekniikka ja ohjelmoitavuus, erilaiset anturit ja muu elektroniikka ja laitteisto, joita hyödynnetään halutunlaisen toiminnan saavuttamiseksi.

Robotiikkaa ja tietokoneita ei ole aina kuitenkaan suunniteltu pelkkään tuottoisaan hyötymistarkoitukseen, vaan niitä on ajan saatossa kehitetty myös esimerkiksi peleihin. IBM:n kehittämä Deep Blue oli ensimmäinen shakkitietokone, joka kykeni voittamaan hallitsevan shakin maailmanmestarin Garri Kasparovin helmikuussa 1996 käydyssä ottelussa [3]. Tämä kehitystyö toimii myös inspiraationa tälle tutkimustyölle.

2. ROBOTIIKAN KEHITYS JA TEKNOLOGIAT

Sana robotti juontuu tsekkiläisestä sanasta *robota*, joka tarkoittaa maaorjaa tai työläistä [1]. Varsinaisesti robotiikka on kehittynyt vasta viime vuosikymmeninä, jolloin teollisuudessa on alettu hyödyntämään robotteja tuotannon eri vaiheissa. Teollisuuden automatisointi on osaltaan vienyt robotiikan kehittymistä paljon eteenpäin. Lähivuosina roboteille on myös pyritty keksimään ja tuottamaan uusia käyttökohteita niin lääketieteeseen kuin myös muuallekin palvelu- ja hoitoalalle.

2.1. Robotiikan alku ja kehitys

Robottien historia on verrattain lyhyt. Jollain tasolla robotteja muistuttavia laitteita on peräisin jo vuosituhansien takaa, mutta tarinat niistä perustuvat lähinnä yksittäisiin legendoihin. Robotti käsitteenä on vasta alle sata vuotta vanha, ja 1900-luvun loppupuolella robotiikka on saavuttanut enemmän nykymuotoaan. Alkuaikoina esiin on kuitenkin noussut joitain yksittäisiä keksintöjä, joita voidaan pitää robotiikan alkulähteinä.

Ensimmäisenä tunnettuna robottia jäljittelevänä laitteena pidetään ranskalaisen keksijä Jacques de Vaucansonin 1700-luvulla rakentamaa mekaanista “ankkarobottia”, joka kykeni kertoman mukaan liikkumaan, “syömään” ja liikuttamaan siipiään [4]. Tätä laitetta ei ole kuitenkaan päästy tutkimaan, eikä sen toiminnasta ole tarkempaa tietoa tai varmuutta.

Kun tsekkiläinen näytelmäkirjailija Karel Capek julkaisi vuonna 1920 näytelmänsä R.U.R. (Rossum’s Universal Robots), hän teki tunnetuksi sanan robotti [5]. Näytelmässä aiheena ovat robotit, jotka ovat ihmisen kaltaisia oikeuksineen ja tunteineen. 1940-luvulla Isaac Asimov loi kuuluisat Robotiikan lait, mutta 1900-luvun alkupuolella robotiikan kehitys ei ollut vielä kovin edistynyt johtuen sen aikaisten teknologioiden kehittymättömyydestä.

2.2. Nykyajan robotiikka

Robotiikalle pyritään jatkuvasti etsimään yhä useampia ja monipuolisempia käyttökohteita, ja viime vuosina robotiikkaa on pyritty tuomaan myös muille aloille kuin teollisuuteen ja lääketieteeseen. Roboteille on kehitelty eri sovelluksia esimerkiksi palvelualoille ja ihmisten arkisiksi apuvälineiksi, joissa automatisointia ja robotiikka voidaan tulevaisuudessa käyttää paremmin hyödyksi. Esimerkiksi autoilu on yksi paljon esillä oleva aihe robotiikassa autonomisten autojen tasoitellessa tietään markkinoille ympäri maailmaa.

2.2.1. Robottiautot ja mobiilirobotit

Robottiautolla (kuva 2) tarkoitetaan autoa, joka on kokonaan tietokoneen ohjaama ja johon ei tarvita ihmisen väliintuloa. Automatisoitu auto voi olla osittain tai täysin tietokoneen ohjaama. Auto käsittelee useista antureista tulevaa tietoa ja tekee niiden perusteella kaikki ohjaamista koskevat päätökset, kuten reitin valinnan, ohitukset, ryhmittäytymiset jne.

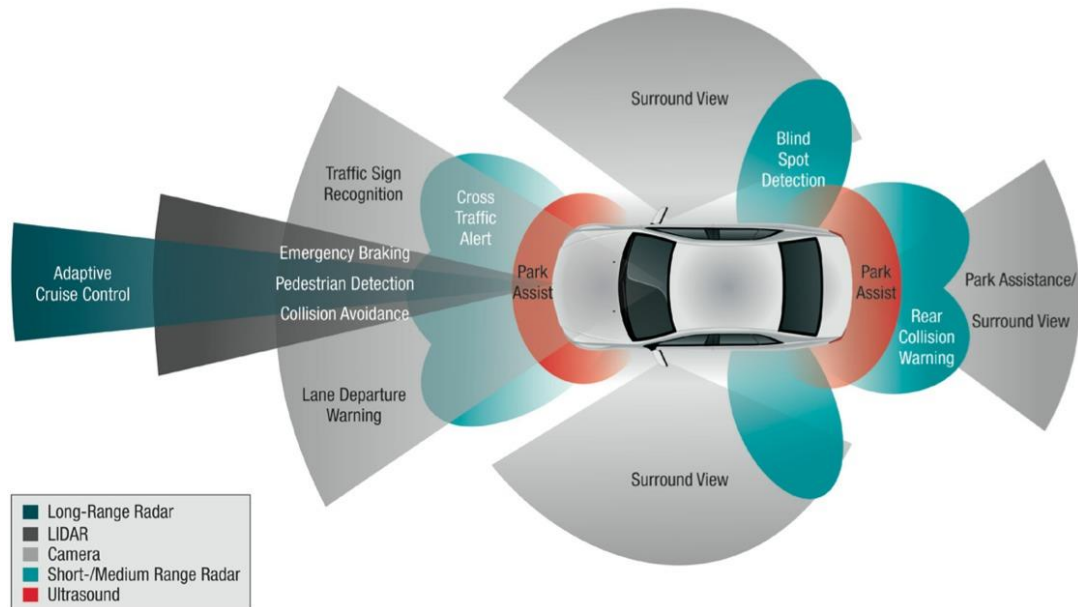


Kuva 2. Robottiauto kaarreajotestissä.²

Robottiautot voisivat muun muassa parantaa liikenteen turvallisuutta tulevaisuudessa, sillä 90% liikenteessä tapahtuvista onnettomuuksista johtuu ihmisen tekemistä virheistä. Robottiautot tuovat kuitenkin liikenteeseen myös riskejä uusien teknologien mukana [6], yhtenä esimerkkinä toimii autoihin liittyvä tietoturva. Robottiautoilla on viisi perusominaisuutta, joiden perusteella ne toimivat:

1. Havaintokyky
 - Radioaaltoihin perustuva tutka (RADAR)
 - Optinen tutka (LiDAR)
 - Tietokonenäkö
2. Paikallistaminen
 - Satelliittipaikannusjärjestelmä (GPS)
 - Merkintälasku
 - Tiekartat
3. Suunnittelu
4. Ohjaaminen
5. Järjestelmänhallinta

² Lähde: Steve Jurvetson, "Google Robocar Racetrack Ride". (CC BY 2.0)



Kuva 3. Sensoriteknologien hyödyntäminen robottiautoissa ympäristön havainnointiin.³

Havaintokyky käyttää useita eri sensoriteknologioita (kuva 3), joita ovat esimerkiksi RADAR ja LiDAR, sekä tietokonenäköä. RADAR-järjestelmällä voidaan mitata radioaaltoilla matkojen etäisyyksiä, suuntaa, nopeutta ja monia muita ominaisuuksia. LiDAR-järjestelmä toimii samalla tavalla kuin RADAR, mutta se lähettää ja vastaanottaa radioaaltojen sijaan valopulsseja. LiDAR kehitettiin alunperin autonomisia autoja varten, mutta se on levinnyt myös moniin muihin aloihin, kuten alueen kartoitukseen [7]. LiDAR:n pienen aallonpituuden ansiosta sillä on mahdollista tunnistaa pienempiä esineitä kuin tutkalla, joka käyttää radioaaltoja objektien havaitsemiseen. Sillä on myös mahdollista rakentaa tarkkoja 3D-kuvia objekteista. LiDAR-teknologiaa ja laserskannausta on myös käytetty ratojen virtualisointiin ajosimulaattoreita varten.

Paikallistaminen käyttää hyväksi tekniikoita kuten GPS, merkintälasku ja tiekartat. Näin itsestään ohjautuvat autot suunnittelevat reittinsä ja ohjaavat autoa käyttäen järjestelmän havaintokykyä ja paikallistamista. Järjestelmänhallinnan tehtävänä on valvoa autonomista järjestelmää esimerkiksi erilaisten virheiden varalta [8].

Robottiautoilun yleistymiseen tarvitaan lakimuutoksia, jotta se olisi sallittua tieliikenteessä. Suomessa robottiautot eivät ole vielä laillisia tieliikenteessä ilman kokeilulupaa, mutta esimerkiksi Yhdysvalloissa Nevadassa, Floridassa, Kaliforniassa ja Michiganissa toiminta on laillistettua kuljettajan ollessa autossa mukana valmiina ottamaan ohjat käsiinsä. Singaporessa ilman kuljettajaa kulkevat robottiautot ovat jo laillisia. Vaikka täysin autonomisten autojen yleistymisprosessi ottaa vielä aikansa, osittainen itseohjautuminen on ollut käytössä tavallisissa kuluttaja-autoissa jo jonkin aikaa. Autot ovat esimerkiksi voineet asettaa itse vakionopeuden edellä ajavan auton

³ Kuva julkaistu oikeudenhaltijan luvalla (William Buller, Michigan Tech Research Institute).

nopeuden mukaan, tai autoissa voi olla itsestään kytkeytyvä hätäjarrutusjärjestelmä ilman kuskin toimia. Taulukossa 1 autonomisille autoille määritellään kuusi tasoa niiden toiminnan mukaan [9]:

Taulukko 1. Autonomisen auton toiminnan tasot.

Taso 0.	Kuljettaja suorittaa kaikki toiminnot itse eli ajaa, ohjaa, jarruttaa jne. Auto ei puutu ajamiseen millään tavalla, eikä sillä ole välttämättä teknologisia edellytyksiäkään hallita autoa.
Taso 1.	Kuljettajan apuna on joitain avustinjärjestelmiä kuten kaistavahti, "kuolleen kulman" varoitusjärjestelmä ja mäkilähtöavustin.
Taso 2.	Auto hoitaa joitain tehtäviä kuljettajan puolesta, esim. pysäköinti- ja kaistallapitoavustimen muodossa. (Tämän hetken tilanne useimmilla valmistajilla kuluttaja-autoissa.)
Taso 3.	Huomattavan automatisoidun ajamisen taso. Auto kykenee omatoimisesti mm. vaihtamaan kaistaa, mutta kuljettaja voi ottaa halutessaan ohjat.
Taso 4.	Täysautomaattinen taso. Auto hoitaa kaikki ajamisen toiminnot, mutta antaa ne ihmisen haltuun vain sellaisessa tilanteessa, missä se ei itse enää hallitse tilannetta.
Taso 5.	Robottiautoilun kehityksen päämäärä. Ihmisestä tulee matkustaja ja auto/järjestelmä tarvitsee vain päämäärän ja luvan lähteä liikkeelle.

Autonomisten autojen yleistyminen voi tuoda tullessaan monenlaisia hyötyjä. Ehkä tärkeimpänä hyötynä voidaan pitää oletettavaa liikenneonnettomuuksien määrän vähentymistä; robottiautot eliminoivat inhimilliset virheet kuten pitkät reaktioajat ja huolimattoman ajotyylin. Myös liikenne voisi sujuvoitua ja nopeutua autojen kommunikoidessa ympäristönsä kanssa ja näin ollen esimerkiksi nopeusrajoituksia voitaisiin nostaa - ainakin teoriassa. Pienemmät onnettomuustilastot vaikuttaisivat todennäköisesti välillisesti myös autoilun kulujen pienenemiseen matalampien vakuutushintojen muodossa, mutta myös autojen vakuutusjärjestelmä kokee varmasti jonkinlaisen muutoksen uuden tilanteen edessä.

Tässä asiassa ei ole kuitenkaan hyötyjä ilman haittoja tai muita riskejä. Kuten robotiikka yleensäkin, robottiautot vievät ihmisiltä työpaikkoja, tässä tapauksessa esimerkiksi bussi- ja taksikuskeilta. Robottiautoista täytyy ottaa huomioon tosiasia, että ne ovat myös koneita, eli voiko niihin täysin luottaa? Jos järjestelmä vikaantuu tai joutuu tietoturvahyökkäyksen kohteeksi, seuraukset voivat olla arvaamattomat ja tuhoisat. Autojen täytyisi myös kyetä selviämään ääriolosuhteissa, kuten pohjoisen kylmässä talvessa, eivätkä autojen sensorit välttämättä pysty keräämään ohjaamiseen tarvittavaa dataa kovassa lumipyryssä.

2010-luvulla yleistyneiden mobiilirobottien eli esimerkiksi itsenäisesti liikkuvien imurointi- ja ruohonleikkuurobottien navigointi perustuu monesti jonkinlaisen kartta-avusteiseen liikkumiseen. Robotit voivat luoda itse kartan liikkessaan ja kohdatessaan esteitä, jolloin niiden liikkuminen jatkossa helpottuu. Tällainen kartan luontiin perustuva navigointi perustuu robotin itse tekemiin havaintoihin ympäristöstään ja sen perusteella tallennettuun informaatioon liikkumisalueestaan. Kartattomassa navigoinnissa robotti ei tarvitse etukäteen tietoa ympäristöstään kyetäkseen liikkumaan. Liikkuminen perustuu siis suoraan ympäristön tarkkailuun ja havaintoihin esteistä kuten seinät, ovet, pöydät ym. [10].

2.2.2. Robotiikka teollisuudessa

Robotiikasta on teollisuuden alalla runsaasti apua. Se on työkalu, jonka avulla saadaan tehtyä nopeasti ja tehokkaasti suuria tuotantoeriä tasaisella laadulla. Teollisuusroboteille (kuva 4) ominaista on niiden uudelleenohjelmoitavuus ja helppous tehdä muutoksia, jotta robottia voidaan hyödyntää esimerkiksi jossain toisessa työvaiheessa. Tämän lisäksi roboteille voidaan antaa ihmisille liian vaarallisia, esimerkiksi tuottamiseen liittyviä tehtäviä [11], tai sotilaallisia tehtäviä esimerkiksi räjähteiden etsintään tai laukaisemiseen.



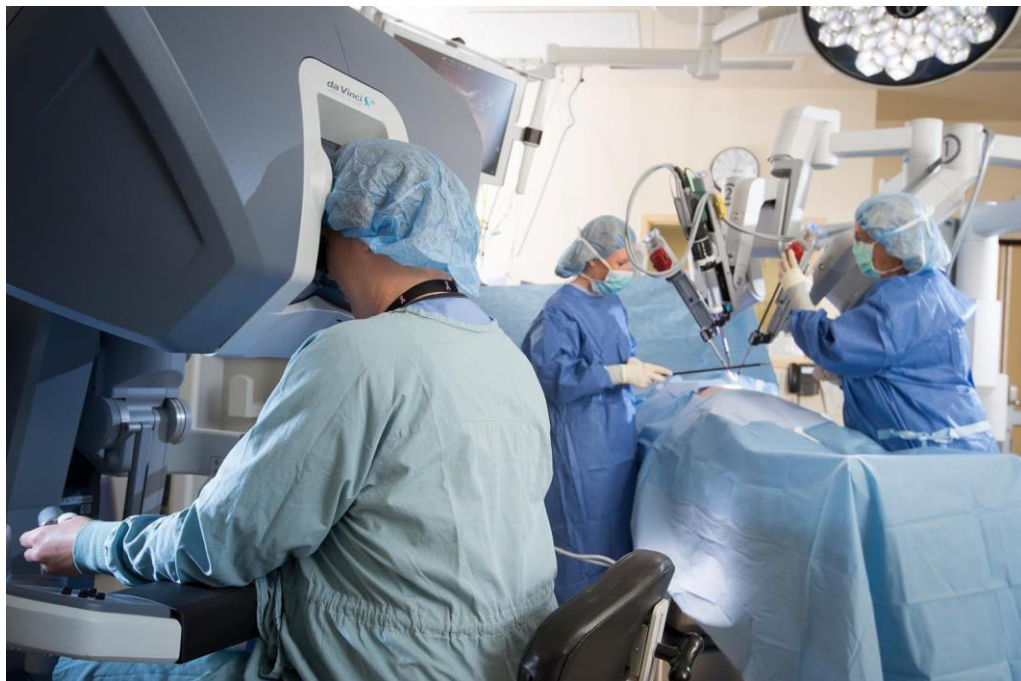
Kuva 4. Teollisuusrobotit työssään auton kokoonpanolinjalla.⁴

⁴ Lähde: Wikimedia Commons, "Industrial robots". (CC BY-SA 4.0)

Robottiikan kehitys sai suuremman sysäyksen 1970-luvulla amerikkalaisen Unimationin tuodessa markkinoille hydraulisen ja tietokoneohjatun teollisuusrobotin. Teollisuusrobotit ovat olleet hyvin vakiintuneita tuotannon alalla siis jo yli neljäkymmentä vuotta. Niiden eri tehtäviin kuuluvat muun muassa pinoaminen, valanta, maalaus, lajittelu, hitsaus, komponenttien juottaminen ja useat muut tehtävät [12]. Vaikka tuotantolinjat vaikuttavat olevan kokonaan automatisoituja, todellisuudessa teollisuusrobotteja kuitenkin täytyy valvoa, ohjata ja huoltaa. Tällaisia robotteja kutsutaan “teleroboteiksi” [13].

2.2.3. Robottiikka terveydenhuollossa

Lääketiede ja terveydenhoito ovat myös hyödyntäneet robottiikkaa jo useamman vuosikymmenen. Esimerkiksi kirurgiassa robottikäsi on paljon tarkempi ja vakaampi kuin yksikään ihmiskäsi. Robotin päämääränä ei ole kuitenkaan kirurgin korvaaminen, vaan robotti toimii kirurgin apuvälineenä antamalla tälle monipuolisia työkaluja työntekoon [14].



Kuva 5. DaVinci-robotti leikkaustyössä.⁵

Robottiikan kaupallinen ja tutkimustyön kiinnostus lääketieteen alalla on lisääntynyt huomattavasti viime vuosikymmenellä. Robotit ovat alkaneet jo vaikuttaa lääketieteen ja terveydenhoidon kehittymiseen. Telerobottiikkaa käytetään kirurgisiin toimenpiteisiin, mikä näkyy lyhyempinä toipumisaikoina ja luotettavampina tuloksina tietyissä leikkauksissa [15]. Tämä voi johtua osaltaan siitä, että robottikäsi

⁵ Lähde: Wikimedia Commons (Franciscan Health), “Robot assisted surgery”. (CC BY-SA 4.0)

ei tärise. Esimerkiksi vuonna 1999 Euroopassa saataville tullut, vuonna 2000 USA:ssa FDA:n (Yhdysvaltain elintarvike- ja lääkevirasto) hyväksynnän saanut DaVinci on robotti, joka on kehitetty kirurgisia toimenpiteitä varten (ks. kuva 5). Kyseisellä robotilla oli aluksi kaksi kättä ja kamera. Jälkeenpäin robottiin on tehty parannuksia, kuten lisätty kolmas käsi ja parannettu kuvanlaatua. DaVinci-robotista on tullut yksi kaikista käytetyimmistä ja arvostetuimmista järjestelmistä robotiikkakirurgian ajanjaksolla [16]. Robotin yleisyyttä kuitenkin rajoittaa ainakin jossain määrin sen lähes kahden miljoonan euron hinta. Markkinoille on kuitenkin tullut vaihtoehtoisia ja selvästi halvempia robotteja pienempiä leikkauksia varten.

2.2.4. Robotiikka koulutuksessa

Robotiikkaa on sovellettu myös koulutukseen. On havaittu, että nuoret lapset suoriutuvat paremmin annetuista tehtävistä ja ovat kiinnostuneempia, kun oppiminen tapahtuu robotin avustuksella [17]. Myös tuoreempi tutkimus on vahvistanut tämän väitteen. Baxterin ja kumppaneiden tekemässä tutkimuksessa tutkitaan robotin vaikutusta oppimiseen. Tutkimuksessa tullaan tulokseen, että yleisesti ottaen robotti on edesauttanut oppimista tekemällä siitä enemmän yksilöllisen tapahtuman [18]. Vaikka robotiikka voi olla hyvä väline oppimisen tehostamiseen, voidaanko joskus tulla tilanteeseen, jossa opettaja on kokonaan korvattu robotilla?

Lähitulevaisuudessa tullaan todennäköisemmin hyödyntämään robotiikkaa opetuksen työkaluna ja apuvälineenä, ei niinkään opettajien korvaajana. Natalie Reich-Stiebert ja Friederike Eyssel tutkivat artikkelissaan opettajien suhtautumista koulutusrobotteihin [19]. He haastattelivat Saksassa 59 opettajaa eri koulutusasteilta. Vastoin odotuksia yleinen suhtautuminen robottien käyttöön työkaluna oli myönteinen etenkin tekniikan ja matematiikan saralla. Opettajat uskoivat robottien mm. olevan hyödyllisiä esimerkiksi pienemmissä ryhmissä ja auttavan yksilöllisissä oppimismuodoissa. Toisaalta epävarmuutta herätti robottien kyky tulkita sosiaalisia tilanteita ja reagoida niihin asianmukaisesti. Vastaavasti Koreassa tehdyssä tutkimuksessa on haastateltu 140 peruskoulun opettajaa, joiden yleinen asenne opettajarobotteja kohtaan on kielteisempi kuin esimerkiksi itse oppilaiden tai heidän vanhempiansa [20].

Eunil Park, Ki Joon Kim ja Angel P.del Pobil tutkivat Soulin yliopistossa opiskelijoiden suhtautumista opettajaan, kun opettajana toimi joko ihminen tai robotti [21]. Tutkimukseen osallistui 126 opiskelijaa, jotka olivat iältään 19-33 vuotiaita. Kyselyn tuloksena opiskelijat ottivat myönteisemmin vastaan ja kuuntelivat mieluummin ihmisopettajan antamia palautteita kuin robotin antamia. Robotti saattoi aiheuttaa myös kiusallisia tilanteita, kun se antoi opiskelijoille palautetta esimerkiksi kokeiden tuloksista. Kaiken kaikkiaan robotti koettiin myös hieman outona vaihtoehtona opettajan tehtäviin.

Opetus on siis kuitenkin tärkeä robotiikan kehityskohde, mutta tälläkin saralla on vielä paljon työtä tehtävänä. Myös hinta on tässäkin tapauksessa yksi elementti,

jota ei voida vain täysin sivuuttaa; robottien hinnat ovat korkeita, joten koulutuslaitoksilla täytyy olla myös taloudellisia resursseja robottien hankkimiseen ja käyttöönottoon.

2.2.5. Robottien käytön turvallisuus

Teollisuuden robotit olivat aiemmin isoja ja tehokkaita, eivätkä välttämättä niin luotettavia. Tämän takia niiden käyttö sisälsi paljon huolenaiheita turvallisuuteen liittyen. Kun robotit olivat toiminnassa, ihmisten täytyi työskennellä niistä täysin erillään mahdollisten vahinkojen varalta. Nykyään kuitenkin tekniikka on kehittynyt niin paljon, että ihmiset pystyvät työskentelemään robotin työalueella ilman isompia riskejä [11]. Vaikka turvallisuuden saralla on tapahtunut paljon kehitystä, robottien toimintaa ei voida vieläkaan pitää täysin riskittömänä ja turvallisena. Huolimatta siitä, että Asimovin lait ovatkin peräisin lähes 80 vuoden takaa, pinnalle on noussut viime aikoinakin lain ensimmäinen kohta: “Robotti ei saa vahingoittaa ihmistä toiminnallaan tai toiminnasta pidättäytymällä”. Tässä kohdassa ei olla kuitenkaan onnistuttu täysin pitäytymään.

Teollisuus on yksi yleinen kuolemantapauksia tuottava kohde varsinkin kehitysmaissa, joissa työturvallisuusasiat eivät ole täysin kunnossa. Kun teollisuusrobotit toimivat linjastoilla, myös turvallisuustekijöiden täytyy olla kunnossa, mutta kuolonuhreilta ei ole viime vuosinakaan täysin välttytty. Vastaava esimerkki löytyy Saksan Kasselista heinäkuulta 2015: 21-vuotias mies oli asentamassa Volkswagenin tehtaalla autojen kokoamiseen käytettävää robottia, kun robotti oli yllättäen iskenyt hänet metallilevyä vasten. Mies kuoli myöhemmin vammoihinsa ja tapaus herätti keskustelua robottien käytön turvallisuudesta. Asiantuntijoiden mukaan tapauksessa oli enemmän kyse “yleisestä” teollisuuden tapaturmasta, joka johtui ihmisten tekemästä virheestä, kuin otsikot keränneistä roboteista [22]. Vaikka näin olisikin, pintaan nousee väkisin kysymys turvallisuustekijöistä robottien suhteen, ja niiden valvonnan tärkeys korostuu entisestään.

Myös liikenne aiheuttaa vuosittain maailmassa useita tuhansia kuolemia. Kun liikenteeseen tuodaan autonomisesti toimivia robottiautoja, myös niiden turvallisuustekijät ovat tarkastelun kohteena. Autonomisen auton ideana on mm. parantaa liikenteen turvallisuutta, sillä siinä poissuljetaan ihmiskuljettajiin liittyvät inhimilliset seikat, kuten keskittymisen herpaantuminen tai väsymyksen aiheuttama huono havainnointi- ja reagoitokyky.

Vuoden 2016 toukokuussa sähköautoja tuottavan Teslan Model S -mallisen auton kuljettaja kuitenkin kuoli Floridassa törmätessään kuorma-autoon. Autossa oli törmäyshetkellä käytössä autopilottijärjestelmä, joka mahdollistaa ajamisen ilman kuljettajan huomiota - auto vaihtaa siis kaistaa ja reagoi ympäröivään liikenteeseen erilaisten sensorien ja kameroiden avulla. Tämä tapaus on tietävästi ensimmäinen ‘robottiauton’ aiheuttama kuolema maailmassa. Teslan mukaan autopilottijärjestelmä

ei ollut havainnut tielle tulleen ajoneuvon valkeana hohtavaa peltiä kirkasta taivasta vasten [23]. Tällaiset tapaukset osaltaan vaikeuttavat robottiautojen yleistymistä liikenteessä, vaikka ne ovatkin laillisia jo muutamissa valtioissa.

Roboteille on siis kehitetty jo monenlaisia sovelluksia; ne osaavat kuljettaa autoa itsenäisesti erilaisten tutka- ja kamerajärjestelmien avulla, toimia tehtaiden kokoonpanolinjoilla tehden tarkkaa ja tasalaatuista työnjälkeä, sekä toimia avustavina käsinä kirurgisissa toimenpiteissä. Roboteilla on ollut kyky tarttua esineisiin jo pitkään, mutta uudempana haasteena on robottikäden kyky arvioida tarttumiseen tarvittavaa puristusvoimaa.

3. ROBOTTITARTTUJAN KOSKETUS JA OHJAUS

Palvelurobotiikan alalla robotilla voi olla kädet, joilla se pystyy kontrolloimaan esineitä ja tarttumaan niihin. Näitä sovelluksia voivat olla esimerkiksi varastoilla käytettävät robotit, joiden on tarkoitus osata paikantaa esine, napata se, ja liikuttaa paikasta A paikkaan B. Myös teollisuusrobotit saattavat olla ikään kuin käsiä, joita liikuttelemalla se pääsee paremmin käsiksi tiettyyn työvaiheeseen todellisessa ympäristössä. Ylipäätään robotiikassa on kiinnitetty paljon huomioita robottien kykyyn käsitellä esineitä sekä niiden muuhun fyysiseen vuorovaikutukseen, jotta saavutettaisiin parempia tuloksia niiden toiminnassa. Robotin kyky tunnistaa esineitä ja tarttua niihin kunnolla on yksi perinteisimmistä ja tutkituimmista roboteille kehitetyistä tehtävistä [24].

3.1. Kosketusaisti ja kosketussensorit

Van den Heever, Schreve ja Scheffer määrittivät tekstissään kosketuksen aistimisen sensorin ja objektin välisen kontaktin parametrien mittaamiseksi. Se on määritelty myös voimien alueellisen jakautumisen havainnoimiseksi ja ennalta määrätyn sensorialueen kohtisuoraksi mittaamiseksi [25].

Aineiston mukaan maailman väestönkasvu on johtanut moniin ongelmiin palvelujen toimittamisesta turvallisuusseikkoihin, ja näin ollen on tutkittu paljon ratkaisuja, joilla toteutetaan ihmiskehoa matkivia humanoideja, jotka käyttävät tekoälyä ympäristönsä ymmärtämiseen. Fritzsche, Elkmann, ja Schulenburg esittelevät keinotekoisien kosketussensorin, joka mahdollistaa turvallisen toimintaympäristön ihmisten ja robottien väliselle vuorovaikutukselle [26]. Kirjoittajat käyttävät erilaisia painesensoreita vaimentamaan robotin rungon iskuja ja havaitsemaan kosketuksia ympäristönsä objektien kanssa.

Jotta ihmisellä ja robotilla voi olla yhteinen työskentelytila, täytyy sovittaa yhteen robottien kyky tuntea ja tunnistaa ympäristö kosketussensoreita käyttäen. Kirjoittajat Van den Heever ym. uskovat, että lähitulevaisuudessa robotit korvaavat ihmisen mm. hyvin riskipitoisissa tehtävissä. Tämän toteutumiseksi roboteilla täytyy olla ihmisen tuntoaisteihin verrattavissa olevaa sensorteekniikkaa, jotta ne toimisivat odotetulla tavalla. Tutkimuksien [25, 26] mukaan kosketussensorien kehityksen esikuvana on ihmisen iho, jolla voidaan tunnistaa siihen kohdistuvaa voimaa, lämpötilaa, kovuutta ja rakennetta.

Tämä tarkastelu on osoittanut, että kosketussensorien kehitykseen liittyy myös haasteita. Fritzschen jne. mukaan sensorien liittäminen osaksi robottien toimintaa vaatii paljon teknistä työtä, kun sillä halutaan saavuttaa tietty toiminnallisuus ja yritetään välttää sensorien vahingoittuminen [26]. Tutkimuksessa pohditaan myös haastetta, onko mahdollista toteuttaa kosketussensori, joka mittaa tarvittavaa voimaa soveltaen sitä robotin tarkoituksiin ja joka myös mahdollistaa robotin objekteihin luoman paineen mittauksen, jotta robotti ei riko, pudota tai

hukkaa kohdeobjektia. Ratkaisuna tähän on kehitetty e-ihoa, eli herkkää ja sähköistä ihon toimintaa matkivaa materiaalia. E-ihon kyky havaita hiuksenhienoja paineita ja lämpötilan muutoksia tekee siitä olennaisen osan esimerkiksi robotin kehoa tai tekoraajoja [27]. Tällaisen tekokäden turvin amputoitu ihminen kykenisi esimerkiksi tarttumaan vadelmaan rikkomatta sen rakennetta.

3.2. Laitteisto

Tässä työssä tutkitaan seuraavien komponenttien ja laitteistojen toimintaa ja pyritään tuottamaan niillä robotin kosketukseen verrattavaa toiminnallisuutta.

- Arduino Uno R3 -kehitysalusta
- Dynamixel XL-320 -servomoottori
- Dynamixel XL 430-W250-T -servomoottori
- Peratech SP200-10 -puristusvoima-anturi

3.2.1. Arduino

Varmasti helpoimpia tapoja lähestyä robotiikkaa on hankkia mikrokontrolleri, kuten Raspberry Pi tai Arduino, joilla voidaan ohjata esimerkiksi servoja sekä vastaanottaa tietoa erilaisilta antureilta. Mikrokontrollereilla voidaan tehdä erilaisia elektroniikkalaitteistoja ohjelmoimalla kyseinen laitteisto halutulla tavalla. Melkeinpä kaikissa vähänkin monimutkaisemmissa elektroniikkalaitteistoissa on mikrokontrolleri. Tällaisissa mikropiireissä on I/O-pinnejä, joihin voi kytkeä komponentteja tiedon lähettämistä ja vastaanottamista varten.



Kuva 6. Arduino Uno -kehitysalusta.

Tutkimustyössä käytetään Arduino Uno (kuva 6). Uno:ssa on 14 digitaalista ja 6 analogista pinniä. Mikrokontrolleri saa virran USB:n kautta tai myös vaihtoehtoisesti erillisen virtalähteen kautta. Uno toimii 5 voltin käyttöjännitteellä, 16 MHz kellotaajuudella ja siinä on flash-muistia 32 KB sekä SRAM:ia 2 KB. Arduino tarjoaa valmiin ohjelmointiympäristön, jota voidaan käyttää koodin luontiin ja sen ajamiseen laitteelle. Koodikielenä on C/C++ -pohjainen Arduinon oma kieli. Koodi perustuu kahteen pääfunktioon (kuva 7):

1. setup() - laitteen asetusten alustaminen,
2. loop() - silmukkarakenne, jota toistetaan virran sammuttamiseen saakka.



```

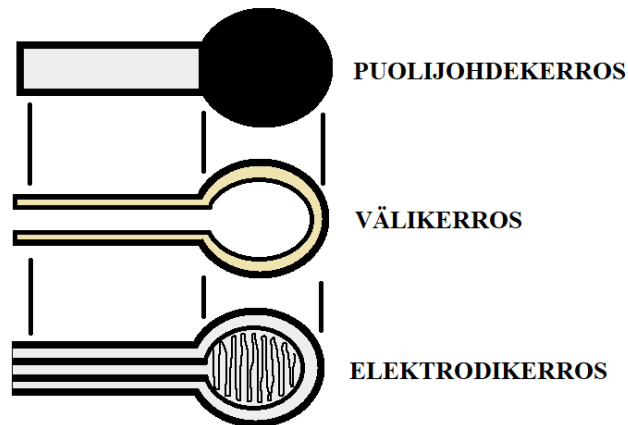
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }

```

Kuva 7. Arduinon ohjelmointiympäristö

3.2.2. Puristusvoima-anturi

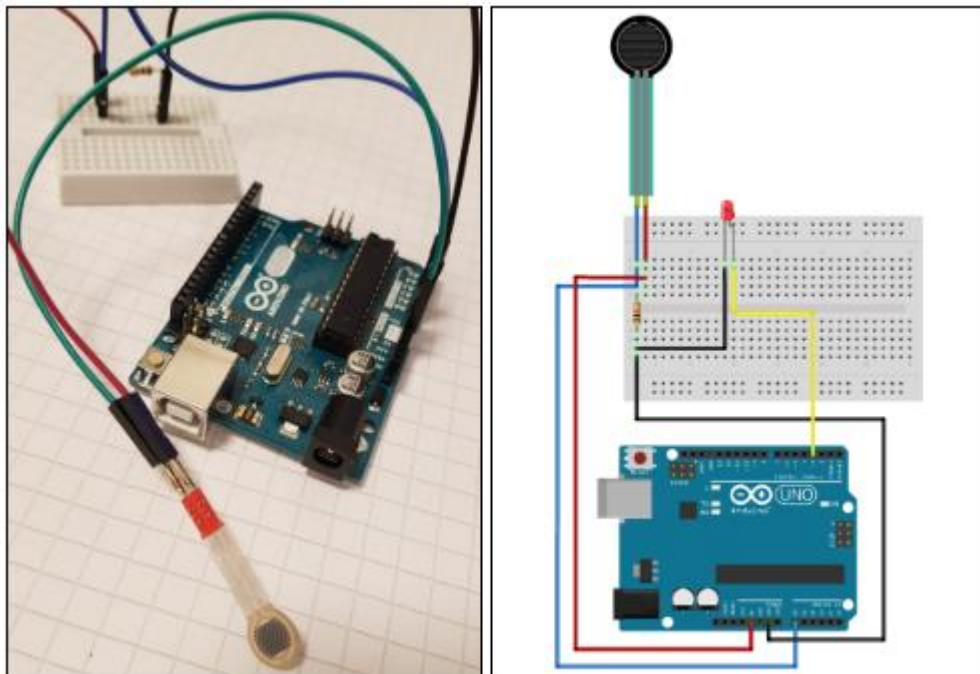
Puristusvoima-antureita käytetään yleisesti esimerkiksi sähköisissä musiikkisoittimissa, robottien tarttumisen ja objektien käsittelyn hallintaan sekä erilaisissa biomekaniikan sovelluksissa. Anturin toiminta perustuu sen resistanssin muuttumiseen, kuten englanninkielisestä termistä FSR (force sensing resistor) käy ilmi. Puristusvoima-anturi koostuu yksinkertaisimmillaan kolmesta eri kerroksesta (kuva 8). Puolijohde- ja elektrodikerroksen erottaa toisistaan välikerros. Mitä enemmän anturiin kohdistetaan painetta, sitä enemmän elektrodiosa koskettaa puolijohdekerrosta, mikä alentaa resistanssia. Anturi on siis oikeastaan vastus, joka muuttaa resistanssiaan riippuen kuinka kovaa siitä painetaan. Tässä työssä on käytetty pyöreää anturia, joka on halkaisijaltaan yhden senttimetrin (Peratech SP200-10).



Kuva 8. Puristusvoima-anturin rakenne.

Puristusvoima-anturin käyttöönotto liitettynä Arduinoon toteutetaan seuraavia komponentteja käyttäen:

- Arduino Uno R3 -kehitysalusta,
- puristusvoima-anturi (FSR),
- 10 k Ω vastus,
- 4 x johdin ja
- koekytkentälevy.



Kuva 9. Puristusvoima-anturin kytkentä Arduinoon (oikealla liitettynä myös led-valo).

Lisäksi toteutetussa testikokoonpanossa (kuva 9) on led-valo, jonka kirkkaus on verrannollinen anturiin kohdistettuun voimaan. Ledin tarkoitus on simuloida myöhemmin liitettävän servomoottorin toimintaa, ja näyttää kuinka koskettamalla puristusvoima-anturia voidaan vaikuttaa oheislaitteiden toimintaan (liite 1). Anturin antaman datan avulla voidaan määrittää raja-arvot esimerkiksi kevyelle kosketukselle tai kovemmalle puristusvoimalle (kuva 10).

```

COM4 (Arduino/Genuino Uno)
Data value = 0 - Pressure rate: Zero
Data value = 0 - Pressure rate: Zero
Data value = 0 - Pressure rate: Zero
Data value = 0 - Pressure rate: Zero
Data value = 0 - Pressure rate: Zero
Data value = 0 - Pressure rate: Zero
Data value = 85 - Pressure rate: Light
Data value = 112 - Pressure rate: Medium
Data value = 100 - Pressure rate: Medium
Data value = 113 - Pressure rate: Medium
Data value = 160 - Pressure rate: Medium
Data value = 209 - Pressure rate: High
Data value = 212 - Pressure rate: High
Data value = 210 - Pressure rate: High
Data value = 208 - Pressure rate: High
Data value = 202 - Pressure rate: High
Data value = 0 - Pressure rate: Zero
  
```

Vieritys Show timestamp NL 9600 baudia Clear output

Kuva 10. Arduino-sarjamonitorin kuvakaappaus. Kuvasta nähdään anturin antamat arvot riippuen anturiin kohdistetun puristuksen voimakkuudesta.

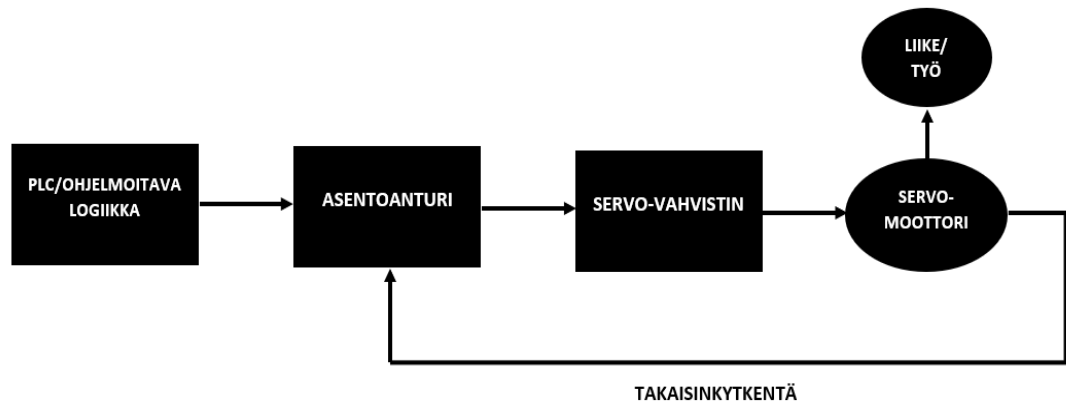
3.2.3. Servomoottorit

Servot ovat yleisesti elektroniikassa ja robotiikassa käytetty sähkömoottorityyppi niiden hyvin soveltuvien ominaisuuksien vuoksi. Kun perinteiset oikosulkumoottorit ovat ominaisuuksiltaan isoja, painavia ja hitaasti kiihtyviä ja pysähtyviä, servomoottorit ovat tarkkoja, kevyitä ja pieniä, sekä vastaavasti nopeasti kiihtyviä ja pysähtyviä [28]. Servomoottoriin sisältyy myös takaisinkytkentä sen paikkatiedosta (ks. kuva 12), eli järjestelmän lähtösignaali kytketään takaisin järjestelmän alkuun tulosignaaliiksi, minkä avulla servoilla voidaan saavuttaa korkea työskentelytarkkuus. Servomoottorit (kuva 11) ovat kuitenkin yleisesti oikosulkumoottoreita kalliimpia valmistaa muun muassa niiden monimutkaisemman rakenteen vuoksi.



Kuva 11. Servomootorit Dynamixel XL-430-W250-T ja XL-320.

Servomootoreissa on liitettyä paikka-anturi, josta saadaan tietoon moottorin tarkka asento. Takaisinkytkentä kulkee servovahvistimelle, joka laskee halutun asennon ja nykyisen asennon erotusta, ja näin ohjaa moottorin haluttuun asentoon määrätyn voima-arvon avulla [28].



Kuva 12. Servomootorin yksinkertaistettu lohkokkaavio

Dynamixel XL430-W250 servon käyttöönotossa ilmeni ongelma, että se ei saa tarpeeksi suurta jännitettä toimiakseen. Servoon syttyi punainen led-valo, mikä ilmoittaa servon ottavan virtaa vastaan. Arduinon antama 5 voltin jännite ei kuitenkaan riitä kyseisen servomootorin liikuttamiseen, joten servo tarvitsee toimiakseen erillisen virtalähteen. Dynamixel XL-320 toimii Arduinon antamalla viiden voltin jännitteellä, vaikka laitteen manuaali kertoo pienemmälle servolle minimijännitteeksi kuusi voltia. Servon käytössä voi kuitenkin ilmetä ongelmia, kun käytetään suositeltua pienempiä jännitteitä.

Taulukko 2. Servojen käyttöjännitteet.

Servo	Dynamixel XL320	Dynamixel XL430-W250
Min (V) - Max (V)	6 - 8.4	6.5 - 12
Suosittelun käyttöjännite (V)	7.4	11.1

XL320-servomoottori pystyy toimimaan maksimissaan 8,4 voltin jännitteellä ja isompi servo (XL430-W250) 12 voltin jännitteellä (taulukko 2). Koska servomootorit ovat hyvin herkkiä liian suurille jännitteille, ne saa helposti rikottua ilman varmuutta niihin kytketystä jännitteestä. Erillisen virtalähteen antama todellinen jännite erosi suuresti virtalähteessä ilmoitetun jännitteen suuruudesta (taulukko 3). Koska virtalähde antoi vääriä lukemia, on hyvin tärkeää olla tarkka todellisen jännitteen suhteen ja mitata se yleismittarilla ennen servoon yhdistämistä.

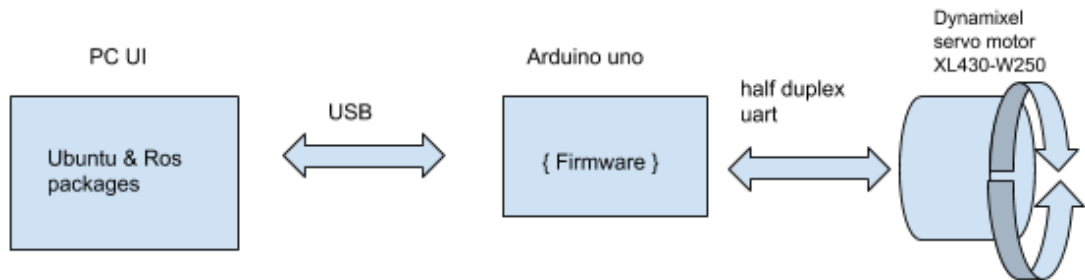
Taulukko 3. Ensimmäisen virtalähteen virheellinen jännitelukema.

Virtalähteeseen asetettu jännite (V)	Yleismittarilla mitattu jännite (V)
4.5	6.9
6.0	8.8
7.5	10.5
9.0	12.6

3.3. ROS

ROS on käyttöjärjestelmä, joka sisältää lukuisia kirjastoja robottiohjelmistojen kirjoittamisen ja ohjaamisen helpottamista varten. ROS:ia käytetään muun muassa autonomisten autojen ja teollisuusrobottien ohjaamiseen. ROS projekti syntyi vuonna 2007, kun Willow Garage alkoi kehittämään työkaluja robottiohjelmistojen nopeampaa kirjoitusta varten. Tutkimuslaitokset ovat tuottaneet monenlaisia hyödyllisiä laitteita, kuten ulkoisia tukirankoja liikuntakyvyttömille ihmisille sekä robottikäsiä, jotka osaavat tarttua esineisiin ja suorittaa tehtäviä.

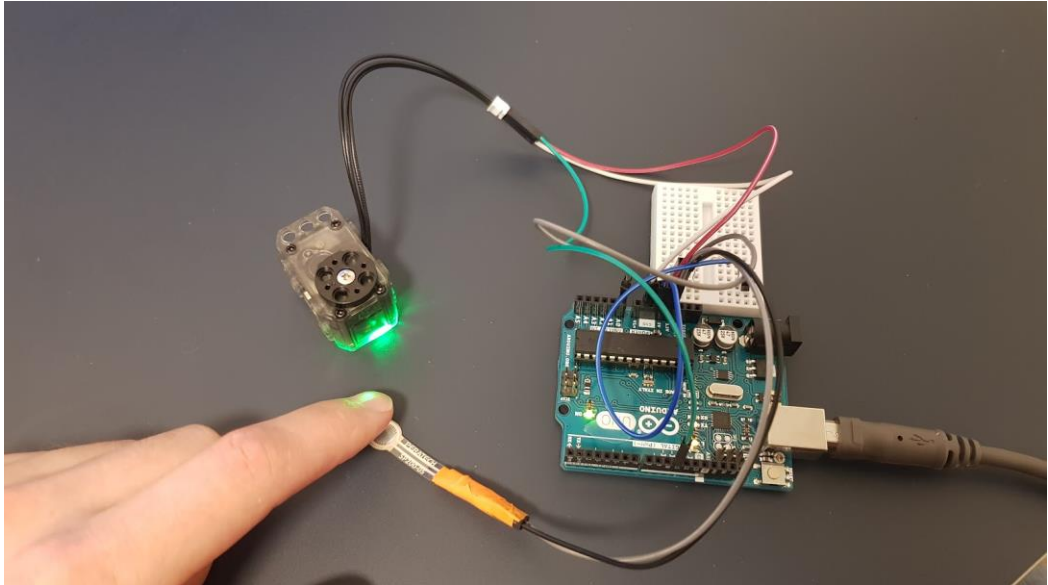
ROS mahdollistaa useampien servojen yhdessä käyttämisen ja informaation vastaanottamisen servoilta, kun välissä käytetään Arduinoa relekoodin kanssa. ROS:ia on käytetty tässä työssä testaamaan XL-430-W250 servon toimintaa kuvan 13 mukaisesti, mutta varsinainen robottikäden tarttumisen testaus on toteutettu ilman ROS:ia.



Kuva 13. Servomoottorin ohjaaminen.

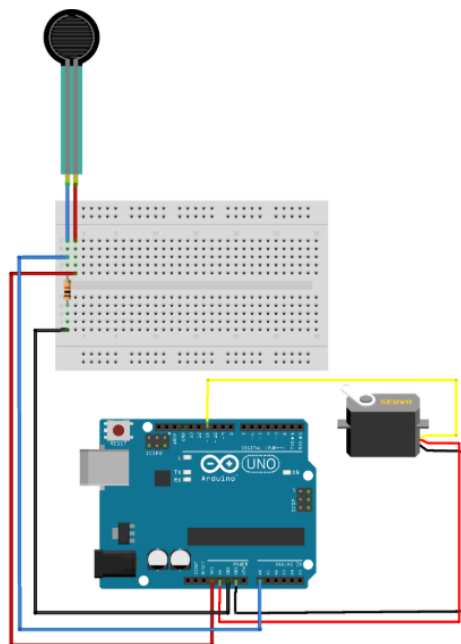
3.4. Anturi-servo -yhdistelmä

Työssä yhdistetään puristusvoima-anturi ja servomoottori (XL-320) Arduinoon (kuvat 13 ja 14), jolloin voidaan simuloida tilannetta, jossa servomoottori (robottitarttuja) reagoi anturin välittämään informaation siihen kohdistuvasta voimasta.



Kuva 14. Servon kontrollointi anturin avulla.

Kuvien 14 ja 15 laiterakenteissa Arduinoon on ajettu koodi (liite 1), joka on ohjelmoitu Arduinon omalla ohjelmointiympäristöllä. Servo on ohjelmoitu tekemään 180° asteen käännös, minkä jälkeen servo alkaa kääntymään 180° toiseen suuntaan. Jos puristusvoima-anturiin kohdistetaan voimaa siirtymisen aikana, servomoottori pysähtyy välittömästi ja vaihtaa suuntaa. Kyseisellä ohjelmalla voidaan testata servomoottorin ja puristusvoima-anturin yhteistoimintaa, mikä osoittautui toimivaksi. Rakennettu piiri ei kuitenkaan mahdollista informaation kulkua servosta Arduinoon, eli servosta ei saada kyseisellä kokoonpanolla tietoa esimerkiksi servomoottorin paikasta, lämpötilasta ja monesta muusta informaatiosta.



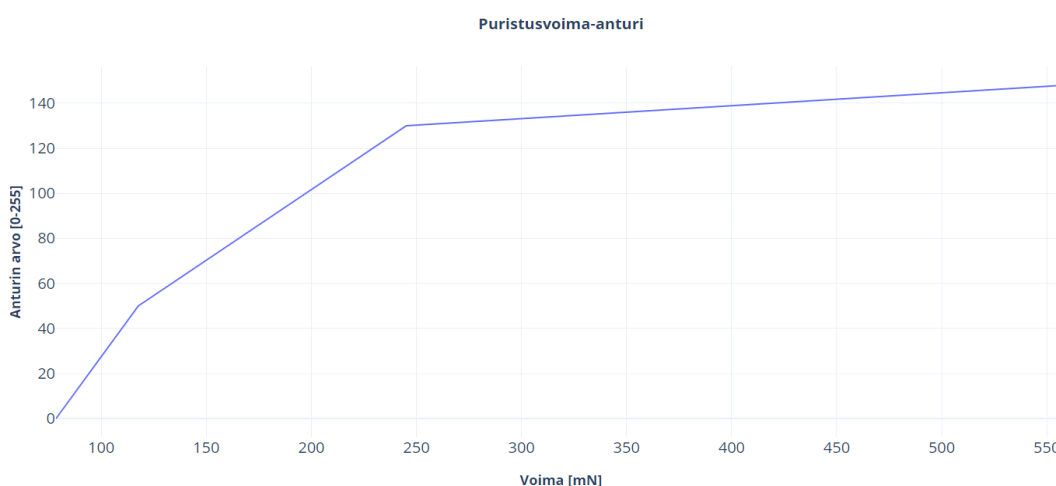
Kuva 15. Anturin ja XL-320-servon kytkentä

Taulukossa 4 on testattu puristusvoima-anturin herkkyyttä eri muotoisilla ja painoisilla esineillä. Anturin päälle asetettiin mm. kynä, pyyhekumi, paristo ja kananmuna, ja anturin antamat arvot kirjattiin ylös.

Taulukko 4. Puristusvoima-anturin herkkyuden testaus.

Esine	Massa (g)	Kohdistuva voima (N)	Anturin palauttama arvo [0-255]
Pyyhekumi	9	0,07848	0
Pyyhekumi	10	0,0981	25
Kynä	12	0,11772	50
AA-paristo	25	0,24525	130
Kananmuna	57	0,55917	148

Taulukosta 4 huomataan että kun puristusvoima-anturiin kohdistuu alle 0,07848 N voima, se palauttaa arvon 0. Kappaleen muoto ja materiaali voivat myös vaikuttaa anturin antamiin arvoihin. Anturi vaikuttaa laadukkaalta ja tarkalta; laite vastaa hyvin siihen kohdistettuun voimaan ja tunnistaa jo 10 gramman painon, kun kappale asetetaan anturin päälle. Taulukosta voidaan siis päätellä kyseisen anturin reagoivan hieman alle 0,1 Newtonin suuruiseen voimaan, joka vastaa suurin piirtein kahden euron kolikon kannatteleamiseen tarvittavaa voimaa.

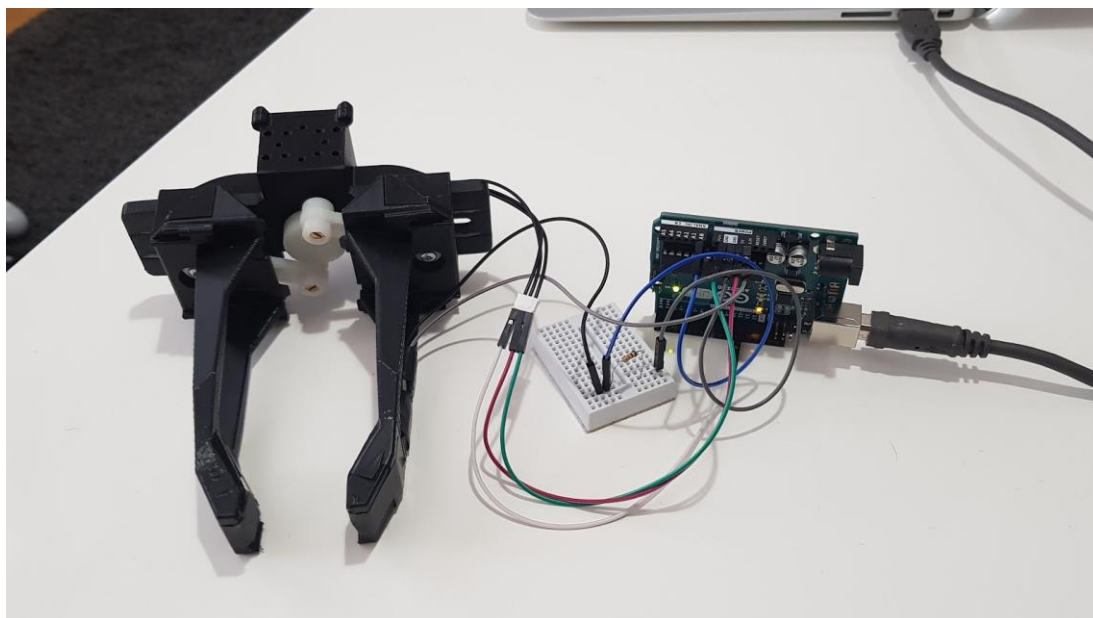


Kuva 16. Anturin palautusarvo suhteessa siihen kohdistuvaan voimaan.

Testituloksista tehty kuvaaja (kuva 16) näyttää, kuinka käyrä muuttuu verrattaessa anturiin kohdistettavaa voimaa sen palauttamaan arvoon. Anturi antaa melko loogisia ja tarkkoja tuloksia, vaikkakin testauksessa on syntynyt pieniä tulosten epätarkkuuksia. Kuvaajan käyrä loivenee selvästi noin 0,25 Newtonin jälkeen, mistä voidaan päätellä anturin soveltuvan paremmin hyvin pienten voimien havainnointiin, eli noin alueelle 0.1 - 0.25 Newtonia. Puristusvoima-anturit ovat kuitenkin huono vaihtoehto esimerkiksi niihin kohdistuvan massan tarkkaan mittaukseen, ja niitä käytetäänkin enemmän sovelluksissa, joissa halutaan määrittää, onko objektiin kohdistunut voimaa ja kuinka kauan.

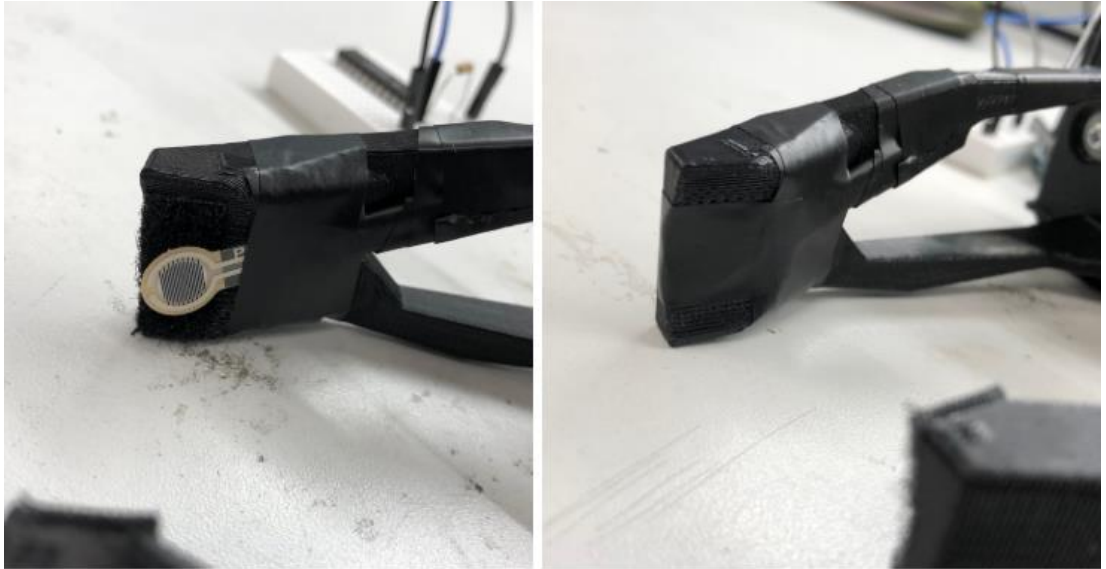
4. LAITTEISTON TESTAUS JA TULOKSET

Osana Sulautettujen ohjelmistojen projektia (521275A) työssä testattiin puristusvoima-anturin soveltuvuutta osaksi robottitarttujaa, joka kykenisi tarttumaan shakkinappuloihin tai muihin pienikokoisiin esineisiin. Pienempi servomoottori XL-320 on valittu toteuttamaan käden osa, jolla esineisiin tartutaan. Dynamixel XL-320 on tarpeeksi suorituskykyinen servo tarttumaan jopa hieman painavammista esineistä ja se on hyvä valinta robottikäden päähän vain 16,7 gramman painonsa ja pienen fyysisen kokonsa ansiosta. Kurssin pohjustuksena servomoottorille on Oulun yliopiston tietotekniikan osaston toimesta 3D-tulostettu Fab Labin tiloissa runko ja muut osat (kuva 17), joilla saadaan aikaan sulkeutuva ja avautuva rakenne pyörittämällä servomoottoria. Tarttujan rungon mustat osat on valmistettu PC/ABS-seoksesta ja valkoiset vahvasta hartsimateriaalista. Koko käden rakenne on toteutettu 3D-tulostuksilla sekä vinyylilevyn laserleikkauksilla; tarkempi kuvaus ja CAD-tiedostot löytyvät GitHubista [29].



Kuva 17. 3D-tulostettu robottitarttuja komponentteineen.

Servomoottorin pyörimisen kalibrointiin käytettiin aikaisemman testikokoonpanon laitteistoa, jotta välttyttäisiin 3D-tulostettujen osien rikkoutumiselta. Kalibroinnissa otettiin ylös moottorin astekulmat tarttujan ääriasennoissa, ja asetettiin ohjelmakoodiin uudet ääriarvot edestakaiselle pyörimiselle. Kun uudet asetukset todettiin turvallisiksi käyttää, asetettiin servomoottori sisään sille valmistettuun koteloon. Johtuen tarttumakohtien hieman koverasta rakenteesta, ennen puristusvoima-anturin liittämistä sen alle on asetettu MDF-levyn palaset antamaan anturille tasainen ja kova alusta (kuva 18), jolloin anturin voimaan reagoiva herkkyys on parempi.







Kuva 18. Puristusvoima-anturi tarttujaan kiinnitettynä, oikealla teipattuna levyä vasten.

4.1. Testaus

Yhdistelmän toimivuutta käytännössä testattiin muutamilla eri muotoisilla, kokoisilla ja painoisilla kappaleilla (taulukko 5). Testauksessa käytettiin pyyhekumia, noppaa, kananmuna ja pelinappulaa. Testauksen päämääränä oli saada selville, onko voima-anturin käytöstä robottitarttujan yhteydessä hyötyä, ja ylipäätään testata kyseisen rakenteen soveltuvuus osaksi projektin tarkoitusta.

Taulukko 5. Tarttujan testaukseen käytetyt esineet

Kappale	 Pyyhekumi	 Noppakuutio	 Kananmuna	 Pelinappula
Paino	2 g	5 g	59 g	8 g

Testiohjelmassa servon led-valon väri on koodattu seuraavasti helpottamaan eri tilanteiden tunnistamista:

- Vihreä
 - Puristusvoima-anturiin ei kohdistu voimaa
 - Servo on ääri-asennossa eli tarttuja on auki ja servo jatkaa pyörimistä sulkeakseen tarttujan.
- Punainen
 - Puristusvoima-anturi on havainnut siihen kohdistuvan voiman, tai
 - Servo on ääri-asennossaan eli tarttuja on suljettuna
 - Viiveen jälkeen tarttuja palautuu auki-asentoon
- Ledien vilkkuminen (kuvissa keltainen valo)
 - Servon oma sisäänrakennettu ominaisuus, joka suojaa laitetta rikkoutumiselta
 - Servo on joutunut vikatilaan ja lopettaa vääntämisen

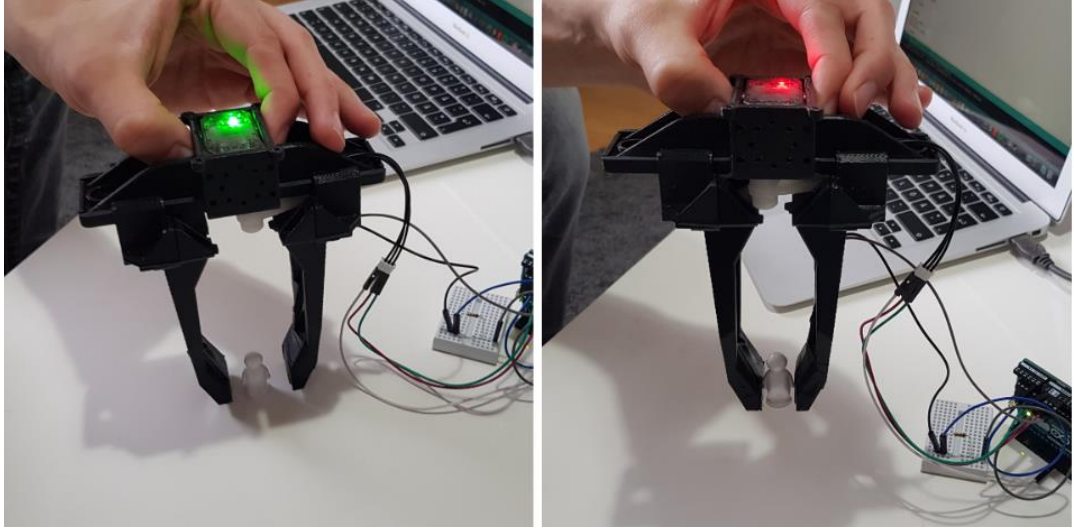


Kuva 19. Noppakuutioon tarttuminen nopan eri asennoilla.

Heti ensimmäisten testausten aikana ilmeni, että anturin elektrodiosan pinta-ala on liian pieni tähän käyttötarkoitukseen. Kun voima-anturia vasten painautuu tasainen pinta, anturi ei havaitse tätä. Hyvänä esimerkkinä toimii noppakuutio. Kun noppa laitetaan tasainen pinta edellä anturia vasten (kuva 19 vas. ja kesk. kuva), tämä ei tunnista sitä ja yrittää painaa esinettä kasaan. Kun noppa asetetaan kulma edellä puristusvoima-anturia vasten, laitteisto toimii ongelmitta, eli anturi huomaa siihen kohdistuvan voiman ja servo lopettaa pyörimisen (kuva 19 oik.). Tällöin tarttujalla on pitävä ote kuutiosta. Testien yhteydessä ilmeni myös toinen tärkeä tilanne servon käyttämisessä. Kun servo ei enää kykene vääntämään kohdatessaan riittävän suuren vastakkaisen voiman momentin, se menee vikatilaan. Tällaisessa tilanteessa koko järjestelmästä joudutaan ottamaan virrat pois ja kytkemään ne takaisin, jotta servon toiminta palaa normaaliksi. Näin ollen voima-anturin käyttö on tärkeää, jotta servo osaa lopettaa pyörimisen ajoissa eikä mene vikatilaan tai pahimmassa tapauksessa rikki.

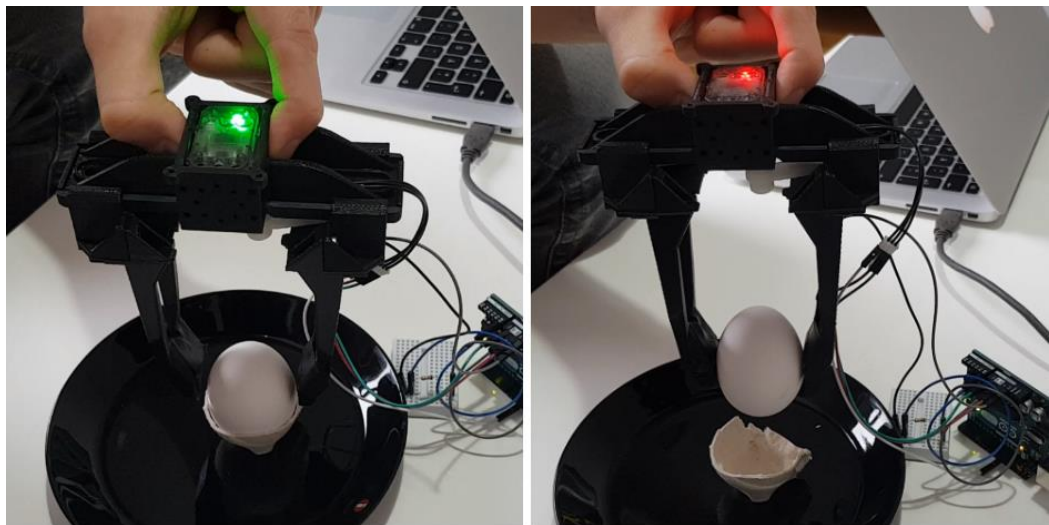
Voimasensorin käyttö toimi parhaiten pyöreän pyyhekumin, kallellaan olevan nopan ja pelinappulan kanssa. Pelinappulaan tarttuminen onnistui ongelmitta

jokaisesta asennosta; servo lopetti välittömästi tarttujan liikuttamisen jokaisella kerralla, kun pelinappulan painautumisesta aiheutuva voima kohdistui anturiin (kuva 20). Vaikka testitilanteessa ei ollut saatavilla oikeaa shakkinappulaa, satunnaisen pelinappulan muoto on hyvin samantyyppinen, jolloin olisi oletettavaa, että isoja ongelmia ei esiintyisi myöskään shakkinappulan kanssa.



Kuva 20. Pelinappulaan tarttuminen.

Servo meni jälleen vikatilaan, kun tarttujan väliin asetettiin kananmuna. Kananmuna on ilmeisesti liian suuri, kun käytössä on pinta-alaltaan näin pieni voima-anturi. Kananmuna ei kuitenkaan rikkoontunut, mikä johtuu vikatilan aktivoitumisesta, tai yksinkertaisesti siitä, että servon voima ei riittänyt rikkomaan tätä. Monen yrityksen jälkeen anturi saatiin kuitenkin huomaamaan kananmunan aiheuttama voima, jolloin kananmunaan tarttuminen ja nostaminen tarttujan avulla onnistui (kuva 21).

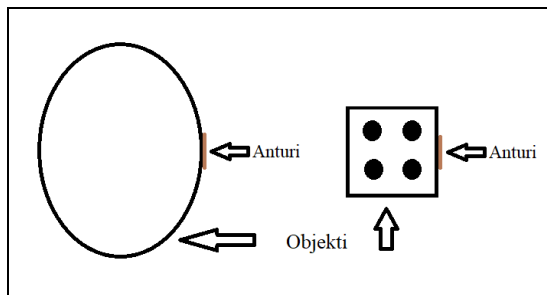


Kuva 21. Kananmunaan tarttuminen.

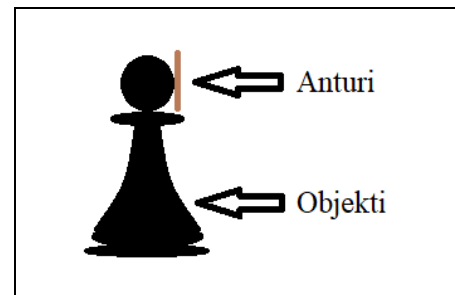
4.2. Testauksen yhteenveto ja tulokset

Testien perusteella tarttujarakenteen toimivuus riippuu pitkälti puristusvoima-anturin toimivuudesta. Kuvissa 22 ja 23 on kuvattu erilaiset pinnat, jotka käytössä ollut puristusvoima-anturi pystyi tai ei pystynyt havaitsemaan. Kun pinta oli täysin tasainen (kuva 22), anturi ei havainnut sitä lainkaan. Hieman kaarevalla, mutta anturin pinta-alaa selvästi suuremmalla muodolla saavutettiin epämääräisiä tuloksia; tarttujalla kyettiin tarttumaan tällaiseen esineeseen noin yhdellä neljästä yrityskerrasta. Vastaavasti kun esineen pinta oli selvästi anturin pintaa pienempi ja se kohdisti anturiin hieman pistemäisen paineen, anturi toimi lähes virheettömästi ja kappaleeseen tarttuminen onnistui.

Anturin avulla voidaan myös säätää voimaa, jolla tarttuja ottaa kappaleen kiinni. Tämä mahdollistaa sen, että tarttujalla voitaisiin tarttua moniin erilaisiin materiaaleihin ja esineisiin. Esimerkiksi raskaamman materiaalin nostamiseen tarvitaan enemmän voimaa, jotta se pysyy tarttujassa. Kevyemmän kappaleen nostamiseen ei tarvita kovin suurta voimaa, joten käytettävää voimaa voitaisiin säätää ohjelmakoodissa hieman pienemmälle.



Kuva 22. Ei-optimaaliset kosketuspinnat.



Kuva 23. Optimaalinen kosketuspinta

4.3. Jatkokehitys

Projektin jatkokehityksessä on monenlaisia asioita, joita voidaan ottaa paremmin huomioon, kun halutaan toteuttaa robottitarttuja, joka kykenee virheettömästi tarttumaan eri kappaleisiin. Testauksissa useaan otteeseen esiin tuli puristusvoima-anturin pinta-ala. Koska käytössä ollut puristusvoima-anturi oli muodoltaan pyöreä ja halkaisijaltaan vain yhden senttimetrin, se ei ollut juuri tähän tarkoitukseen optimaalinen, vaikkakin ajoi joissain tilanteessa asiansa oikein hyvin. Parempi anturi voisi olla suorakulmion muotoinen ja elektrodiosan pinta-alaltaan hieman suurempi, jolloin se vastaisi paremmin tarttumiskohdan muotoa. Tällöin voiman havaitseminen ei olisi niin riippuvainen esineen muodosta tai asennosta. Lisäksi tarkkuutta voisi parantaa kahden erillisen puristusvoima-anturin käyttö tarttujan molemmissa pihdeissä, jolloin voitaisiin hyödyntää molemmista antureista saatavaa dataa ja saavuttaa parempi herkkyys koskettaessa kappaleisiin.

Koska testausohjelma oli tarkoitettu lähinnä puristusvoima-anturin ja servomoottorin yhteistoiminnan kokeiluun, olisi siihen tehtävä vielä joitain periaatteellisia muutoksia. Huomiota olisi kiinnitettävä muun muassa viiveeseen, eli kuinka kauan tarttuja pitää kappaletta puristusotteessaan. Projektin lopullisen tarttujan toiminnan kanssa riittää varmasti vielä paljon töitä näiden asioiden ratkaisemiseksi, varsinkin kun mietitään lopullista toteutustapaa.

5. PROJEKTIN KUVAUS

Projekti toteutettiin suurimmaksi osaksi yhteisissä tapaamisissa. Tapaamisissa suunniteltiin projektin toteutusta ja tehtiin erilaisia testauksia. Lisäksi tekijät ovat käyttäneet aikaa itsekseen kirjoitustyöhön sekä aiheesta opiskeluun ja muihin erilaisiin selvitystöihin. Taulukossa 6 on esitetty projektiin osallistuneiden henkilöiden työhön käyttämät tunnit.

Taulukko 6. Työhön käytetyt tunnit projektin jäsenten välillä.

Tekijän nimi	Työhön käytetty tuntimäärä
Jonne Taipale	159
Joonas Päivärinta	164
Eliud Keino (ei kandidaatintyö)	155

6. YHTEENVETO

Kandidaatintyössä esiteltiin robotiikkaa yleisellä tasolla, robotiikan historiaa, nykyajan käyttökohteita, teknologioita ja niihin liittyviä turvallisuuskysymyksiä. Projektin tarkoituksena oli myös tutkia puristusvoima-anturin ja servomoottorien yhteistoiminnalla toteutettavaa robottitarttujaa, jonka tarkoituksena on kyetä tarttumaan esineisiin niitä rikkomatta tai pudottamatta.

Työssä käytetty puristusvoima-anturi osoittautui laadukkaaksi ja tarkaksi komponentiksi. Peratech SP200-10 -anturi tunnistaa jo 10 gramman painon tämän päällä, mikä vastaa 0,0981 newtonin voimaa. Työssä tehtiin Arduinon omalla ohjelmointiympäristöllä ohjelmakoodi, jolla saatiin testattua anturin toimintaa robottitarttujassa.

Puristusvoima-anturin huomattiin olevan hyödyllinen servon apuna robottitarttujan toiminnalle. Näin voidaan tarttua esineisiin ilman, että robottitarttuja tai itse servo rikkoontuu liiallisen puristuksen takia. Käytössä olleen puristusvoima-anturin elektrodipinta on kuitenkin liian pieni, kun tarttujalla yritetään ottaa isompia esineitä. Suuremmalla elektrodipinnalla voitaisiin tarttua myös suurempiin esineisiin ja anturi voisi tunnistaa paremmin esineiden tasaisia pintoja.

7. LÄHTEET

- [1] Hockstein, N. G., Gourin, C. G., Faust, R. A., & Terris, D. J. (2007). A history of robots: from science fiction to surgical robotics. *Journal of robotic surgery*, 1(2), 113-118.
- [2] Murphy, R., & Woods, D. D. (2009). Beyond Asimov: the three laws of responsible robotics. *IEEE Intelligent Systems*, 24(4).
- [3] Hsu, F. (2004). Behind deep blue: Building the computer that defeated the world chess champion Princeton University Press.
- [4] Poupyrev, I., Nashida, T., & Okabe, M. (2007). Actuation and tangible user interfaces: The vaucanson duck, robots, and shape displays. Paper presented at the Proceedings of the 1st International Conference on Tangible and Embedded Interaction, 205-212.
- [5] Bergin, J., Stehlik, M., Roberts, J., & Pattis, R. (2005). Karel J. robot: A gentle introduction to the art of object-oriented programming in java Dream Songs.
- [6] Litman, T. (2017). Autonomous vehicle implementation predictions Victoria Transport Policy Institute Victoria, Canada.
- [7] Schwarz, B. (2010). LIDAR: Mapping the world in 3D. *Nature Photonics*, 4(7), 429.
- [8] Jo, K., Kim, J., Kim, D., Jang, C., & Sunwoo, M. (2014). Development of autonomous car—Part I: Distributed system architecture and development process. *IEEE Transactions on Industrial Electronics*, 61(12), 7131-7140.
- [9] AutoBild, Juha Peltonen (22.10.2018) Autonominen ajaminen: Tiedätkö, miten pitkällä itsestään ajavien autojen kehitys tällä hetkellä on? <https://autobild.fi/artikkeli/autonominen-ajaminen-tiedatko-miten-pitkalla-itsesta-an-ajavien-autojen-kehitys-talla-hetkella-on-6.182.120316.6411d5cf93>
- [10] Bonin-Font, F., Ortiz, A., & Oliver, G. (2008). Visual navigation for mobile robots: A survey. *Journal of Intelligent and Robotic Systems*, 53(3), 263-296.
- [11] Fryman, J., & Matthias, B. (2012). Safety of industrial robots: From conventional to collaborative applications. Paper presented at the Robotics; Proceedings of ROBOTIK 2012; 7th German Conference On, 1-5.
- [12] Heyer, C. (2010, October). Human-robot interaction and future industrial robotics applications. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 4749-4754). IEEE.
- [13] Sheridan, T. B. (2016). Human–robot interaction: status and challenges. *Human factors*, 58(4), 525-532

- [14] Taylor, R. H., Menciassi, A., Fichtinger, G., Fiorini, P., & Dario, P. (2016). Medical robotics and computer-integrated surgery. Springer handbook of robotics (pp. 1657-1684) Springer.
- [15] Okamura, A. M., Mataric, M. J., & Christensen, H. I. (2010). Medical and health-care robotics. *IEEE Robotics & Automation Magazine*, 17(3), 26-37.
- [16] Shah, J., Vyas, A., & Vyas, D. (2014). The history of robotics in surgical specialties. *American Journal of Robotic Surgery*, 1(1), 12-20.
- [17] Mubin, O., Stevens, C. J., Shahid, S., Al Mahmud, A., & Dong, J. J. (2013). A review of the applicability of robots in education. *Journal of Technology in Education and Learning*, 1(209-0015), 13.
- [18] Baxter, P., Ashurst, E., Read, R., Kennedy, J., & Belpaeme, T. (2017). Robot education peers in a situated primary school study: Personalisation promotes child learning. *PloS one*, 12(5), e0178126.
- [19] Reich-Stiebert, N., & Eyssel, F. (2016, November). Robots in the classroom: what teachers think about teaching and learning with education robots. In *International Conference on Social Robotics* (pp. 671-680). Springer, Cham.
- [20] Kim, S. W., & Lee, Y. (2015, October). A Survey on Elementary School Teachers' Attitude toward Robot. In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (pp. 1802-1807). Association for the Advancement of Computing in Education (AACE).
- [21] Park, E., Kim, K. J., & Del Pobil, A. P. (2011, November). The effects of a robot instructor's positive vs. negative feedbacks on attraction and acceptance towards the robot in classroom. In *International Conference on Social Robotics* (pp. 135-141). Springer, Berlin, Heidelberg.
- [22] Financial Times, Ed Crooks (1.7.2015). Worker at Volkswagen plant killed in robot accident. <https://www.ft.com/content/0c8034a6-200f-11e5-aa5a-398b2169cf79>
- [23] Tekniikka & Talous, Janne Luotola (1.7.2016). Ensimmäinen kuolonuhri – Robottiauto tappoi kuljettajansa, viranomaiset etsivät syyllistä. <https://www.tekniikkatalous.fi/tekniikka/autot/ensimmainen-kuolonuhri-robottiauto-tappoi-kuljettajansa-viranomaiset-etsivat-syylista-6563767>
- [24] Guo, D., Sun, F., Fang, B., Yang, C., & Xi, N. (2017). Robotic grasping using visual and tactile sensing. *Information Sciences*, 417, 274-286.
- [25] Van den Heever, D. J., Schreve, K., & Scheffer, C. (2009). Tactile sensing using force sensing resistors and a super-resolution algorithm. *IEEE Sensors Journal*, 9(1), 29-35.

- [26] Fritzsche, M., Elkmann, N., & Schulenburg, E. (2011, March). Tactile sensing: A key technology for safe physical human robot interaction. In *Proceedings of the 6th International Conference on Human-robot Interaction* (pp. 139-140). ACM.
- [27] Dahiya, R. (2019). E-Skin: From Humanoids to Humans. *Proceedings of the IEEE*, 107(2), 247-252.
- [28] Kauria, T. (2013). Robotin ja konenäköjärjestelmän liittäminen.
- [29] GitHub, Teemu Tokola (ttokola), roboticTurk: K.O.U.R.A. (1st major version)
<https://github.com/ttokola/roboticTurk/blob/master/docs/KOURA.md>

8. LIITTEET

Liite 1. Puristusvoima-anturin ja servomoottorin testauksen koodit.

Liite 1. Puristusvoima-anturin ja servomoottorin testauksen koodit.

```
✓ ↻ 📄 ⬆️ ⬇️ Varmista
Fsr$
1 /* Arduino program to read force sensor
2  * and send pwm to the LED/servo
3  */
4 //Initializing pins 3 for LED vizulization
5 const int ledOut = 3;
6 //Defining Analog input pin
7 const int forceSensor = A0;
8 //Storing analog value
9 int analogData;
10
11 void setup(){
12 //light LED up when force is high
13 pinMode(ledOut,OUTPUT);
14 //send data to serial at 9600 baudrate
15 Serial.begin(9600);
16 }
17
18 void loop() {
19 //read and store analog data from pin A0 and store in analogData
20 analogData = analogRead(forceSensor);
21
22 // mapping sensor values(analogData) from 0-1023 to 0-255
23 analogData = map(analogData,0,1023,0,255);
24
25 //print analogData to serial
26 Serial.print("Data value = ");
27 Serial.print(analogData);
28 Serial.print(" - Pressure rate: ");
29
30 //Send data to led to light it up; to be changed to servo
31 analogWrite(ledOut,analogData);
32
33 //Setting threshold if the pressure detected is equivalent to the given condition
34 if(analogData < 1){
35     Serial.println("Zero");
36 }
37 else if(analogData < 100){
38     Serial.println("Light");
39 }
40 else if(analogData <200){
41     Serial.println("Medium");
42 }
43 else{
44     Serial.println("High");
45 }
46 //short delay to show changes in pressure
47 delay(300);
48 }
```

Testausohjelma 1. Puristusvoima-anturin testaaaminen.

Liite 1. Puristusvoima-anturin ja servomoottorin testauksen koodit.

```
servo_and_sensor
1
2 #include "XL320.h"
3 #include <SoftwareSerial.h>
4
5 //servo name
6 XL320 robot;
7 int forceSensor = A0;
8 int analogData;
9 int asento = 980;
10 SoftwareSerial mySerial(10, 11); // (RX, TX)
11 char rgb[] = "rgbypcwo";
12 int ledColour = 0;
13 int servoID = 1;
14
15 void setup() {
16     // Setup Software Serial
17     mySerial.begin(1000000);
18
19     // Initialise your robot
20     robot.begin(mySerial); // Hand in the serial object you're using
21
22     // Setting joint speed
23     robot.setJointSpeed(servoID, 120);
24
25     //Send data to serial at 9600 baudrate
26     Serial.begin(9600);
27     analogData = analogRead(forceSensor);
28
29     robot.moveJoint(1, asento);
30 }
31
32 void loop() {
33
34     //Set joint values between 980-350 to turn about 180 degrees)
35     robot.LED(1, &rgb[0]);
36     for (asento; asento >= 350; asento -= 10) {
37         robot.moveJoint(1, asento);
38         analogData = analogRead(forceSensor);
39
40         // mapping sensor values(analogData) from 0-1023 to 0-255
41         analogData = map(analogData, 0, 1023, 0, 255);
42
43         Serial.print("Data value = ");
44         Serial.println(analogData);
45         delay(100);
46         if(analogData >= 50){
47             break;
48         }
49     }
50
51     delay(2000);
52
53     robot.LED(1, &rgb[1]);
54     for (asento; asento <= 980; asento += 10) {
55         robot.moveJoint(1, asento);
56         analogData = analogRead(forceSensor);
57
58         // mapping sensor values(analogData) from 0-1023 to 0-255
59         analogData = map(analogData, 0, 1023, 0, 255);
60
61         Serial.print("Data value = ");
62         Serial.println(analogData);
63         delay(100);
64         if(analogData > 50)
65             break;
66     }
67     delay(2000);
68 }
```

Testausohjelma 2. Puristusvoima-anturi ja servo.

Liite 1. Puristusvoima-anturin ja servomoottorin testauksen koodit.

```
testaus $
1
2 #include "XL320.h"
3 #include <SoftwareSerial.h>
4
5 //servo name
6 XL320 robot;
7
8 int forceSensor = A0;
9 int analogData;
10 int asento = 600;
11
12 // Set the SoftwareSerial RX & TX pins
13 SoftwareSerial mySerial(10, 11); // (RX, TX)
14
15 // Set some variables for incrementing position & LED colour
16 char rgb[] = "rgbypcwo";
17 int ledColour = 0;
18 // Set the default servoID to talk to
19 int servoID = 1;
20
21 void setup() {
22
23     // Setup Software Serial
24     mySerial.begin(1000000);
25
26     // Initialise your robot
27     robot.begin(mySerial); // Hand in the serial object you're using
28
29     // Setting joint speed
30     robot.setJointSpeed(servoID, 120);
31
32     //setup for sensor
33     //send data to serial at 9600 baudrate
34     Serial.begin(9600);
35     analogData = analogRead(forceSensor);
36
37     robot.moveJoint(1, asento);
38
39 }
40
41
42 void loop() {
43
44     // Led turns green when closing the gripper
45     robot.LED(1, &rgb[1]);
46
47     // When asento == 1030, gripper is completely closed
48     for (asento; asento <= 1025; asento += 5) {
49
50         robot.moveJoint(1, asento);
51         analogData = analogRead(forceSensor);
52         // mapping sensor values(analogData) from 0-1023 to 0-255
53         analogData = map(analogData,0,1023,0,255);
54         Serial.print("Data value = ");
55         Serial.println(analogData);
56         delay(10);
57         if(analogData > 15) {
58             // Led turns red when force is applied to sensor
59             robot.LED(1, &rgb[0]);
60             break;
61         }
62         delay(100); // Slows down the closing
63     }
64
65     delay(4000);
66
67     robot.LED(1, &rgb[0]);
68     asento = 600; // Opens the gripper
69     robot.moveJoint(1, asento);
70
71     delay(2000); // Time to place the object in the gripper
72 }
```

Testausohjelma 3. Tarttujan testaaminen.