

*Robottiohjelmoinnin harjoitustyö*

# BALANSSIBOTTI

**Jouni Koponen**  
012602454  
[jouni.koponen@helsinki.fi](mailto:jouni.koponen@helsinki.fi)

## Sisällysluettelo

<b>1. <u>BALANSSIBOTTI</u></b>	<b>3</b>
<b>1.1. YLEISKUVAUS</b>	<b>3</b>
<b>1.2. PID-SÄÄDIN</b>	<b>3</b>
<b>1.3. ROBOTIN RAKENNE</b>	<b>4</b>
<b>2. <u>KOODI</u></b>	<b>8</b>
<b>2.1. OHJELMOINTIYMPÄRISTÖ</b>	<b>8</b>
<b>2.2. KOODIN RAKENNE</b>	<b>8</b>
<b>3. <u>TESTAUS</u></b>	<b>9</b>
<b>3.1. TESTITAPAUS 1</b>	<b>9</b>
<b>3.2. TESTITAPAUS 2</b>	<b>10</b>
<b>3.3. TESTITAPAUS 3</b>	<b>10</b>
<b>4. <u>ROBOTIN JATKOKEHITYS JA ONGELMAT</u></b>	<b>10</b>
<b>5. <u>KÄYTTÖ- JA RAKENNUSOHJE</u></b>	<b>11</b>
<b>5.1. BALANSSIBOTIN KÄYTTÖ</b>	<b>11</b>
<b>5.2. BALANSSIBOTIN RAKENNUSOHJE</b>	<b>12</b>

## 1. Balanssibotti

### 1.1. Yleiskuvaus

Balanssibotti on *Lego Mindstorm NXT (1. Versio)* -sarjan avulla rakennettu robotti, joka tasapainoilee kahdella pyörällä. Robotti muistuttaa perusidealtaan *Segway*-moottoroitua kulkuvälinettä, jossa kuljettaja seisoo kahden pyörän välissä liikkuen eteen- tai taaksepäin, mutta toisaalta segway kykenee tasapainoilemaan paikallaankin.

Alussa robotti asetetaan käsin tasapainoon, jolloin saadaan tasapainon referenssiarvo. Balanssibotti käyttää hyödykseen valosensoria siten, että se lukee valosensorin lukemia ja yrittää korjata asentoaan vertaamalla arvoa aluksi asetettuun tasapainon referenssiarvoon. Robotti pyörittää moottoreihin kytkeytyjä renkaita sen verran, että tasapaino saadaan automaattisesti korjattua.

Lisäksi robottiin voidaan kytkeä kolmas moottori NXT yksikön päälle, jolloin robotin tasapainopistettä voidaan manuaalisesti muuttaa ja saada robotti liikkumaan haluttuun suuntaan, käytännössä kuitenkin eteen- tai taaksepäin. Tätä kolmatta moottoria voidaan ohjata joko usb- tai bluetooth-yhteyden kautta.

Robotin toiminta perustuu pääasiallisesti säätötekniikasta tuttuun PID-säätimen (*Proportional-Integral-Derivative*) ideaan. Tätä kuvataan tarkemmin seuraavassa luvussa 1.2.

Robotin pääasiallinen tarkoitus on pitää alussa robottille asetettu tasapaino. Robotti ei itse osaa hakea tasapainoa, jos robottia ei ole aluksi tasapainotettu. On myös oleellista huomata, että käytettävällä alustalla on merkitystä robotin toiminnalle. Myös valaistusolosuhteet saattavat vaikuttaa toimintaan. Robottia tulisi käyttää heiastamattomalla vaalealla alustalla. Valkoinen paperi on esimerkiksi hyvä materiaali.

### 1.2. PID-säädin

Robotin toiminnan kannalta on aivan keskeistä ymmärtää PID-säätimen ideaa. Sisäänmeno arvona käytetään asetusarvon ja mittausarvon erotusta. Aetusarvo on alussa käsin asetettu tasapainopiste, toisin sanoen tämän tasapainopisteen valoanturin arvo. Ulostuloarvo koostuu kolmesta eri termistä, mistä myös säätimen nimi tulee: suhde, integroiva ja derivoiva. Robotti yrittää siis pitää säätövirheen mahdollisimman pienenä.

Matemaattisesti, kun ulostolo arvo on  $u$  ja erosuure on  $e$  niin,

$$u = K_P e + K_I \int e dt + K_D \frac{de}{dt}$$

missä  $K_p$  kuvaavat vahvistusta,  $K_i$  integroi erosuuretta ajassa sekä  $K_d$  huomioi erosuureen muutosnopeutta.

PID-säätimen virittäminen on ollut robotin rakennuksen ja onnistuneen tasapainoilun kulmakiviä. Käytettävissä on erilaisia matemaattisia ratkaisuja PID-säätimen virittämiseen, mutta tässä harjoitustyössä on edetty enemmänkin yritys ja erehdys menetelmällä. Säädintä on siis viritetty manuaalisesti arvoja muuttamalla ja tutkimalla miten robotin käytös muuttuu.

### 1.3. Robotin rakenne

Ensiksi rakenteesta on todettava, jos lukija haluaa itse rakentaa vastaanlainen robotin, suosittelen ehdottomasti rakentamaan sen ensin ilman NXT yksikön päällä olevaa kolmatta moottoria ja testaamaan ja virittämään robotin toiminnan mahdollisimman hyväksi tällaisena. Katso myös tarkemmat rakennusohjeet tämän dokumentin viimeisestä kappaleesta.

Robotissa on käytetty vain *Lego Mindstorms NXT* ensimmäisen version osia. Robotti koostuu NXT yksiköstä jonka sivulle on liitetty kaksi moottoria. Moottoreihin on akseleilla kiinnitetty pyörät joiden halkaisija on noin 5,5 cm. Testailun perusteella renkaat on kiinnitetty moottorien väliin. Renkaat voisi kiinnittää myös robotin ulkosivulle. Lisäksi NXT yksikön päälle on kiinnitetty hätipysäytyskytkin joka on NXT kosketusanturi. Valoanturi on sijoitettu robotin alle kahden poikittaistuen avulla.

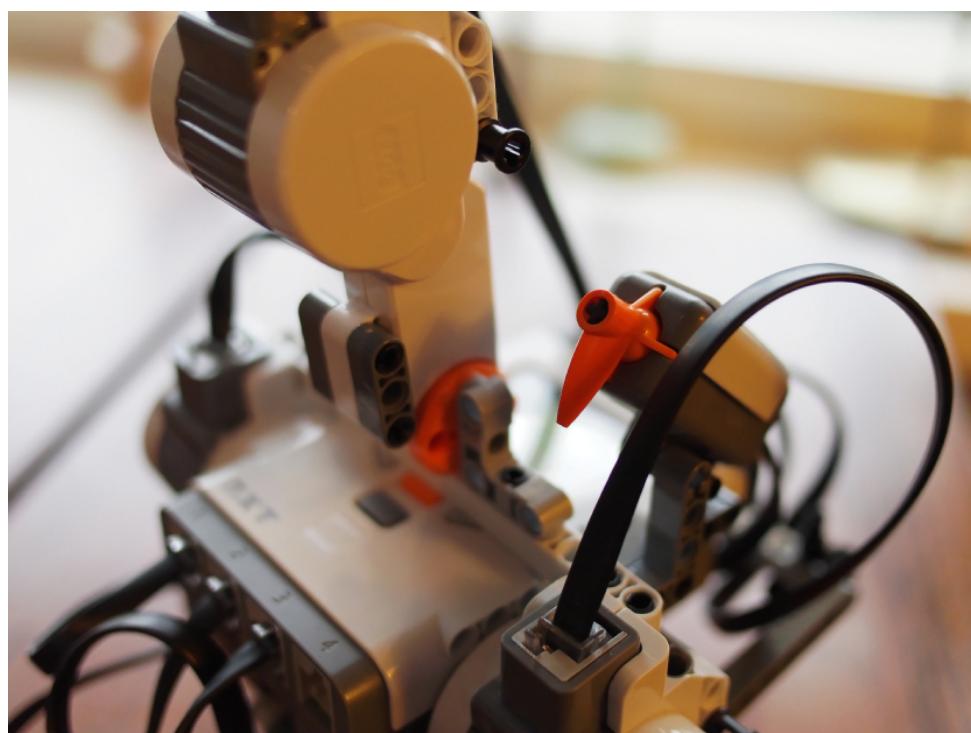
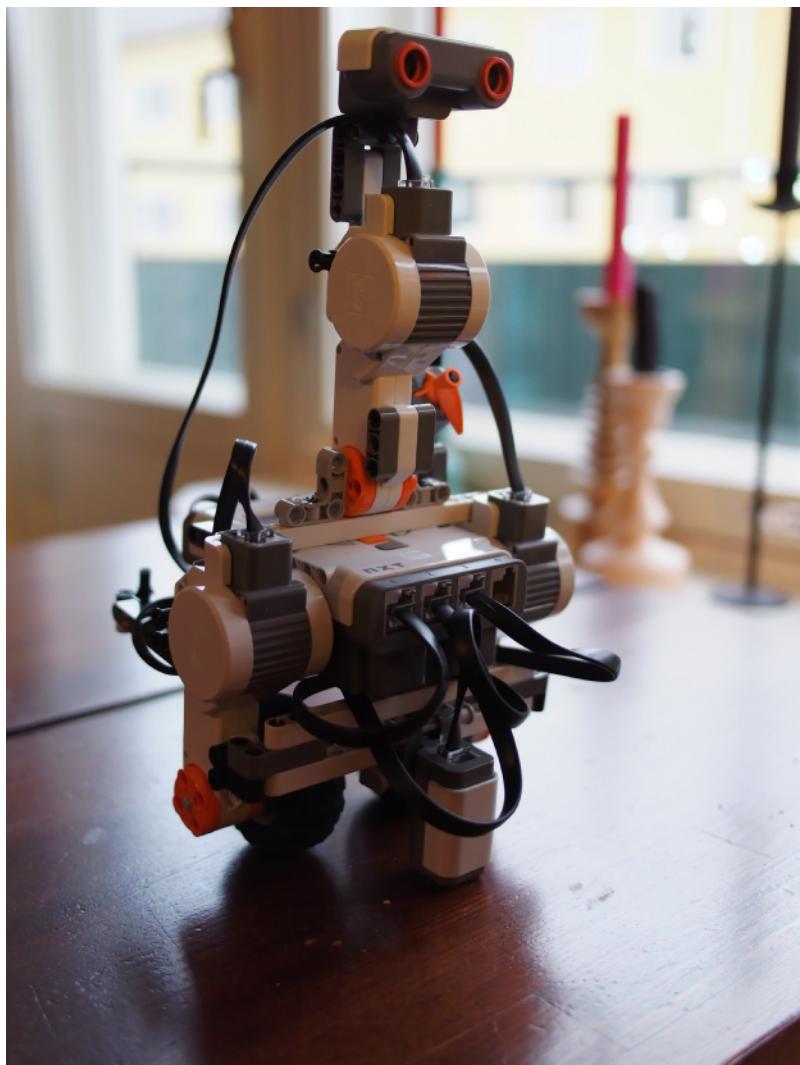
Robotin maksimi mitat ylhäältä pään katsottuna on 15 x 15 cm. Robotin suurin korkeus ilman kolmatta moottoria ja ultraäänisensoria on 17 cm. Ultraäänisensori ja kolmas moottori lisää robotin korkeutta noin 13 cm.

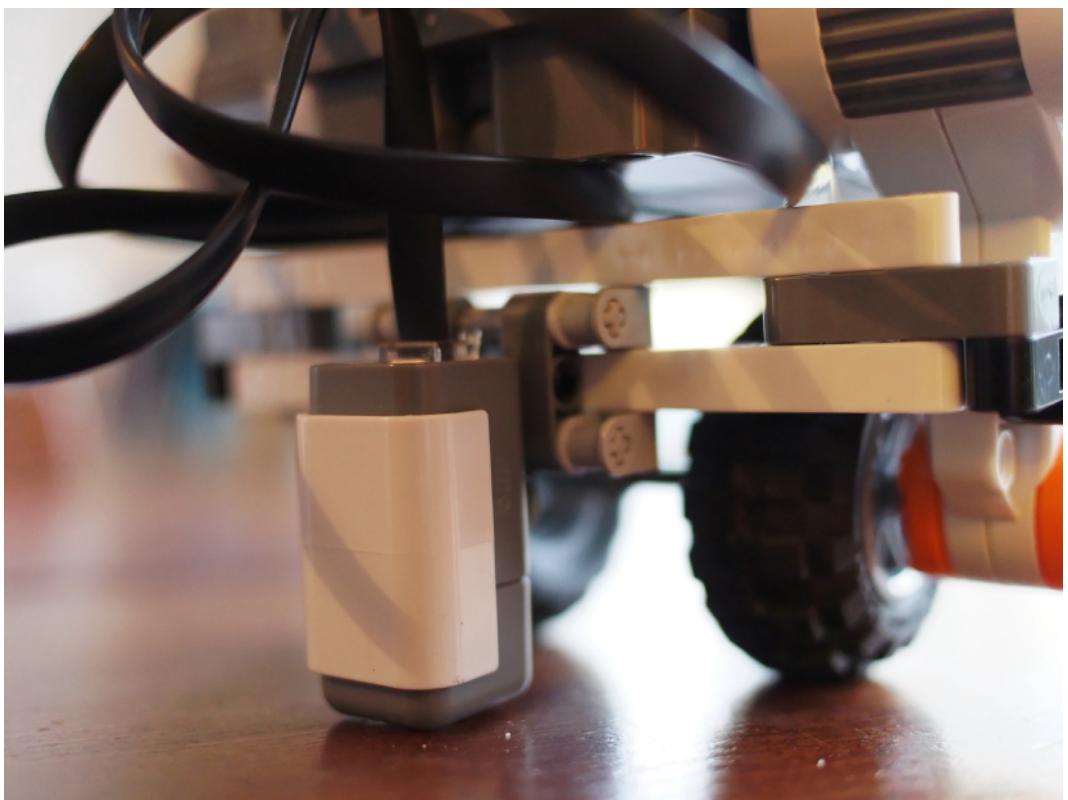
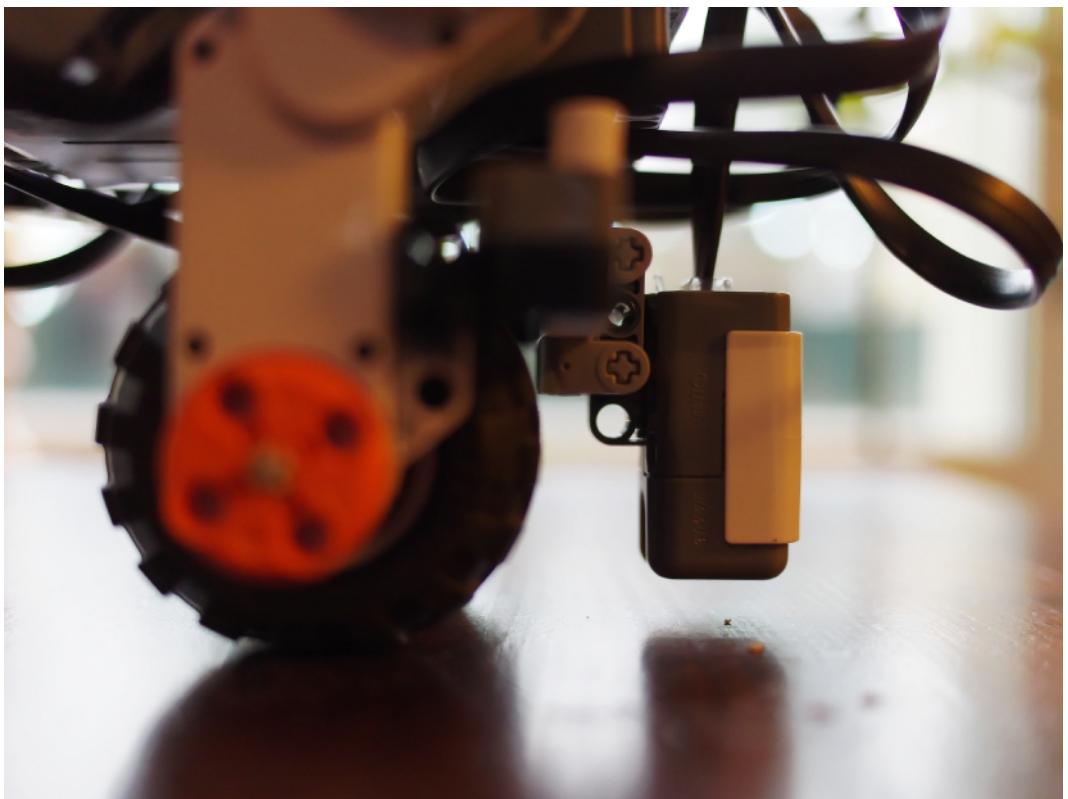
Liitosjohdot on viety kaikki robotin alta. Patterikotelon aukaisua varten täytyy valoanturin poikittaistuki irrottaa jommalta kummalta puoleltta.

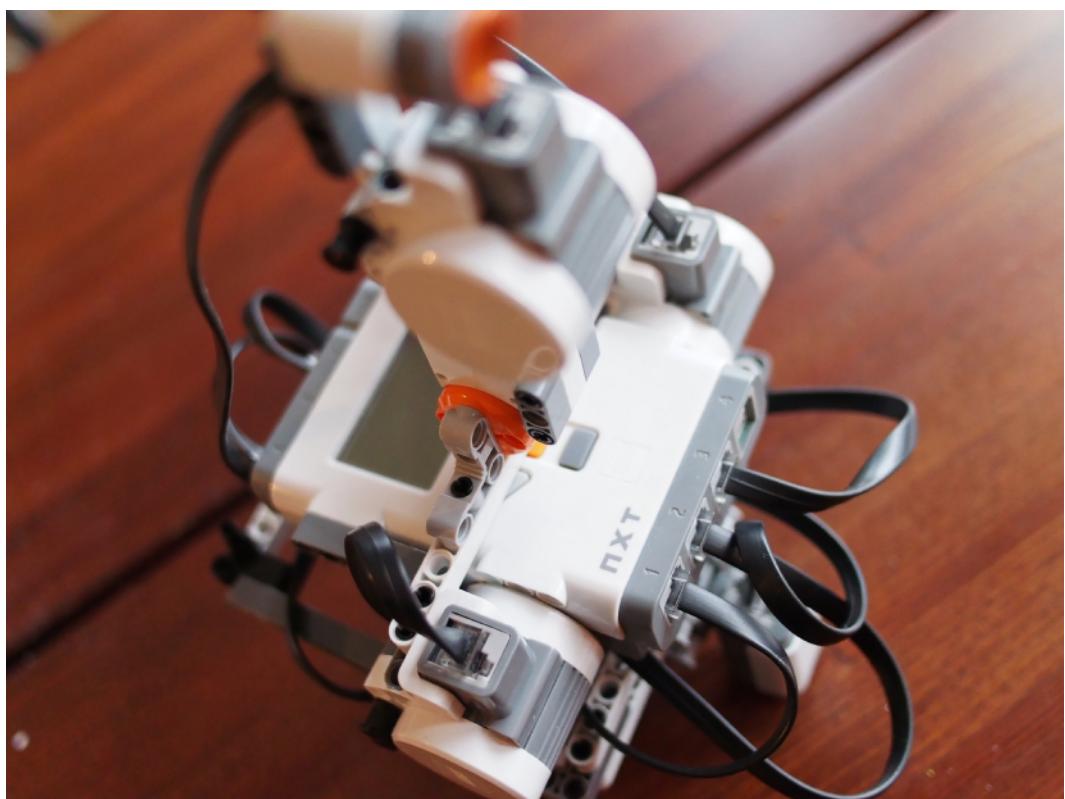
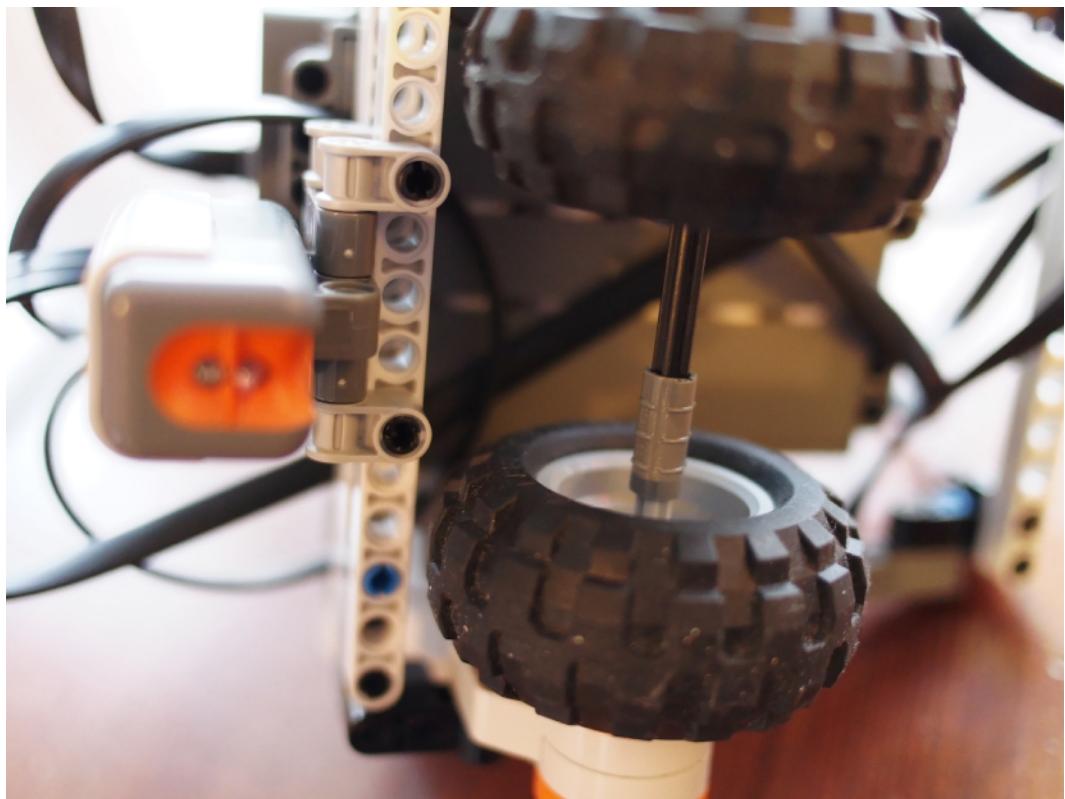
Lisäksi robotin renkaat on yhdistetty aksellilla toisiinsa. Tämä vähentää huomattavasti sivuttaisoskillaatiota.

Robotin taakse on rakennettu suojavaari suojaamaan moottorin kiinnitysjohtojen liitoksia (A-C).

Alla on esitetty kuvia robotista täysin koottuna, kolmas moottori kiinnitetynä ultraäänisensorin kanssa NXT yksikön päälle. On kuitenkin huomattavaa, kuten tämän kappaleena alussakin todettiin, että robottia saattaa olla vaikea saada toimimaan luotettavasti ensimmäisen version valosensorilla jos paino piste on hyvin korkealla. Tämän vuoksi on suositeltavaa rakentaa robotti ainakin ensin ilman kolmatta moottoria ja ultraäänisensoria.







## 2. Koodi

### 2.1. Ohjelmointiympäristö

Robotti on koodattu Java -kielellä. NXT ohjelmointiyksikköön on vaihdettu javaa ymmärtävä *Lejos* -firmware.

### 2.2. Koodin rakenne

Koodi koostuu yhdestä pääluokasta *BalanssiBotti*. Robotti käynnistyy normaalisti main metodista:

```
public static void main(String[] args) {  
  
    BalanssiBotti bot = new BalanssiBotti();  
  
    bot.haeBalanssi();  
    bot.pidKontrolli();  
    bot.sammuta();  
  
}
```

Robotin tasapainopiste haetaan ensin manuaalisesti, sillä tarvitsemme PID-säädintä varten asetusarvon. Ohjelma käynnistyy siis `haeBalanssi()`-metodilla. Robottia tasapainotetaan käsin ja haetaan robotille asetusarvo: `offset = valo.readNormalizedValue()`. Arvo vahvistetaan painamalla NXT yksikön oranssia nappulaa. Toisin sanoen, pyörimme while loopissa ja luetaan arvoja (ja kirjoitetaan ne näytölle) kunnes nappia painetaan: `while (!Button.ENTER.isPressed())`

Kun offset arvo on asetettu käynnistyy robotin pääasiallinen toiminta, `pidKontrolli()` -metodi, eli tasapainoilu. PID-säätimen kannalta oleellisia olivat seuraavat parametrit:

```
// 1. Proportional gain  
KP = 25;  
// 2. Integral gain  
KI = 5;  
// 3. Derivative gain  
KD = 35;  
// PID skaalaus  
SCALE = 16;
```

Valosensorilta luetaan arvoja: `int normVal = valo.readNormalizedValue()` ja verrataan niitä `haeBalanssi()`-metodissa asetettuun offset arvoon: `int error = normVal - offset.`

PID-säätimen arvo saadaan lopulta seuraavasti:

```
int pid_val = (int)(KP * error + KI * int_error + KD *
```

```
deriv_error) /SCALE;
```

Moottoreita pyöritetään `pid_val` arvon mukaisesti. Jos arvo on suurempi kuin nolla niin moottoreita pyöritetään eteenpäin ja muuten taaksepäin.

Koko metodi pyörii while loopissa `while (!kytkin.isPressed())` eli kun hätäkytkintä painetaan poistutaan loopista ja `pidKontrolli()` metodista. Sen jälkeen robotti sammuu suorittaen `sammuta()`-metodin jossa sammutetaan moottorit `Motor.X.flt()`, missä X on moottorin kirjain, ja valoanturi sammutetaan myös `valo.setFloodlight(false)`. Ohjelman suoritus on päättynyt.

### 3. Testaus

Kaikki testaus liittyy robotin tasapainoilun optimointiin, sillä se on robotin tärkein ominaisuus.

#### 3.1. Testitapaus 1

Robotin toiminnan kannalta on keskeistä millaisella alustalla robotti käytetään sillä valosensorin antaman datan virheettömyys ja johdonmukaisuus on toiminnan kannalta oleellisia.

Testailin erilaisia alustoja kiiltävistä pinnoista aivan mattapintaisiin. Lisäksi tummien sävyjen ja vaaleiden sävyjen välillä tehtiin testausta. Kiiltävät pinnat soveltuivat robotin alustaksi kaikista huonoimmin, robotti ei onnistunut tasapainoilemaan ollenkaan tällaisilla pinnoilla. Ilmeisesti valoa heijastuu liikaa takaisinpäin ja valosensori ei pysty toimimaan.

Parhaaksi alustaksi osoittautui mahdollisimman vaalea ja mattapintainen alusta. Tällainen on esimerkiksi tavallinen tulostuspaperi. Kun testasin mustan ja valkoisen mattapintaisen alustan eroa oli valosensorin lukemissa suuria eroja. Valosensori antoi 1,5 kertaisesti suurempia lukemia vaalealla alustalla.

Lopuksi testattiin tavallista ruskeaa pahvia. Pahvi tuntui myös toimivan oikein mainiosti.

Alustan testauksesta nousi esiin keskeisin asia; PID-säädin täytyy optimoida jokaiselle alustalle erikseen. Robotti toimii eri tavalla erilaisilla alustoilla. Esimerkiksi pahvi voi toimia oikein mainiosti, mutta jos robotti on optimoitu toimimaan valkoisen tulostuspaperin päälle, ei samat asetukset tuota hyvää loppitulosta pahvin päällä.

### **3.2. Testitapaus 2**

Robotin toiminnan optimoimiseksi testattiin pahvisen tuubin vaikutusta valosensorin ympärillä. Rakensiin pahvista valosensorin ympärille pujotettavan valosuojan. Tämän ideana oli estää ulkopäin tulevan valon vaikutus sensorin toimintaan. Tämä paransi testituloksia, mutta toisen robottiversion kanssa valosensori asettui niin lähelle maata, että pahvisesta tuubista ei ollut hyötyä.

Robotin rakentajalle vinkiksi; jos robottisi valosensori on kaukana maasta, yli 1 cm kun robottisi NXT yksikkö on vaakasuorassa, suosittelemat valosuojan rakentamissa. Muussa tapauksessa robotin oskillaation vuoksi sellaista ei kannata rakentaa.

### **3.3. Testitapaus 3**

Koodin osalta testasin valosensorin antamien lukemien korjaamista niin, että jos valosensorilta saatava arvo on välillä [0,3] niin virhe asetetaan nollaksi. Tämän testin tarkoituksesta oli tutkia voiko robotin oskillaatiota vähentää itse koodissa.

Testissä havaittiin, että robotin oskillaatio kyllä väheni selvästi, mutta robotti ei pystynyt enää reagoimaan tasapainon muutokseen tarpeeksi nopeasti. Jos virheen korjaus rajattiin välillä [0,1] -> 0 ei robotin toiminnassa voitu havaita silmin nähtävää muutosta.

Tämä virheenkorjaus poistettiin testauksen tuloksena lopullisesta versiosta.

## **4. Robotin jatkokehitys ja ongelmat**

Robotin fyysisessä rakenteessa on kehitettävää. Suurempien pyörien vaihtaminen pienien tilalle parantaisi varmasti robotin suorituskykyä. Uudemmassa toisen version NXT sarjassa valosensorin korvaa värisensori, joka luultavammin toimisi huomattavasti paremmin ja luotettavammin balanssibotin kanssa.

Jos käytettävissä on gyroskooppi, on sellaisen käyttäminen ehdottoman suositeltavaa. Robotti kannattaa rakentaa tällöin mieluiten gyroskoopin datan varaan ennenmin kuin valosensorin.

Robotin ultraäänisensorin voisi tulevaisuudessa ottaa käyttöön niin, että havaitessaan esteen liikkuessaan eteenpäin, kallistaa kolmatta NXT yksikön

päällä olevaa moottoria taaksepäin siten, että liike pysähtyy. Näin robottiin saa myös väistö ominaisuuden.

Robotin painopisteen merkitys toiminnan kannalta on siis oleellinen. Tulevaisuudessa voisi kokeilla konstruktioita jossa NXT yksikkö on pystyssä aivan lattian rajassa ja pyörät ovat NXT yksikön sivuilla. Näin painopisteen saisi aivan alas. Uudenlaisessa konstruktiossa tosin kolmannen moottorin sijoittaminen painopisteen siirtoa varten saattaa osoittautua hankalaksi.

Yksi tulevaisuudessa toteutettavissa oleva, mutta erittäin haastava ongelma on, miten saada balanssibotti kiertämään ympyrää. Toisin sanoen miten balanssibotin saisi käänymään oikealle tai vasemmalle.

Koodin puolesta PID-säädintä voi optimoida vaikka koko loppuelämän ajan. Yksiselitteisiä arvoja PID parametreille ei ole olemassa. Jokaisen parametrin hienosäätökin voi vaikuttaa ratkaisevasti toiseen parametriin. Keskeistä olisi saada robotti reagoimaan tasapainon muutokseen mahdollisimman nopeasti, mutta pitää muutoksen voimakkuus sellaisena ettei robotti korjaa virhettä liiaksi. Tällöin robotin oskillaatio kasvaa liian voimakkaaksi eikä tasapainoilu ole mahdollista. Itse keskityin lopulta ehkä liiaksikin PID säätimen optimointiin.

Lopuksi vinkiksi rakentajalle, joka suunnittelee balansoivan robotin rakentamista, robottia ei ehkä ole kannattavaa rakentaa ensimmäisen version sarjasta. On todella haastavaa saada robottia tasapainoilemaan luotettavasti muutamaa sekuntia pidempään pelkän valosensorin avulla. Toisen sarjan värisensori korjaa tilannetta jo huomattavasti, puhumattakaan gyroskoopista.

## 5. Käyttö- ja rakennusohje

### 5.1. Balanssibotin käyttö

Rakenna robotti seuraavassa 5.2. alakohdassa esitettyjen ohjeiden mukaisesti. Suosittelen käyttämään Jouko Strömmelin kasaamaa Linux distroa, jolta löytyy kaikki tarvittava NXT:n ohjelointiin. Lisäksi asenna NXT yksikköön uusin Lejos firmware. Ohjeita tähän löydät Lejos.org sivustolta. Ohjelointi kannattaa suorittaa Eclipseä käyttäen. Tällöin kun olet yhdistänyt NXT yksikön usb-johdolla tietokoneeseen, voit yksinkertaisesti ajaa build.xml tiedoston.

1. Ohjelma käynnistyy BalanssiBotti -tiedostosta. Voit valita sen myös NXT yksiköstä selaamalla vasemmalle tai oikealle valikkoon "files". Täältä löydät BalanssiBotti -tiedoston. Suorita tiedosto ja ohjelma käynnistyy.
2. Kun valosensoriin syttyy valo tiedät, että ohjelma on käynnissä. NXT yksikön ruudulla näkyy nyt myös valosensorin antama arvo. Robotti

täytyy nyt asettaa tasapainoon. Hae käsin tunnustelemalla robottille tasapainoinen aloituspiste. Huomaa, että tämä on ensiarvoisen tärkeää. Jos robotti ei ole alkuasennossa tasapainossa, ei robotti pysty tasapainoilemaan luotettavasti. Kun olet saanut robotin tasapainoon (huomaa, että voit käyttää valosensorin lukemaa avuksi) paina NXT yksikön oranssia keskinäppääin. Tällöin tasapainoarvo lukittuu ja robotti aloittaa tasapainoilun.

3. Ohjelma on käynnissä ja robotti tasapainoilee.
4. Sammuta robotti NXT yksikön päällä olevasta hätäpysäytyskytkimestä.<<<z<<

## 5.2. Balanssibotin rakennusohje



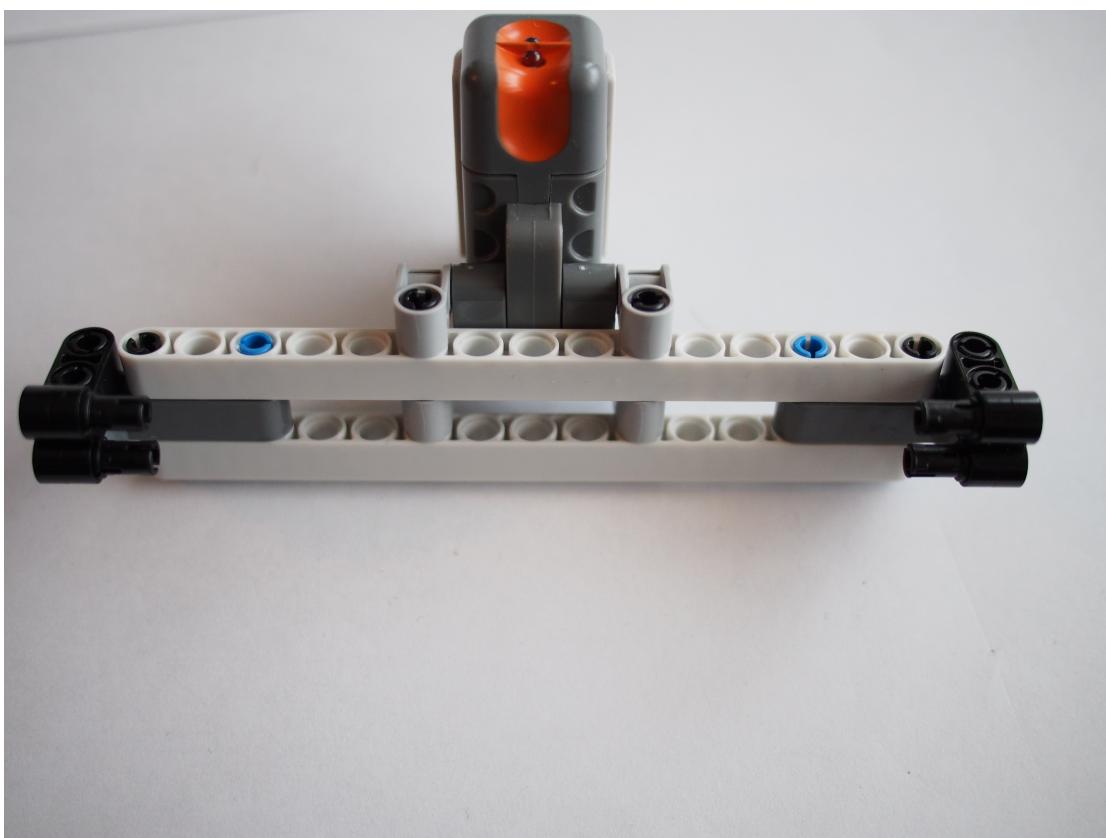
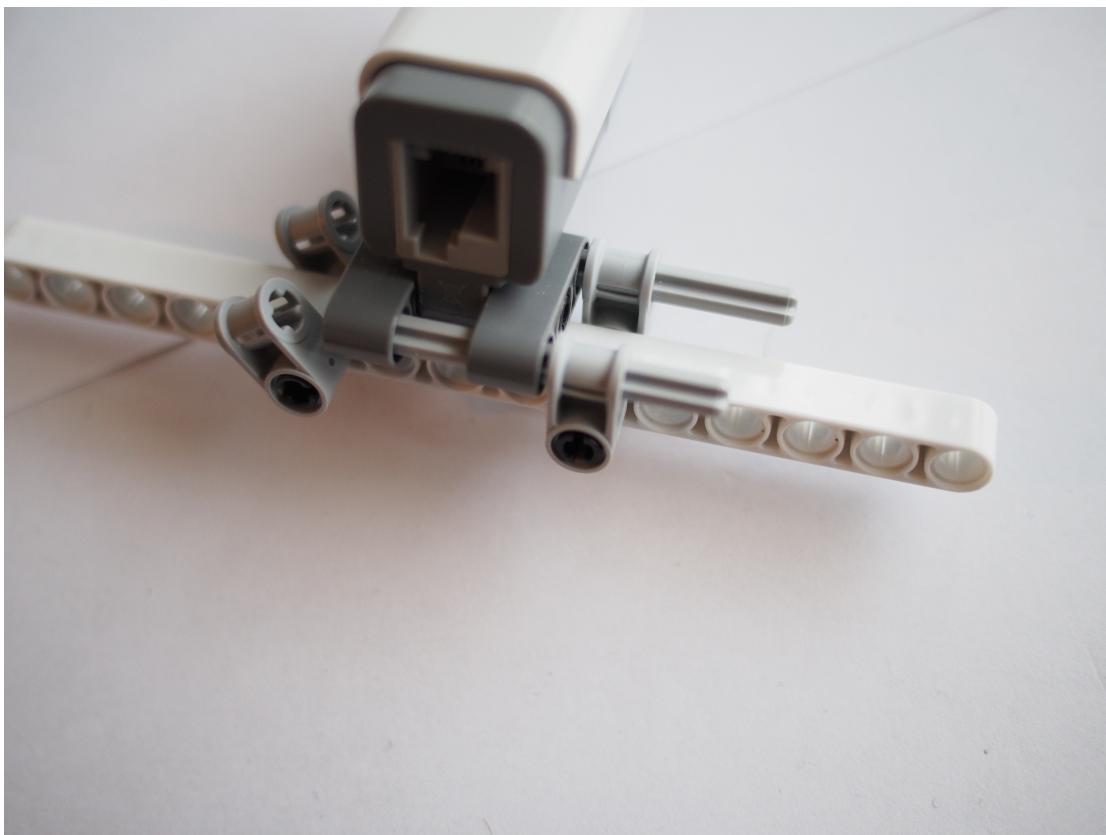
Kiinnitetään moottoreihin tuet seuraavasti.



Moottorit kiinnitetään NXT yksikön sivuille kolmen kiinnityspinnin avulla.



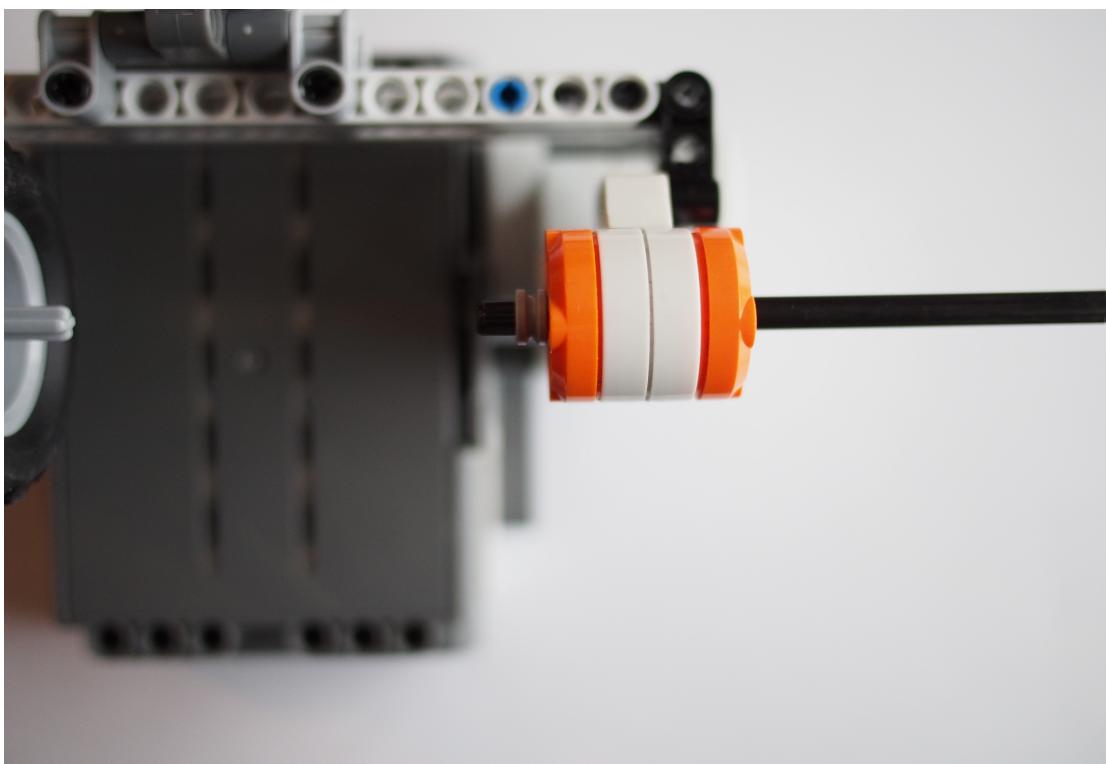
NXT yksikön päälle on hyvä asentaa sivuttaistuki. Toisaalta nyt näppäimiä on painattava lyhyen akselin avulla, mutta tämä ei ole suuri haitta.



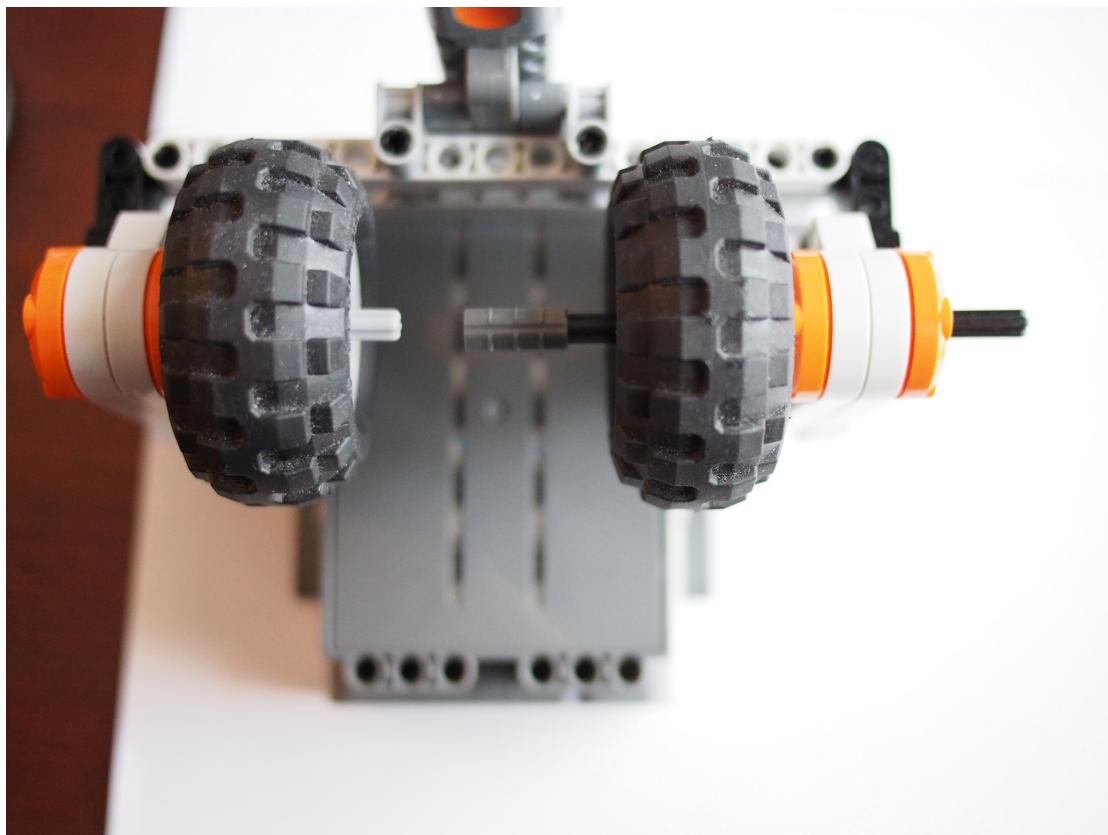
Kiinnitä valosensoriin kaksi poikittaistukea yllä olevien kuvien mukaisesti.



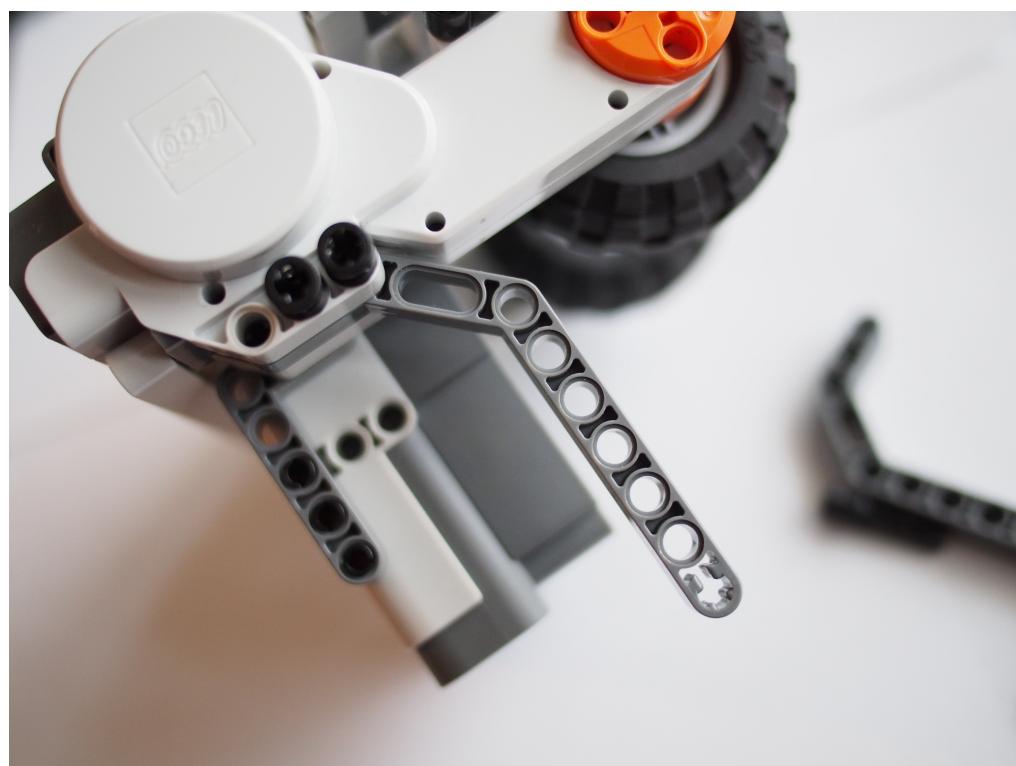
Valosensori on nyt valmiina kiinnitettäväksi moottorien sivuille.



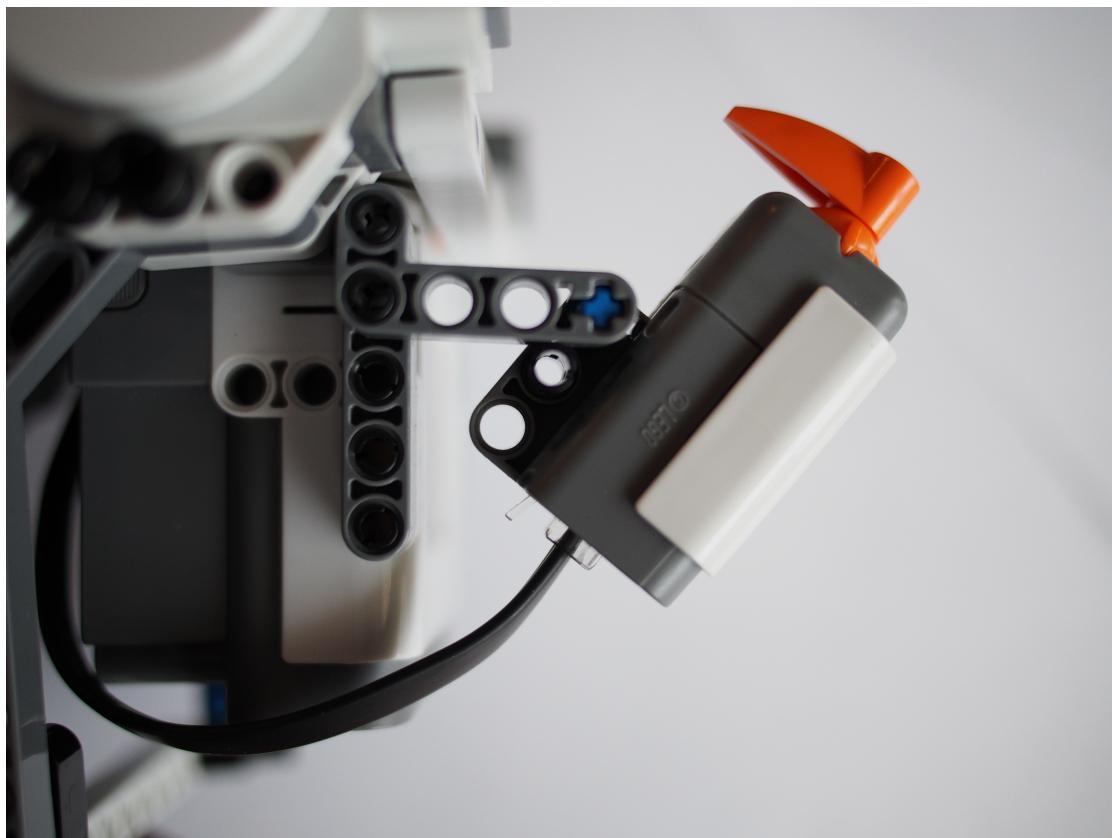
Akselia asentaessasi asenna renkaan ja moottorin väliin välikepala. Nämä  
Rengas ei ota kiinni moottoriin.



Yhdistää akselit. Voit kokeilla robottia myös ilman yhdysakselia.



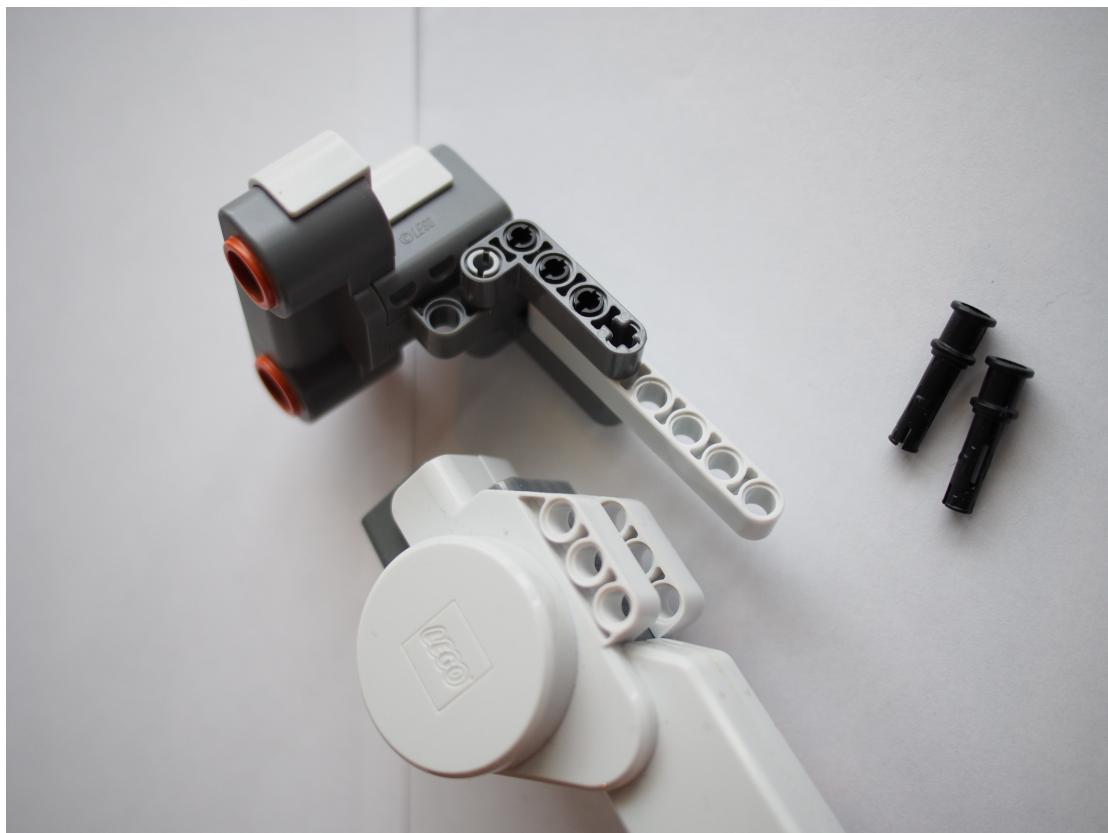
Kiinnitetään moottorien kiinnitysjohdonsuojat. Tämä parantaa myös tasapainopistettä. Kiinnitä samanlainen toiselle puolelle ja yhdistä poikittaistuella.



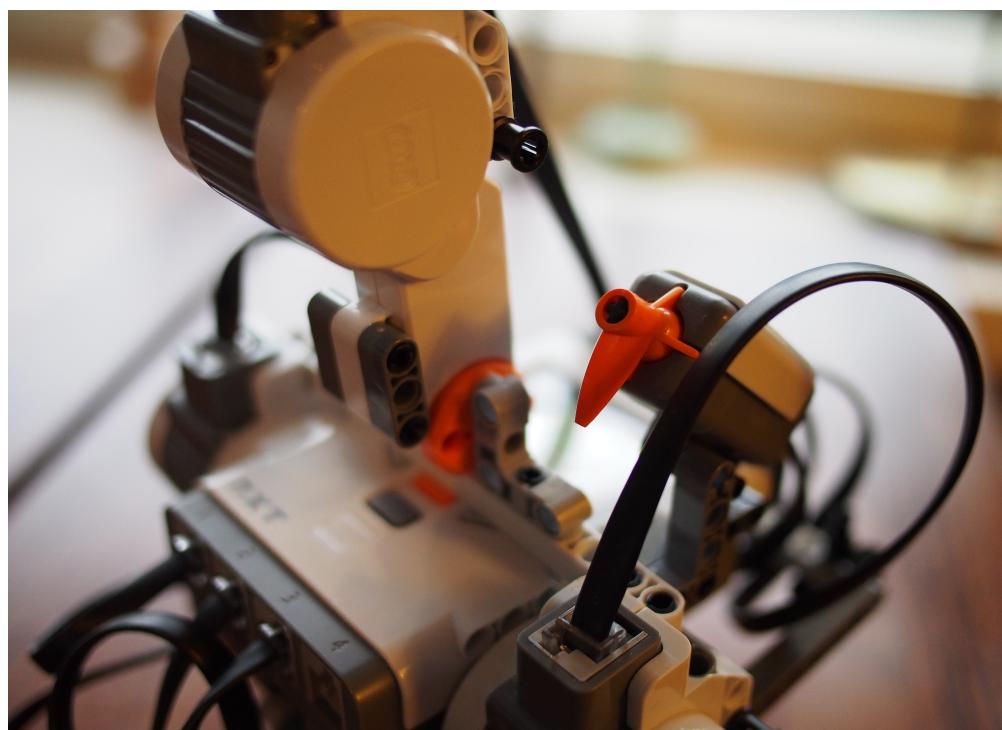
Hätäkytkin on asennettu vinoon jotta sitä olisi helpompi käyttää etupuolelta.



NXT yksikön päälle tulevan kolmannen moottorin tuenta NXT yksikön päälle.



Kiinnitetään ultraääni sensori kahden pitkän pinnin avulla moottoriin.



Yhdistä lopuksi kaikki johdot vieden ne NXT yksikön alta. Moottorit ovat porteissa A ja C. Hätkäytkin on portissa 2 ja valosensori portissa 1. Robotti on nyt valmis.