

KOSKETUSNÄYTÖLLISEN MATKAPUHELIMEN TUNTOPALAUTETOIMINNON TESTAUS

Heikki Ahvensalmi
Opinnäytetyö
2.11.2010
Kone- ja tuotantotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

Koulutusohjelma	Opinnäytetyö	Sivuja	+	Liitteitä
Kone- ja tuotantotekniikka	Insinöörityö	42	+	7
Suuntautumisvaihtoehto	Aika			
Energiatekniikka	2010			
Työn tilaaja	Työn tekijä			
Nokia Oyj	Heikki Ahvensalmi			
Työn nimi				
Kosketusnäytöllisen matkapuhelimen tuntopalautustoiminnon testaus				
Avainsanat				
Matkapuhelin, kosketus, kiihtyvyys, Gage R&R				

Matkapuhelimen tuntopalautustoiminto antaa käyttäjälle palautteen kosketuksen perille menosta. Mittalaite pyrkii mittaamaan tämän palautteen voimakkuuden ja erottamaan hyvän testikappaleet huonosta. Mittaus tapahtuu kiihtyvyyssanturia hyväksi käyttäen. Työn keskeisin osuus on testilaitteen tarkkuuden ja toistettavuuden arviointi.

Tulosten analysointiin käytettiin Gage R&R:ää. Sen avulla saadaan selville mittalaitteen mittavirhe. Kaksi eri operaattoria (käyttäjää) mittasi testattavat kappaleet kolme kertaa. Mittausjärjestys oli arvottu, ja operaattorit suorittivat mittaukset annettujen ohjeiden mukaisesti. Tulosten lukemiseen käytettiin kahta tapaa, oskilloskooppia ja Arduino-mikrokontrolleria. Näitä kahta tapaa vertailtiin keskenään.

Lopputuloksena saatiin toleranssirajat, jotka mittalaite pystyy havaitsemaan. Tuloksista käy myös ilmi, että Arduino-mikrokontrolleri on tarkempi ja helpokäyttöisempi kuin oskilloskooppi.

SISÄLTÖ

TIIVISTELMÄ

SISÄLTÖ

1 JOHDANTO	6
2 KIIHTYVYYSANTURI	7
2.1 Kiihtyvyys.....	7
2.2 Toimintaperiaatteet.....	7
2.3 Mahdollisuudet	8
3 GAGE R&R.....	9
3.1 Prosessin hajonta (repeatability)	9
3.2 Prosessin toistettavuus (reproducibility)	10
3.3 Mittausvaihtelu	11
3.4 %-toleranssi	11
3.5 Luottamusväli	13
3.6 Normaalijakauma	13
4 ARDUINO.....	14
4.1 Ominaisuudet	14
4.2 Ohjelmisto.....	14
5 MITTAUSLAITTEEN RAKENNE	16
5.1 Jigi	16
5.2 Robotti.....	17
5.3 Linearijohde ja kiihtyvyyssanturi	20
5.4 Oskilloskooppi	22
5.5 Arduinon ja testeri.....	23
5.6 Arduinon ohjelmointi	24
6 MITTAUSTAPA	27
6.1 Periaate	27
6.2 Mittausohjeet	29
7 TULOKSET JA ANALYSOINTI	30
7.1 Oskilloskooppi	30
7.2 Arduino.....	32
7.3 Oskilloskooppi tulosten analysointi	33

7.4 Arduino tulosten analysointi.....	34
8 LOPPUTULOKSET	35
9 POHDINTA.....	38
LÄHTEET.....	40
LIITEET	42

1 JOHDANTO

Työssä määritellään kosketusnäyttöpuhelimissa käytettävän tuntopalautusominaisuuden mittaamiseen kehitetyn mittalaitteen tarkkuutta, toistettavuutta ja rakennetta. Myös kehitys- ja mahdollisuustapoja on tuotu esiin. Toimeksiantajana on Nokia Oyj, ja mittalaite on käytössä Nokian Peltolan yksikössä.

Testilaitteella tehtiin koemittauksia, jotta tarkkuutta voitaisiin selvittää. Mittaustapana käytettiin Gage R&R:ää. Gage R&R antaa tuloksia mittauslaitteen luotettavuudesta. Sen avulla voidaan selvittää, johtuuko laitteen mittavirhe operaattoreista (käyttäjistä) vai onko itse mittalaitteen mittavirhe liian suurta.

Mittalaitteen rakenne tehtiin mahdollisimman yksinkertaiseksi, koska aikataulu oli tiukka. Mittalaitteen osina käytettiin valmiiksi löytyvää laitteistoa mahdollisimman paljon. Laitteen tärkein osa on kiihtyvyyssanturi, joka sijaitsee teollisuusrobottiin kiinnitetyn lineaarijohteen päässä.

Työssä ei salassapitovelvollisuuden vuoksi ilmoiteta, mitä tuotetta mittauksissa on käytetty. Nokialle on tehty testilaitteesta oma dokumentti, joka sisältää salassapidettävät asiat.

Gage R&R mittaukset tehtiin kaksi kertaa. Mittauksen suorittivat kaksi eri operaattoria (käyttäjää). Testipuhelimia oli kymmenen kappaletta, jotka kukin operaattori mittasi kolme kertaa.

2 KIIHTYVYYSSANTURI

2.1 Kiihtyvyys

Kiihtyvyys kuvaa kappaleen nopeuden muutosta tietyssä ajassa (1, s. 7.). Kiihtyvyyden tunnus on a ja yksikkö m/s^2 . Painovoiman putoamiskiihtyvyyden tunnus on g , ja sen likiarvona käytetään $9,81 \text{ m/s}^2$.

Kappaleen nopeuden muutos voi olla joko kiihtyvää, hidastuvaa tai suunnan muutosta. Jos kappale on vakionopeudessa, sen kiihtyvyys on nolla. Kiihtyvyys on vektorisuure, ja kappaleen kiihtyvyys määritellään nopeuden ensimmäisenä derivaattana ajan suhteen (kaava 1). (2.)

$$a = \frac{dv}{dt} \quad \text{KAAVA 1}$$

a = kiihtyvyys

t = aika

v = nopeus

2.2 Toimintaperiaatteet

Kiihtyvyysanturi on sähkömekaaninen laite, joka mittaa kiihtyvyyden voimia. Nämä voimat voivat olla joko staattisia tai dynaamisia. Anturin perusosia ovat runko, jousi ja massa. Anturi mittaa tunnettuun massaan kohdistuvan liikkeen muutoksen ja esittää sen esimerkiksi sähköisessä muodossa, minkä voi tulkita oskilloskoopilla. Kiihtyvyyden yksikkö on m/s^2 , ja kappaleeseen vaikuttavan voima F antaa kiihtyvyyden a (kaava 2). (3.)

$$F = ma \quad \text{KAAVA 2}$$

$F = \text{voima}$

$m = \text{massa}$

$a = \text{kiihtyvyys}$

On olemassa erilaisia tapoja tehdä kiihtyvyysanturi. Jotkin kiihtyvyysanturit käyttävät hyväksi pietsosähköistä vaikutusta. Pietsosähköinen anturi sisältää mikroskooppisen pieniä kiteitä, jotka reagoivat muodonmuutoksiin. Muodonmuutos synnyttää kiteisiin sähkövarauksen, minkä kautta voidaan saada selville kiteeseen kohdistuva voima. (1, s. 9.)

Toinen tapa on aistia kapasitanssin muutos. Kahden mikrorakenteen välillä on tietty kapasitanssi. Jos toinen liikkuu voiman vaikutuksesta, kapasitanssi muuttuu. Kapasitanssin muutos voidaan esimerkiksi siltakytkenällä muuttaa sähköksi ja lukea. (1, s. 10.)

2.3 Mahdollisuudet

Kiihtyvyysantureilla on laaja käyttömahdollisuus esimerkiksi autojen turvatyynyissä ja askelmittareissa. Kiihtyvyysanturia käytetään matkapuhelimissa tunnistamaan esimerkiksi puhelimen asento, jolloin puhelimen tekstit kääntyvät vaakapystysuunnassa huomioiden sen käyttösuunnan. Lääketeollisuus on ottanut myös kiihtyvyysanturit käyttöönsä sydämentahdistimissa (1, s. 12.). Peliteollisuus käyttää kiihtyvyysantureita peliohjaimissa, joiden avulla peleistä, ja etenkin ohjaamisesta, saadaan realistisempaa.

Kiihtyvyysantureita voi myös käyttää tulevaisuudessa esimerkiksi kaukosäätiminä. Ranteessa kiinni ollessaan anturilla voidaan antaa käskyjä erilaisilla pienillä eleillä, esimerkiksi televisiokanavan vaihto onnistuisi jollakin tietyllä liikkeellä.

Koska kiihtyvyysanturi saadaan pieneen tilaan, on sen käyttömahdollisuudet laajat. Vanhojen keksintöjen ominaisuuksia voidaan parantaa kiihtyvyysantureilla tai kehittää kokonaan uusi tapa hyödyntää liikkeentunnistusta.

3 GAGE R&R

Gage R&R (Gage Repeatability and Reproducibility) on tilastollinen työkalu. Gage R&R:n avulla selvitetään mittalaitteen mittavirhe. Mittalaitteen vaihtelu voi johtua joko operaattorista (käyttäjä) tai laitteiston omasta mittavirheestä. Operaattorista johtuva vaihtelu voi aiheutua esimerkiksi siitä, että jokainen operaattori mittaa asian hieman erilailla. (4, s. 25.)

Tavoite on, että mittaustulosten vaihtelu johtuisi operaattorista tai osista, ja itse mittalaite olisi luotettava. Mittaustapahtuma täytyy olla hallinnassa, jotta tulokset olisivat luotettavia ja voidaan sanoa, ovatko mitatut osat hyviä vai huonoja (pass/fail). Tämä on tärkeää hyväksyttäessä esimerkiksi tuotteen osia. (5.)

Gage R&R koostuu kahdesta asiasta, prosessin toistokyvystä (repeatability) ja uusittavuudesta (reproducibility). Prosessin toistokyky käsittää tiedon siitä, kuinka paljon saman operaattorin samalla mittauslaitteella tehdyt mittaukset eroavat eri mittauskerroilla. Uusittavuus taas tarkoittaa mittausten eroja eri operaattorien välillä. (5.)

Tämän työn Gage R&R tutkimus tehtiin MiniTab-ohjelmistolla.

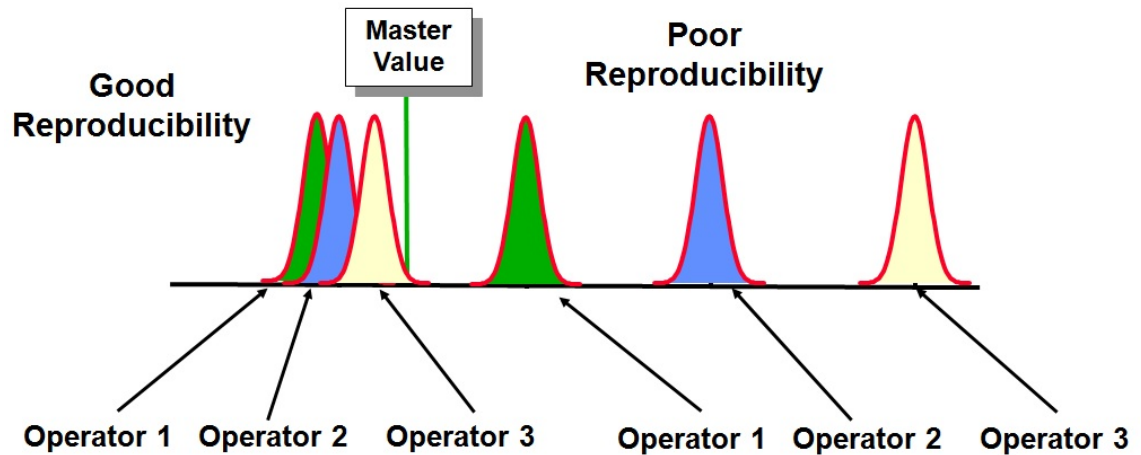
3.1 Prosessin hajonta (repeatability)

Jos prosessin hajonta (repeatability) on liian suuri, kannattaa tarkistaa:

- tarvitseeko mittauslaitteisto huoltoa
- onko jiggi tarpeeksi tarkka kappaleiden kohdistukseen toistettavasti
- onko sensorin kohdistus kunnossa
- onko mittauslaitteisto oikein valittu (6, s.14.).

3.2 Prosessin toistettavuus (reproducibility)

Jotta prosessiin saataisiin hyvä toistettavuus (reproducibility), pitäisi käyttäjien mittaushajonnat olla lähellä toisiaan (kuva 1).



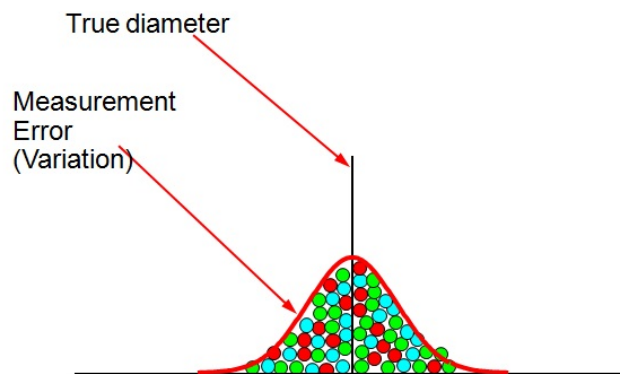
KUVA 1. Reproducibility (6, s. 13)

Jos toistettavuuden hajonta on liian suurta, kannattaa tarkistaa seuraavat asiat:

- onko operaattoreilla selkeät ohjeet, kuinka käyttää mittalaitetta
- käyttävätkö operaattorit mittalaitetta johdonmukaisesti
- tarvitsevatko operaattorit lisäkoulutusta laitteen käyttöön
- mittaavatko operaattorit kappaletta samasta paikasta joka kerta
- onko mittauspiste selkeästi määritelty esimerkiksi 2D kuvassa (6, s. 16.).

3.3 Mittausvaihtelu

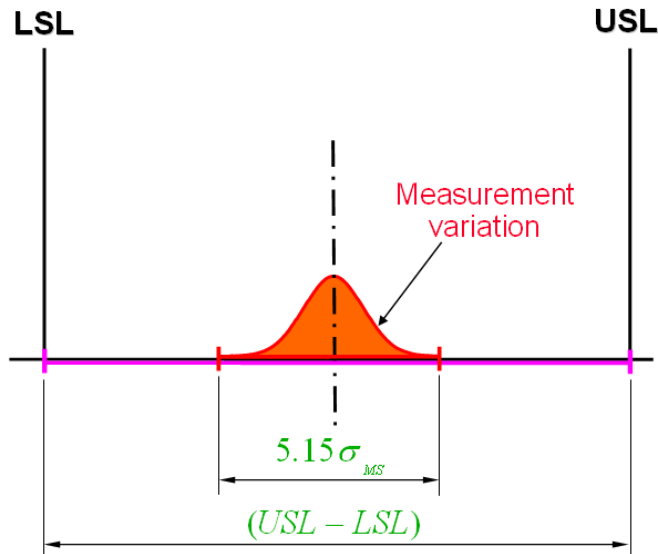
Mitattaessa samaa kappaletta useita kertoja voi operaattorien välillä tulla eroja. Esimerkiksi yksi operaattori mittaa kappaleen hieman eri paikasta kuin toinen tai tulkitsee mitan hiukan eri tavalla. Tästä syntyy mittausvaihtelua. Esimerkissä kolme operaattoria on mitannut tietyn kappaleen useita kertoja ja kuvaaja kertoo tulokset ja niiden vaihtelun (kuva 2). (6, s. 6.)



KUVA 2. Mittausvaihtelu operaattorien kesken (6, s. 6.)

3.4 %-toleranssi

Jotta saataisiin todellisia hyväksytyjä tai hylättyjä (pass or fail) tuloksia, täytyy mittausvaihtelun olla pieni verrattuna mitattavan kappaleen hyväksymisrajoihin. Tämä tarkoittaa %-toleranssia. Toleranssialue on välillä USL-LSL, jos halutaan verrata mittausvaihtelua vaihteluvälin rajoissa (kuva 3). (6, s. 25.)



KUVA 3. Prosenttiosuus toleranssi vaihtelusta verrattuna toleranssiväliin (6, s. 26)

Toleranssialue saadaan siirtämällä termi $USL - LSL$ vasemmalle puolelle yhtälöä ja jakamalla $5,15 \cdot \sigma_{MS}$ haluamalla %-toleranssilla (kaava 3).

%-toleranssin määrittelyssä käytetään perussääntönä seuraavia:

%-toleranssi ≤ 10 %	HYVÄ
10 % \leq %-toleranssi ≤ 30 %	MARGINAALINEN
30 % \leq %-toleranssi	HUONO.

$$\%Tolerance = \frac{5,15 \cdot \sigma_{MS}}{USL - LSL} \quad \text{KAAVA 3}$$

$5,15 \cdot \sigma_{MS}$ = olettaa, että 99,00% mittaustuloksista on normaalijakauman sisällä

σ_{MS} = Minitab-ohjelman antaman mittaushajonta (Total Gage R&R) (liite 1)

$USL - LSL$ = Toleranssialue, joka saadaan laskutoimituksesta. Antaa mitattavien kappaleiden valmistustoleranssialueen, jonka mittalaite pystyy vielä havaitsemaan.

3.5 Luottamusväli

Luottamusväli kertoo, mille välille perusjoukon tunnusluvun, esimerkiksi keskiarvon arvo, sijoittuu tietyllä todennäköisyydellä. Otoksesta valittuun tunnuslukuun, esimerkiksi keskiarvo tai toleranssialue, voidaan laskea esimerkiksi 95 prosentin virhemarginaalit (7.). Valittaessa todennäköisyydeksi 95 %, tarkoittaa, että on 5 %:n riski, ettei keskiarvo olekaan luottamusvälillä. Kaavalla 4 saadaan laskettua luottamusväli. (8, s. 37.)

$$E = \bar{x} \pm kr \cdot \frac{s}{\sqrt{n}}$$

KAAVA 4

3.6 Normaalijakauma

Normaalijakauma (kuva 3) on tilastotieteen tärkein teoreettinen jakauma (8, s. 34.). Sen muoto riippuu jakauman keskiarvosta eli odotusarvosta μ ja keskihajonnasta σ . Tällä kuvaajalla on keskellä huippu ja se supistuu reunoja kohden. Normaalijakauma kuvaa sitä, miten suuri pinta-ala mahtuu kullekin osalle tätä Gaussin käyrää (9.). Normaalijakauman tiheysfunktio $f(x)$ (kaava 5).

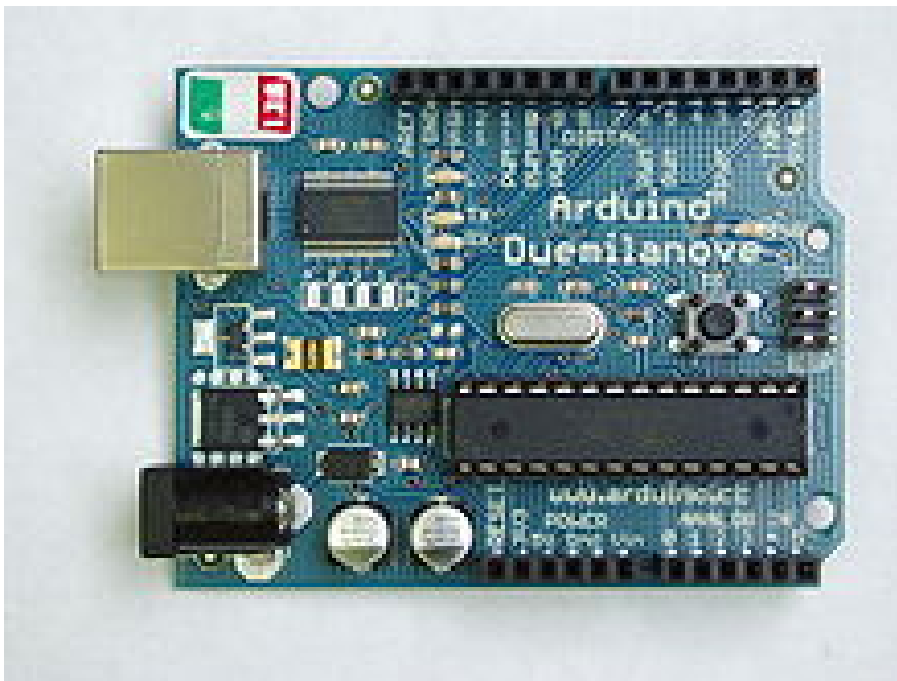
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

KAAVA 5

4 ARDUINO

4.1 Ominaisuudet

Arduino (kuva 4) on avoimeen lähdekoodiin ja suunnitteluun perustuva mikrokontrolleri, jonka ytimenä on ATmega328-mikropiiri. Arduinossa on 14 digitaalista IO-nastaa, 6 analogista sisääntuloa, 16 MHz kide, usb-liitäntä, virtaliitäntä, ICSP-liitin ja reset-painike. Arduinon ohjelmointikieli on pääasiassa C++. (10.)



KUVA 4. Arduino Duemilanove (12.)

4.2 Ohjelmisto

Arduinon käyttöohjelma (kuva 5) on Java-perustainen. Se on avoimeen lähdekoodiin perustuva. Ohjelmassa pystytään koodaamaan ohjelma, joka ladataan Arduinon muistiin. Mikrokontrolleri toimii tällöin ohjelman mukaisella tavalla. Käyttökielenä on pääasiassa C++. Eri käyttömahdollisuuksia on lukuisia led-valon ohjauksesta erilaisiin kiihtyvyyssanturisovelluksiin asti.



KUVA 5. Arduinon ohjelmisto

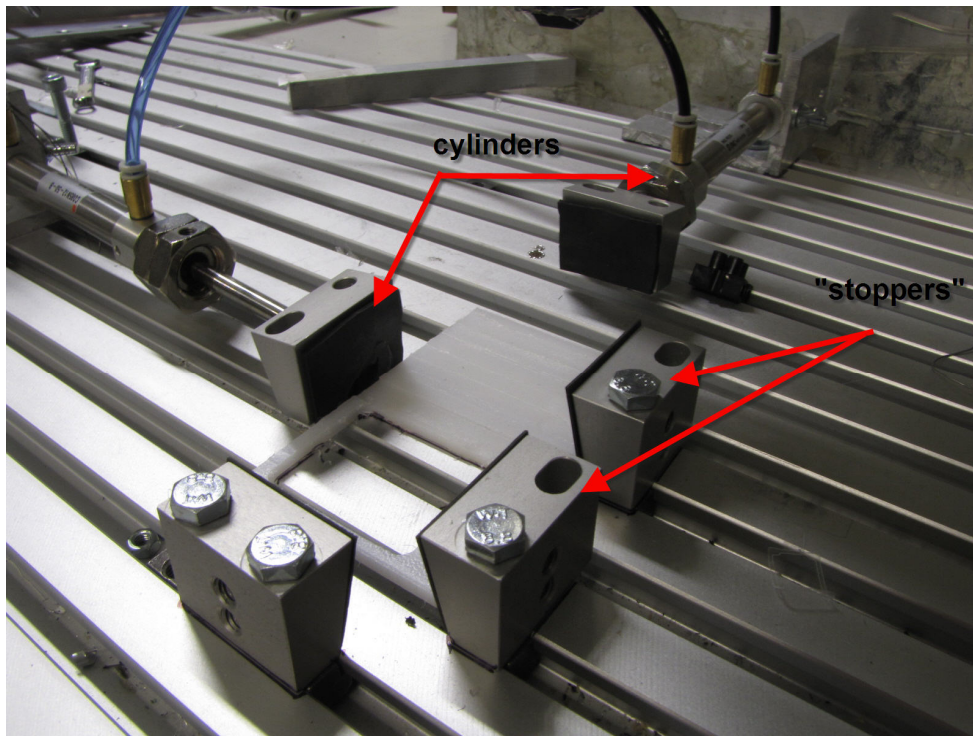
Internetistä on mahdollista etsiä valmiita ohjelmia tai käyttäjä voi tehdä oman ohjelman. Asennus on yksinkertainen, tarvitaan vain tietokone, ajurit ja ohjelmisto. Ohjelmisto on mahdollista ladata internetistä, josta löytyy myös ohjeita ja neuvoja ongelmatilanteisiin.

5 MITTAUSLAITTEEN RAKENNE

Tuntopalautustesterin rakenne sisältää jigien, lineaarijohteen, kiihtyvyyssanturin ja robotin. Tulosten lukemiseen käytettiin oskilloskooppia ja Arduino-mikrokontrolleria. Kummastakin luvutavasta on omat tuloksensa.

5.1 Jigi

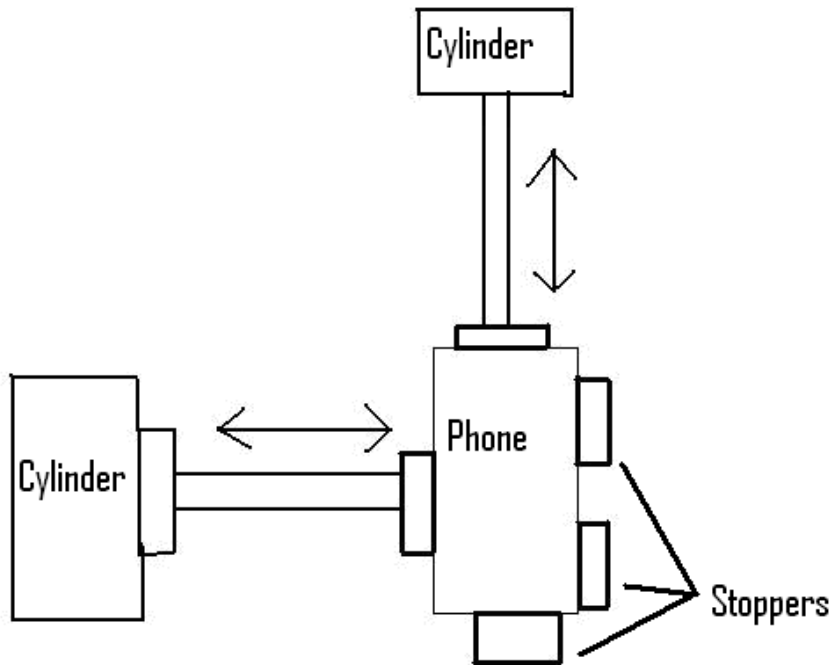
Jigin (kuva 6) tärkein tehtävä on pitää testattava kappale (puhelin) oikeassa paikassa vakaana. Toinen tärkeä tehtävä on saada kappale samaan paikkaan vaihdon jälkeen.



KUVA 6. Teollisuusrobotin työpöytä

Jigi rakennettiin yksinkertaisesti käytettävän teollisuusrobotin työpöydälle. Jigissä on kolme pysäytintä (kuva 7), jotka menevät puhelimen ympärille, ja kaksi sylinteriä, jotka painavat ja pitävät puhelimen oikeassa paikassa. Robotti ohjaa sylintereitä. Käyttäjä käy manuaalisesti vaihtamassa testattavan kappaleen.

Pysäyttimien ja sylintereiden päät on pehmustettu kumilla, jolloin laite ei aiheuta vahinkoa itse testattavalle kappaleelle. Ulkoiset naarmut tai suorat kovat törmäykset eivät ole sallittuja.



KUVA 7. Jigin rakenne

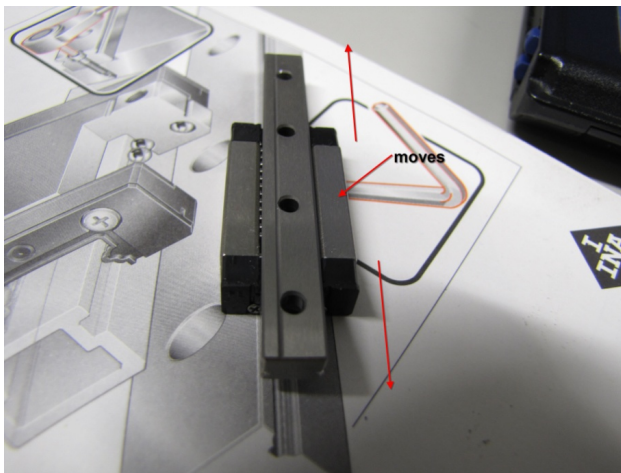
5.2 Robotti

Robotti (kuva 8) on ABB:n valmistama ja mallimerkki on IMM1400. Robottia käytetään, koska on tärkeää saada kiihtyvyyssanturi täsmälleen samaan paikkaan jokaisella mittauskerralla. Jos mittapaikka vaihtelee, on tulosten vertailu mahdotonta. Robotilla on monia liikeakseleita, joten se on mahdollista ohjelmoida suorittamaan haluttuja liikkeitä monipuolisesti. Tässä tapauksessa robotti liikkui kolmen mittauspisteen välillä pysty- ja sivuttaissuunnassa. Käyttäjä voi halutessaan tehdä muutoksia robotin ohjelmaan, jos mittaus halutaan suorittaa eri tavalla.



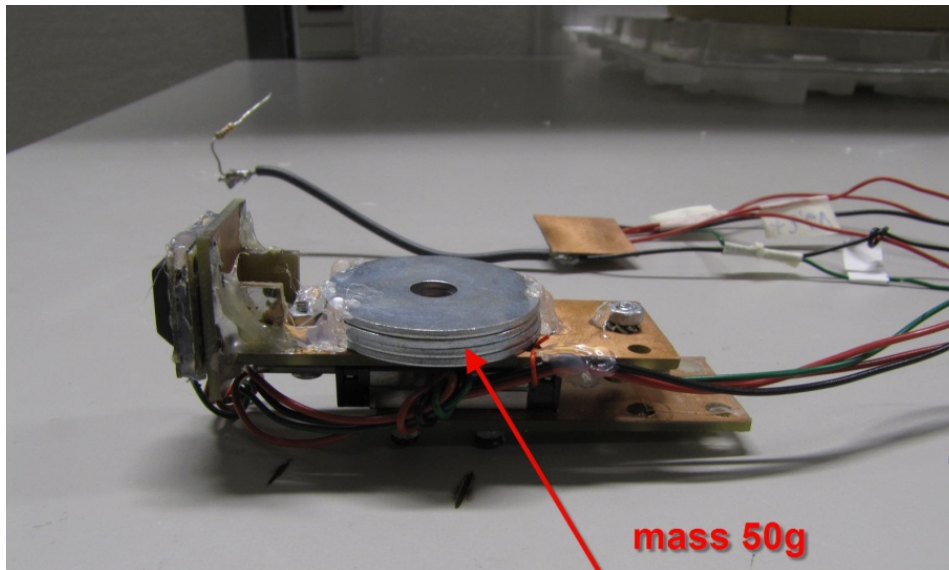
KUVA 8. Teollisuusrobotti

Testerin lineaarijohde (kuva 9), jossa myös kiihtyvyyssanturi sijaitsee, on kiinnitetty robotin päähän. Robotti laskee lineaarijohteen testattavan kappaleen pinnalle. Lineaarijohdin liikkuu pystysuunnassa ylös tai alas, jotta robotti ei pääse vahingoittamaan testeriä tai mitattavaa kappaletta. Kiihtyvyyssanturia pidetään paikallaan 10 s, jotta käyttäjä ehtii lukea tuloksen. Sitten robotti liikkuu seuraavaan mittauspisteeseen.



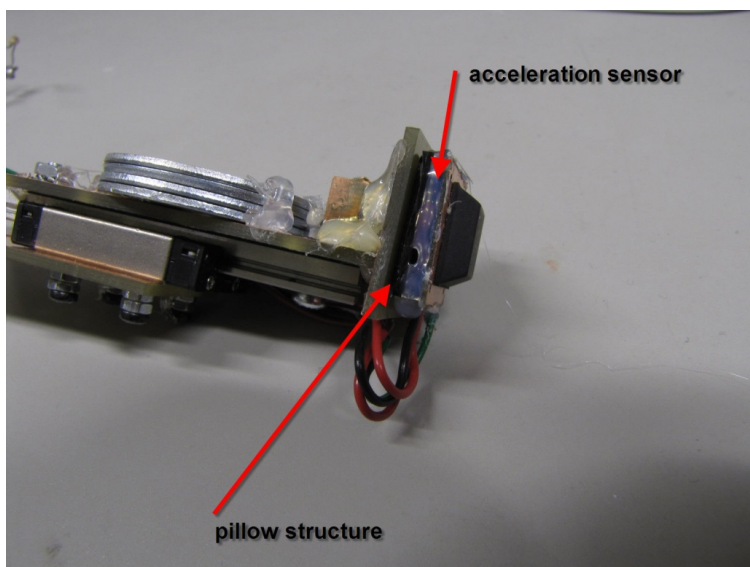
KUVA 9. Lineaarijohde

Kuvassa 10 on nähtävissä mittalaitteen rakenne sivusta kuvattuna. Kuvan nuoli osoittaa massaa (mass), joka on liimattu lineaarijohteeseen.

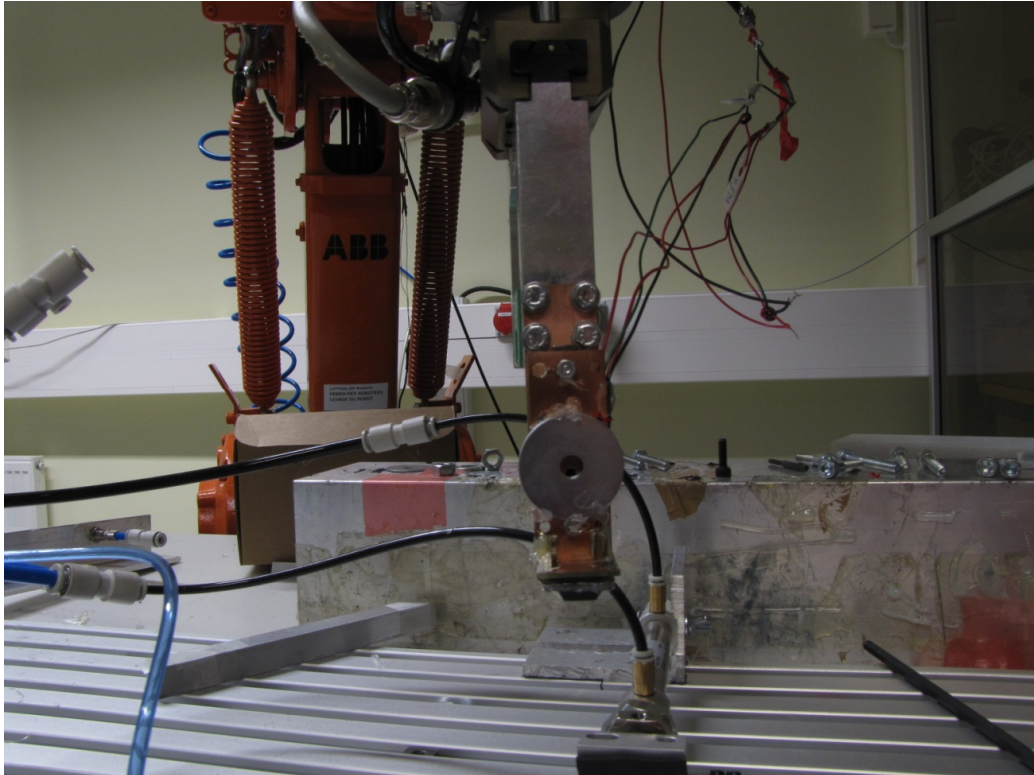


KUVA 10. Lineaarijohde koottuna

Pehmustetyyny (pillow structure) kiihtyvyyssanturin (acceleration sensor) ja pään välillä (kuva 11). Tyynyrakenne antaa kiihtyvyyssanturille tilaa liikkua vapaasti pystysuunnassa ylös ja alas.



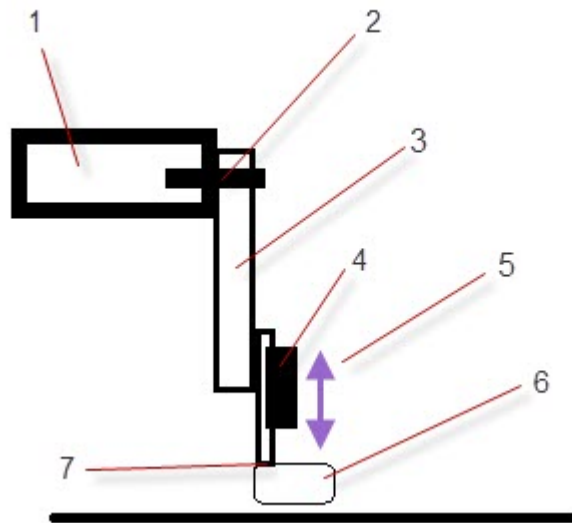
KUVA 11. Lineaarijohde ja kiihtyvyyssanturi



KUVA 12. Testeri koottuna

5.3 Linearijohde ja kiihtyvyyssanturi

Mittalaite (kuva 13) koostuu lineaarijohteesta (osa 3). Lineaarijohteen tarkoitus on, että kiihtyvyyssanturi pääsee liukumaan pystysuunnassa ylös tai alas (osa 5). Tämä auttaa mittauspään ajamista kiinni mitattavan kappaleen pinnalle. Näin robotti ei riko itse kiihtyvyyssanturia (osa 6) eikä mitattavaa kappaletta. Lineaarijohteeseen on kiinnitetty myös 50 gramman massa (osa 4). Kiihtyvyyssanturi on kiinnitetty kuumaliimalla ja sen välissä on pehmustetyyny (osa 7). Pehmustetyynyn tarkoitus on antaa kiihtyvyyssanturin väristä vapaasti. Koko paketti on kiinnitetty (osa 2) lineaarijohteesta teollisuusrobotin päähän (osa 1) (kuva 12).



KUVA 13. Periaatekuva testerin rakenteesta

Kiihtyvyyssanturin malli on LIS344AHL (liite 3), ja sen valmistaja on STMicroelectronics. Se on vähävirtainen, 3-akselinen kapasitiivinen kiihtyvyyssanturi, joka antaa analogista jännitesignaalia ulos. Kiihtyvyyssanturin käyttöjännite on 3,6 v. (11.)

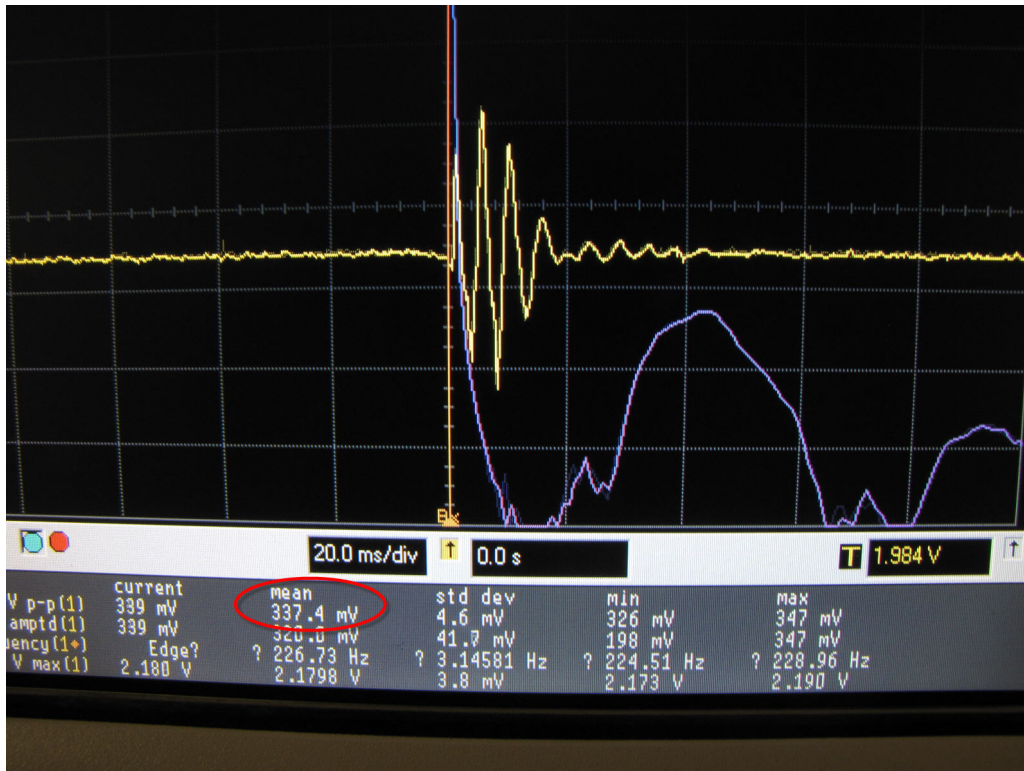
Kiihtyvyyssanturin kalibrointi suoritettiin ennen mittauksia. Kalibrointi tapahtui pitämällä kiihtyvyyssanturia ensin suoraan alaspäin suunnattuna ja sen jälkeen 180 astetta pystysuunnassa ja ottamalla anturin ilmoittamat luvut talteen. Tämän jälkeen vähentämällä suuremmasta arvosta pienempi ja jakamalla se kahdella saadaan selville kiihtyvyyssanturin herkkyys ja samalla Zero-G level (0 G taso) (liite 4).

Esimerkiksi, jos alaspäin tulos on 400 mV ja ylöspäin 200 mV, saadaan erotukseksi 200mV. Kun $200\text{mV}/2$, saadaan 100 mV. Tämä on kiihtyvyyssanturin herkkyys eli 1G on 100mV. Zero-G level (0 G) on näin ollen 400 mV. Ylös- ja alaspäin suunnattuna anturin ero on aina 2G.

Herkkyuden määrittäminen suoritettiin oskilloskoopille ja Arduino-mikrokontrollerille erikseen. Kiihtyvyyssanturin saatua Z-suunnan, arvoa käytettiin mittaustuloksena.

5.4 Oskilloskooppi

Oskilloskoopin merkki on Agilent infiniium. Oskilloskooppi näyttää kiihtyvyyssanturilta tulevan jännitteen. V.amtid lukema (kuva 14) on tässä testissä käytetty lukema. Se tarkoittaa keskiarvoa amplitudista.



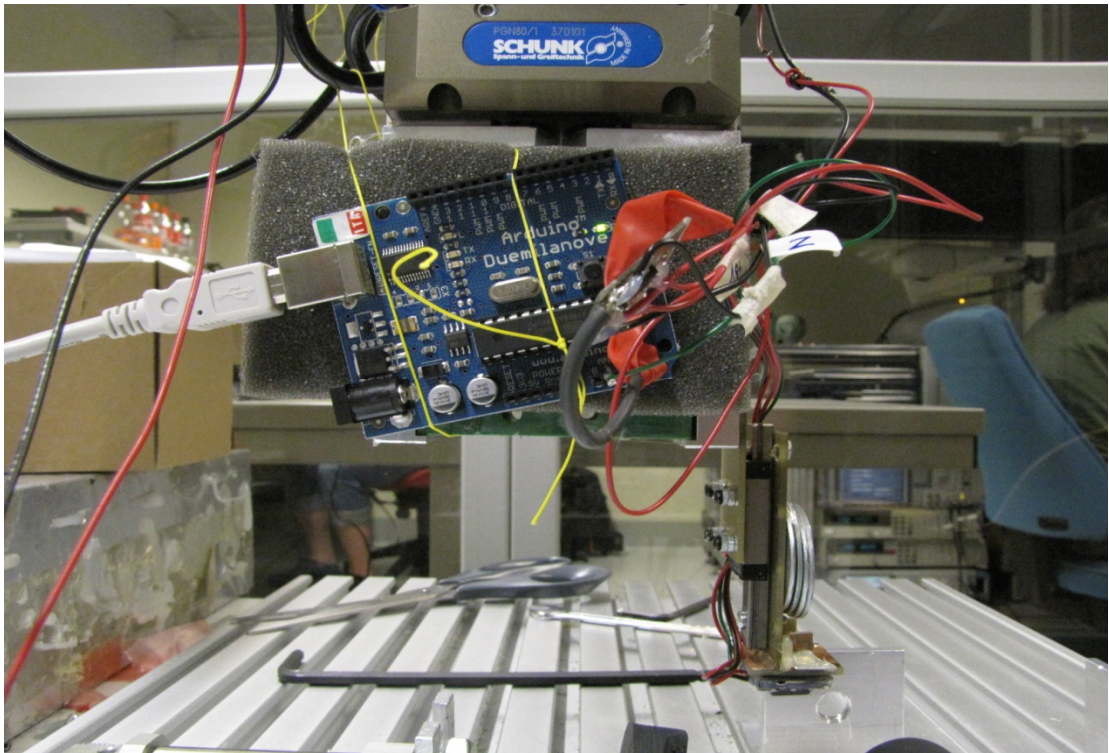
KUVA 14. Oskilloskoopin näyttö

Oskilloskooppi oli säädetty näyttämään horisontaalisesti 50ms/div ja vertikaalisesti 100mv/div. Nämä säädöt haettiin kokeilemalla. Tarkoitus oli, että amplitudista saataisiin mahdollisimman selvä ja käyttäjä pystyisi lukemaan tuloksen selkeästi näytöltä.

5.5 Arduinon ja testeri

Arduino-mikrokontrolleriin kytkettiin virta- ja maajohto sekä kiihtyvyyssanturilta tuleva z-akseli (kuva 15). Arduino kytkettiin tietokoneeseen usb-kaapelilla ja asennettiin tarvittavat ohjelmistot.

Internetistä löytyi valmista koodipätkää kiihtyvyyssanturin ja Arduinon yhdistämiseksi. Kokeilemalla selvisi, että oikean toiminnan saamiseksi tällaisessä käytössä laitteen ohjelmaan oli tehtävä muutoksia. Vaatimuksia olivat, että tietokone antaisi suoraan tuloksen G-voimina ja tulos olisi selkeästi luettavissa ja tarkka.



KUVA 15. Arduino kytkettynä testeriin

5.6 Arduinin ohjelmointi

Seuraava on ohjelmakoodi, joka ladattiin Arduinoon: (12.) (13.)

```
The circuit:  
analog 0: accelerometer self test  
analog 1: z-axis  
analog 2: y-axis  
analog 3: x-axis  
analog 4: ground  
analog 5: vcc
```

Arduino-mikrokontrolleriin on kytkettynä 4. pinniin maa (ground), jonka ohjelma lukee. Myös Z-suunnan johto on kytkettynä.

```
const int groundpin = 18;      // analog input pin 4 -- ground  
const int powerpin = 19;      // analog input pin 5 -- voltage  
const int zpin = 1;           // z-axis (only on 3-axis models)
```

Anturi kalibroitiin yksinkertaisella tavalla. Levossa alaspäin suuntautuneena anturi ilmoitti luvuksi 417 mV ja ylöspäin suunnattuna 317 mV. Sivuttain suunnattuna anturi ilmoitti noin 367 mV, joka tarkoittaa sitä, että 1 G on tällä kiihtyvyyssanturilla noin 50 mV. Ylöspäin ja alaspäin suunnatun anturin ero on aina 2G:tä. Tämä Zero-G syötettiin myös ohjelmaan alla olevaan kohtaan.

```
int maxValue = 0;  
int average_maxValue = 0;  
int ZeroG = 417;  
float GValue, averageGValue;
```

Ohjelma käyttää tietokoneen tietoliikenneporttia (COM1), joka on asetettu toimimaan nopeudella 9 600 bittiä sekunnissa.

```
void setup()  
{  
  // initialize the serial communications:  
  Serial.begin(9600);  
  
  pinMode(groundpin, OUTPUT);  
  pinMode(powerpin, OUTPUT);  
  digitalWrite(groundpin, LOW);
```

```

digitalWrite(powerpin, HIGH);
Serial.print("*****");
Serial.println();

```

Ohjelma käskää kiihtyvyyssanturin lukea z-akselin tuloksia koko ajan. Ohjelmallisesti tulosten lukeminen toteutettiin siten, että tietokone ilmoittaa 20 000 mittatuloksen seasta suurimman lukeman. Tämän suurimman luvun ja kahden sitä edeltävän luvun keskiarvo on yhtäkuin mittatulos. Tämä keskiarvo siksi, jotta tulos vastaisi paremmin koko mittaustapahtumaa kuin vain yhtä amplitudin huippua, mikä ei välttämättä vastaa todellisuutta.

```

}

void loop()
{
  int zvalue,zvalue_old1,zvalue_old2;
  int count;

  count = 0;
  maxValue = zvalue_old1 = zvalue_old2 = zvalue = 0;
  while(count < 20000)
  {
    count++;
    zvalue_old2 = zvalue_old1;
    zvalue_old1 = zvalue;
    zvalue = analogRead( zpin );

    if ( zvalue > maxValue )
    {
      maxValue = zvalue;
      average_maxValue = (zvalue_old2 + zvalue_old1 + zvalue)/3;

```

Seuraaksi ohjelma muuntaa saadun tuloksen G-voimiksi. Ensin se vähentää saadusta arvosta Zero-G arvon (417mV). Tämän jälkeen tulosta muutetaan, jotta tietokone osaisi ilmoittaa luvun kolmen desimaalin tarkkuudella. Tietokone ilmoittaa (printtaa) kolme arvoa käyttäjän tietoon. Se kertoo tuloksen keskiarvon G-voimina ja maksimiarvon G-voimina ja raaka-datan milliwatteina.

```

}
}
GValue = ((maxValue - ZeroG)*20); // 100 => 2G
averageGValue = ((average_maxValue - ZeroG)*20);

```



```
Serial.print( GValue / 1000);  
Serial.print( " G " );  
Serial.print( averageGValue / 1000 );  
Serial.print( " G " );  
Serial.print( "(raw:" );  
Serial.print( maxValue );  
Serial.print( ")" );  
Serial.println();  
}
```

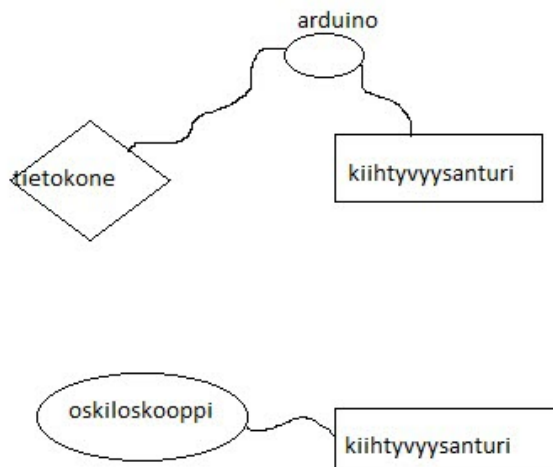
Teollisuusrobotti tuo laitteen testattavan kappaleen pinnalle pitäen sitä paikallaan 10 sekunnin ajan. Tässä ajassa käyttäjä pystyy hyvin lukemaan tuloksen ja ottamaan sen talteen. Muuttamalla tulos valmiiksi G-voimiksi säästetään myöhemmiltä laskutoimituksilta.

6 MITTAUSTAPA

6.1 Periaate

Gage R&R-tutkimuksessa oli mukana kymmenen testikappaletta. Operaattoreita oli kaksi ja mittauskierroksia kolme. Yhdelle operaattorille tuli yhteensä 30 mittausta.

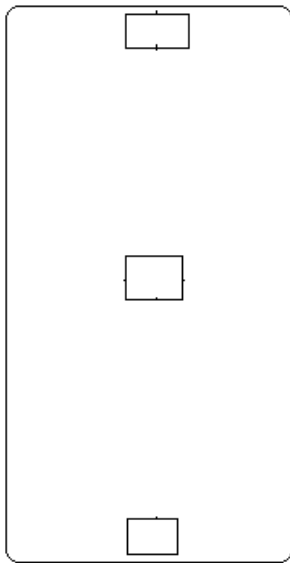
Mittauksessa käytettiin kahta eri tulosten lukutapaa, joten mittaustapahtuma järjestettiin kahdesti. Oskilloskooppia käytettäessä kiihtyvyyssanturi oli kytketty suoraan kiinni oskilloskooppiin (kuva 16). Arduinon käytössä kytkentä muuttui siten, että oskilloskooppi poistettiin ja anturin ja tietokoneen väliin kytkettiin Arduino-mikrokontrolleri. Virran kiihtyvyyssanturi sai samalla tavalla kummassakin tapauksessa.



KUVA 16. Laitteiston erit lukutavat

Operaattoreille annettiin toimintaohjeet, ja kumpikin mittasi ohjeiden mukaan. Mittausjärjestys oli arvottu, mutta kummassakin mittaustapahtumassa käytettiin samaa arvottua järjestystä. Myös testattavat kappaleet olivat samoja kummallakin kerralla.

Operaattori otti testattavan kappaleen, kytki sen päälle ja asetti testilaitteen jigiin. Mitauksen jälkeen testikappale sammutettiin ja siirrettiin pois. Tämän jälkeen seurasi seuraava mittaus samalla tavalla. Ylimääräiset muuttujat suljettiin pois pitämällä mittausympäristö samana mittausten välillä. Mittauspisteitä oli kolme (kuva 17). Kun mitaukset oli suoritettu, tulokset analysoitiin käyttäen Minitab-ohjelmaa.



KUVA 17. Mittauspisteet

6.2 Mittausohjeet

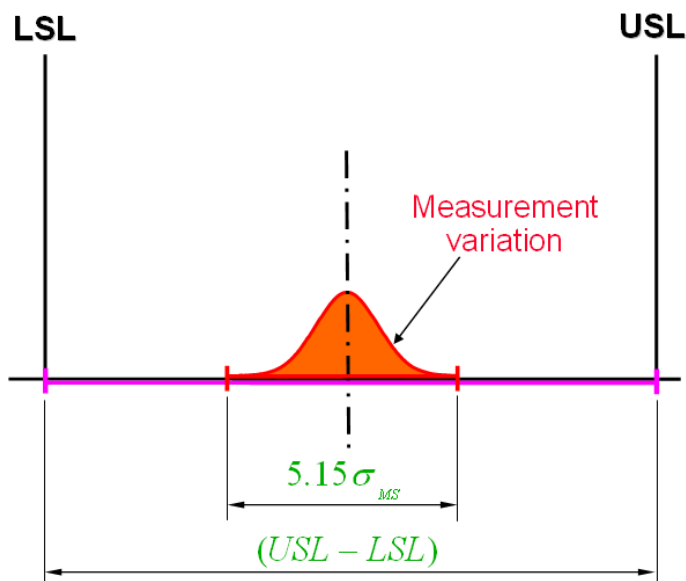
Operaattori sai seuraavat mittausohjeet:

1. Käyttäjä valmistelee robotin.
2. Puhelin asetetaan jigiiin ja varmistetaan oikea sijainti.
3. Oskilloskooppi valmistellaan kuntoon.
4. Testattava kappale käynnistetään.
5. Robotti käynnistetään ja aloitetaan mittaus.
6. Tulokset kirjataan muistiin.
7. Testattava kappale vaihdetaan.
8. Sama järjestys toistuu jokaisen kappaleen kohdalla.
9. Mittaustulokset analysoidaan.

7 TULOKSET JA ANALYSOINTI

7.1 Oskilloskooppi

Jotta saataisiin todellisia pass- tai fail-tuloksia, täytyy mittausvaihtelun (mittauskohinan) olla pieni verrattuna mitattavan kappaleen hyväksymisrajoihin (kuva 18). Tämä tarkoittaa %-toleranssia. (5, s. 1.)



KUVA 18. Toleranssirajat (6, s. 26)

Toleranssiraja saadaan sijoittamalla Minitab-ohjelmistosta saatu Total Gage R&R tulos (liite 1) ja käyttämällä haluttua %-toleranssialuetta. Termi USL-LSL viedään vasemmalle puolelle yhtälöä ja termit $5,15 * \sigma_{MS}$ jaetaan käytetyllä %-toleranssialueella.

$$\% - Tolerance = \frac{5,15 * \sigma_{MS}}{USL - LSL}$$

KAAVA 4

$$0.10 = \frac{5,15 * \sigma_{MS}}{USL - LSL}$$

$$\Leftrightarrow USL - LSL = \frac{5,15}{0,10} * \sigma_{MS}$$

$$\Rightarrow USL - LSL = 51,5 * 0,136447$$

$$\Rightarrow USL - LSL = 7,03 \rightarrow 7,03 / 2 = \pm 3,5$$

0,10 -> 10 % = Suhde mittaushajonnan ja toleranssialueen välillä.

5,15* σ_{MS} = olettaa, että 99,00 % mittaustuloksista on kyseisen alueen sisällä

σ_{MS} = on mittaushajonta (Total Gage R&R). Minitab ohjelmasta saatu tulos (liite 1)

Tulos $\pm 3,5$ tarkoittaa, että jos oletetaan, että mittauslaitteiston tarkkuus pitäisi olla 10 kertaa parempi kuin mitattavien kappaleiden toleranssi, niin kappaleiden valmistustoleranssi pitäisi olla $\pm 3,5$ tai suurempi.

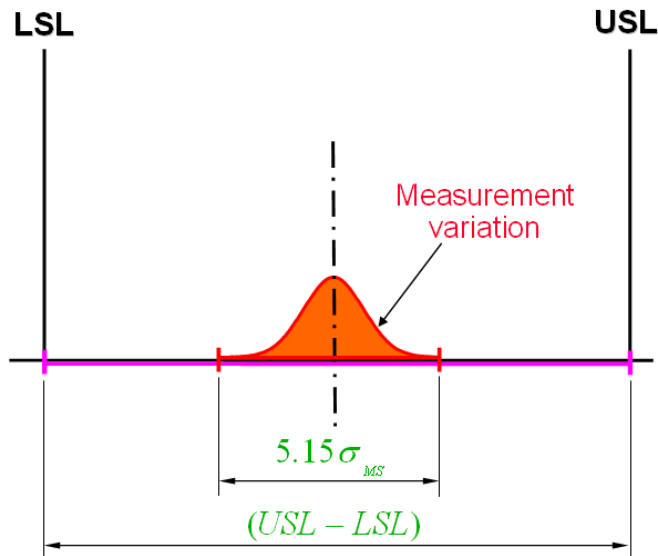
Jos kappaleiden valmistustoleranssi on pienempi. Riski, että hyväksytään hylkyosia (fail parts), kasvaa. Toleranssiraja $\pm 3,5$ päästään %-toleranssilukemaan 10 %.

%-toleranssi rajoina käytetään seuraavaa:

%-toleranssi ≤ 10 %	HYVÄ
10 % \leq %-toleranssi ≤ 30 %	MARGINAALINEN
30 % \leq %-toleranssi	HUONO.

7.2 Arduino

Arduinoa käytettäessä mittausjärjestys oli samanlainen kuin oskilloskoopin kanssa. Vain tuloksen lukeminen tapahtui erilailla. Tulokset analysoitiin samalla tavalla. Mittapiste kolmonen antoi suurimman mittaushajonnan tuloksiin (liite 2). Tätä suurinta mittaushajontaa käytettiin toleranssirajan (kuva 19) laskemiseen.



KUVA 19. Toleranssirajat (6, s. 26)

Toleranssiraja laskettiin samalla tavalla kuin oskilloskoopille (kaava 5).

$$\% - Tolerance = \frac{5,15 * \sigma_{MS}}{USL - LSL} \quad \text{KAAVA5}$$

$$0.10 = \frac{5,15 * \sigma_{MS}}{USL - LSL}$$

$$\Leftrightarrow USL - LSL = \frac{5,15}{0.10} * \sigma_{MS}$$

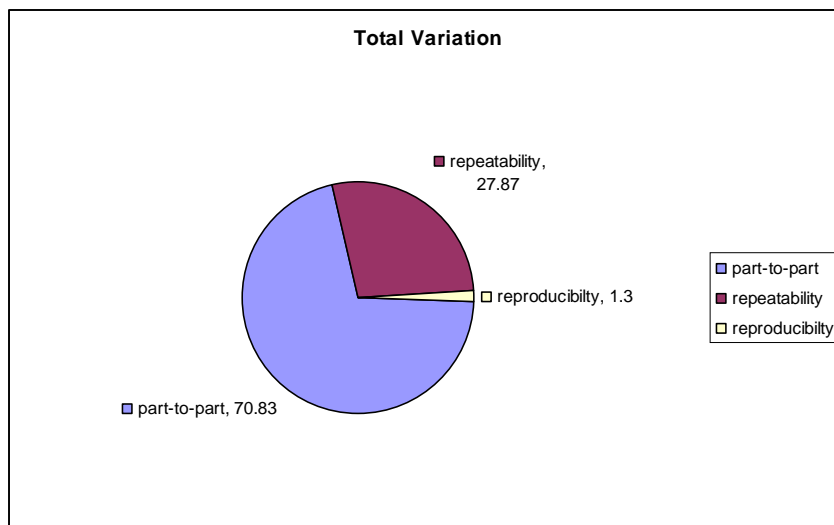
$$\Rightarrow USL - LSL = 51,5 * 0.083950$$

$$\Rightarrow USL - LSL = 4,32 \text{ -- } > 4,32 / 2 = \pm 2,2$$

Vaaditaan siis $\pm 2,2$ tarkkuus, jotta päästään %-toleranssilukemaan 10 %.

7.3 Oskilloskooppi tulosten analysointi

Koko mittausepätkarkkuus (kuva 20) 100 % koostuu osien (part-to-part), laitteiston (repeatability) ja käyttäjien (reproducibility) mittavirheistä. Total Gage R&R 27,87 % +1.3 % = 29,17 % (liite1) sisältää vain käyttäjien ja laitteiston epätarkkuudet. Tämä on mittalaitteiston mittavirhe.



KUVA 20. Mittausepätkarkkuus koostuu

Käyttäjistä johtuva mittausepätkarkkuus (reproducibility) syntyy käyttäjän käyttäessä samaa mittauslaitteistoa useita kertoja:

$$\frac{1,30}{29,17} \cdot 100\% = 4,5 \%$$

Laitteistosta johtuva mittausepätkarkkuus (repeatability) on mittauksen keskiarvon mittavirhe mitattaessa samassa mittausolosuhteissa eri käyttäjillä samoja osia ja mittoja.

$$\frac{27,87}{29,17} \cdot 100\% = 95,5 \%$$

Nähdään, että laitteistosta aiheutuva mittausvirhe on suurta. Itse operaattorista ei synny niin paljon vaihtelua.

7.4 Arduino tulosten analysointi

Arduinon toleranssirajaksi saatiin $\pm 2,2$, joten tulos oli tarkempi kuin oskilloskooppia käyttäen. Itse mittaus oli yksinkertaisempaa ja nopeampaa.

Käyttäjistä johtuva mittavirhe (reproducibility) (liite 2) on seuraava:

$$\frac{1,57}{14,05} \cdot 100\% = 11,1 \%$$

Laitteistosta johtuva mittavirhe (repeatability) (liite 2) on seuraava:

$$\frac{12,48}{14,05} \cdot 100\% = 88,9 \%$$

8 LOPPUTULOKSET

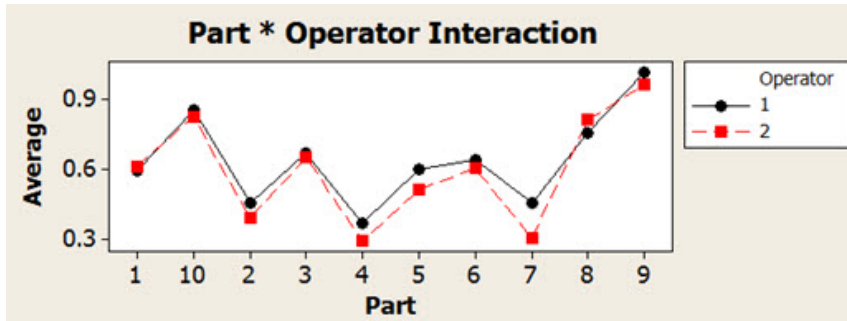
Nähdään, että Ardionoa käyttämällä päästää 37 % pienempään toleranssirajaan (taulukko 1). Tämä taas tarkoittaa sitä, että laite pystyy mittaamaan tarkemmin ja huomaamaan pienempiä eroja kappaleiden välillä.

TAULUKKO 1. Tulokset

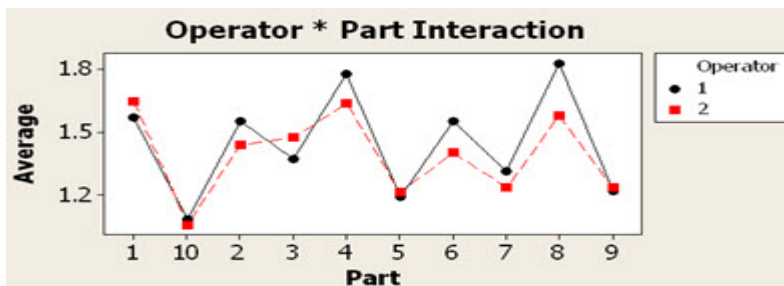
	Toleranssiraja	Repeatability	Reproducibility
Oskiloskooppi	±3,5	95,5	4,5
Arduino	±2,2	88,9	11,1

Laitteistosta aiheutuva mittavirhe on pienempi Arduinolla kuin oskilloskoopilla. Tämä johtuu siitä, että tulokset olivat helpommin luettavissa Arduinon käytettäessä. Käyttäjillä oli helpompi toimia mittalaitteen kanssa. Operaattorista aiheutuva suhteellinen osuus virheestä on taas Arduinossa suurempaa.

Kun verrataan kuvissa (kuva 21 ja kuva 22) olevia diagrammeja, huomataan, että Arduinolla mitattaessa käyttäjien mittatulokset seuraavat toisiaan huomattavasti paremmin kuin oskilloskoopilla mitattuna. Oskilloskoopilla tulokset ovat vaihdelleet melko paljon operaattorien kesken. Joko tulos on ollut vaikea lukea laitteelta tai sitten mitattavat kappaleet ovat olleet vaikeita mitata. Mittausvaikeus voi johtua kappaleen ohjelmiston keskeneräisyydestä. Arduinon käytettäessä mittakappaleet olivat täsmälleen samoja, joten niiden ei pitäisi olla syynä vaihteluihin. Lopun liitteistä voi tarkastella myös muiden mittapisteiden välisiä eroja, joita en ottanut mukaan tulosten analysointiin, koska otin vain pahimman pisteen mittaheitot käsittelyyn.

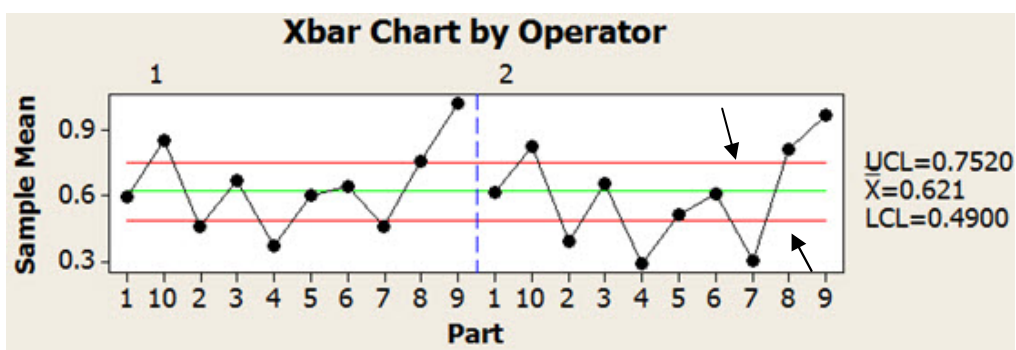


KUVA 21. Arduinon mittapiste 3

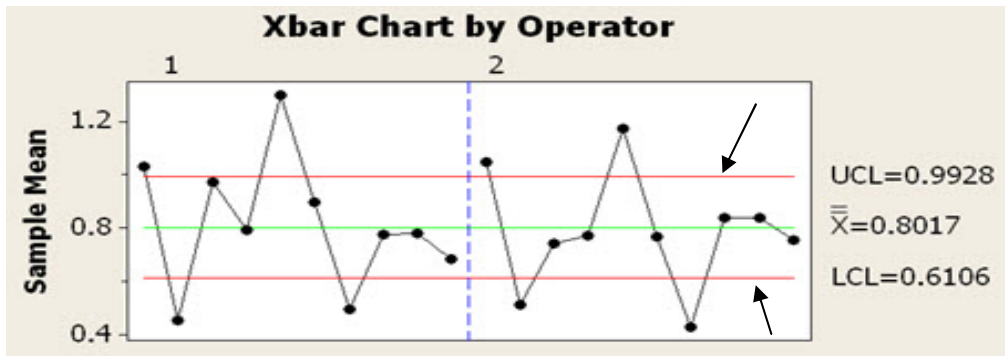


KUVA 22. Oskilloskoopin mittapiste 2

Verrattaessa (kuva 23 ja kuva 24) kummankin mittaustavan mittakeskiarvoja (mean), pitäisi niiden olla mahdollisimman kaukana punaisista viivoista eli kaukana toisistaan, jotta testilaite pystyisi havaitsemaan tulokset kunnolla. Kumpikaan mittaustapa ei täytä tätä vaatimusta kunnolla. Arduinolla on hieman parempi tulos kuin oskilloskoopilla.



KUVA 23. Arduino mittapiste 3 Mean



KUVA 24. Oskilloskooppi mittapiste 2 Mean

9 POHDINTA

Kun aloitin kosketusnäytöllisen matkapuhelimen tuntopalautustoiminnan testilaitteen kehittämisen, oli, tilaajayrityksen projektin aikataulujen takia, laitteiston valmiiksi saamisella kova kiire. Lähtökohtaisesti päätettiin käyttää hyväksi mahdollisimman paljon jo valmiina olevaa laitteistoa ja tehdä testilaitteesta mahdollisimman yksinkertainen.

Itse mittaaminen laitteella on yksinkertaista. Käyttäjän täytyy vain osata kytkeä testilaitte kiinni robottiin ja ohjata sitä oikein. Mittatuloksen voi lukea kahdella eri tavalla, joko Arduinoa hyväksi käyttäen tai oskilloskoopilla. Oskilloskoopilla mitattaessa huomasin suurimman ongelman olevan sen, miten käyttäjä tulkitsee tuloksen. Aluksi mittatuloksen lukeminen oli erittäin vaikeaa, koska tulos päivittyi liian nopeasti. Kokeilujen jälkeen oskilloskoopille löytyi hyvät säädöt, jotta lukeminen onnistui.

Arduinoa käyttäessä tulosten lukeminen oli helppoa, koska ohjelma ilmoitti luvut selkeästi ja mittausloki piti huolen siitä, että tulos jäi talteen. Tämä auttoi paljon, eikä tuloksen talteen ottamisessa tarvinnut kiirehtiä. Arduinon suurin ongelma käyttöön otettaessa on löytää tai tehdä sopiva ohjelma, jotta mittaus toimisi.

Ottaen huomioon testilaitteen yksinkertaisuuden, mielestäni mittausten tarkkuus oli siedettävällä tasolla. Mitattavissa kappaleissa oli havaittavissa ongelmia, ei niinkään visuaalisesti vaan ohjelmallisesti, sillä sen keskeneräisyys vaikutti puhelimen vakauteen ja toimintavarmuuteen.

Lopputuloksena voidaan sanoa, että Arduinoa käyttäen mittatulos oli tarkempi. Mittaaminen oli myös mielekkäämpää ja käyttäjälle jäi vähemmän arpomista tuloksen saamiseksi. Arduinon käytön ainoa miinuspuoli on ohjelmointikielen hallitsemisen tarve. Kustannuksia verrattaessa Arduinon noin 20 € hinta oli pieni verrattuna hyvän oskilloskooppiin usean sadan euron hintaan.

Itse työ oli mielenkiintoista, ja sain toteuttaa sitä työni ohella, mikä auttoi tiukassa aikataulussa töiden ja koulun välillä tasapainoiluun. Kiitän tästä joustavaa työnantajaa, joka mahdollisti tämän.

LÄHTEET

1. Hämäläinen, Timo 2007. Heittoliikkeen arviointi kiihtyvyyssanturia käyttäen. Oulu: Oulun seudun ammattikorkeakoulu, hyvinvointiteknologian koulutusohjelma. Opinnäytetyö.
2. Wikipedia. 2010. Kiihtyvyys. Saatavissa: <http://fi.wikipedia.org/wiki/Kiihtyvyys>. Hakupäivä 20.2.2010.
3. Dimension engineering. 2010. A beginner's guide to accelerometers. Saatavissa: <http://www.dimensionengineering.com/accelerometers.htm>. Hakupäivä 20.2.2010.
4. Hedman, Antti 2007. Tietokannan ja analysointityökalujen kehittäminen piirikorttien tuotantotestereille prosessiparannustarkoitukseen. Espoo: Teknillinen korkeakoulu, sähkö- ja tietoliikennetekniikan osasto. Diplomityö. Saatavissa: <http://lib.tkk.fi/Dipl/2007/urn009925.pdf>. Hakupäivä 20.2.2010.
5. Pylvänäinen, Mika 2010. Principal Specialist, Perceived Mechanic. Nokia Oyj. Keskustelut.
6. Pylvänäinen, Mika 2008. Variable Gauge R&R Training Material. Oulu. Nokia Oyj.
7. Tilastokeskus. 2010. Luottamusväli. Saatavissa: <http://www.stat.fi/meta/kas/luottamusvali.html>. Hakupäivä 14.9.2010.
8. Haukipuro, Ari 2007. GPRS:n lähetinmittaukset. Oulu: Oulun seudun ammattikorkeakoulu, tietoliikennetekniikan koulutusohjelma. Opinnäytetyö.
9. Yle. 2010. Abitreenit. Normaalijakauma, keskiarvo ja keskihajonta. Saatavissa: <http://abitreenit.yle.fi/treenaa/lukio/aine/14/78/208/216>. Hakupäivä 14.9.2010.
10. Arduino. 2010. Saatavissa: <http://en.wikipedia.org/wiki/Arduino>. Hakupäivä 9.3.2010.

11. STMicroelectronics. 2010. LIS344AHL. Saatavissa:
<http://www.st.com/stonline/products/literature/ds/14337.pdf>. Hakupäivä 13.09.2010.
12. Arduino. 2010. ADXL3xxAccelerometer. Saatavissa:
<http://arduino.cc/en/Tutorial/ADXL3xx>. Hakupäivä 8.7.2010.
13. Sormunen, Jouko 2010. EXT-Senior Designer. Nokia Oyj. Keskustelut.

LIITEET

Liite 1. Oskilloskoopin Gage R&R:n tulokset

Liite 2. Arduino-mikrokontrolleri Gage R&R:n tulokset

Liite 3. Kiihtyvyyssanturi LIS344ALH dokumentti 1

Liite 4. Kiihtyvyyssanturi LIS344ALH dokumentti 3

Liite 5. Kiihtyvyyssanturi LIS344ALH dokumentti 2

Liite 6. Oskilloskoopin Gage R&R:n tulostkäyrät

Liite 7. Arduino-mikrokontrolleri Gage R&R:n tulostkäyrät

2G_Gage R&R Study - ANOVA Method**2G_Two-Way ANOVA Table With Interaction**

Source	DF	SS	MS	F	P
Part	9	2.60091	0.288990	15.5398	0.000
Operator	1	0.04267	0.042667	2.2943	0.164
Part * Operator	9	0.16737	0.018597	1.0562	0.415
Repeatability	40	0.70427	0.017607		
Total	59	3.51521			

Alpha to remove interaction term = 0.25

2G_Two-Way ANOVA Table Without Interaction

Source	DF	SS	MS	F	P
Part	9	2.60091	0.288990	16.2458	0.000
Operator	1	0.04267	0.042667	2.3986	0.128
Repeatability	49	0.87164	0.017789		
Total	59	3.51521			

2G_Gage R&R

Source	VarComp	%Contribution (of VarComp)
Total Gage R&R	0.0186178	29.17
Repeatability	0.0177885	27.87
Reproducibility	0.0008293	1.30
Operator	0.0008293	1.30
Part-To-Part	0.0452002	70.83
Total Variation	0.0638180	100.00

Source	StdDev (SD)	Study Var (6 * SD)	%Study Var (%SV)
Total Gage R&R	0.136447	0.81868	54.01
Repeatability	0.133374	0.80024	52.80
Reproducibility	0.028797	0.17278	11.40
Operator	0.028797	0.17278	11.40
Part-To-Part	0.212603	1.27562	84.16
Total Variation	0.252622	1.51573	100.00

Number of Distinct Categories = 2

Gage R&R Study - ANOVA Method

Gage R&R for Response3

Gage name: Response3
 Date of study: 16.8.2010
 Reported by:
 Tolerance:
 Misc:

Two-Way ANOVA Table With Interaction

Source	DF	SS	MS	F	P
Part	9	2.38374	0.264860	52.5748	0.000
Operator	1	0.02993	0.029927	5.9404	0.038
Part * Operator	9	0.04534	0.005038	0.7711	0.643
Repeatability	40	0.26133	0.006533		
Total	59	2.72034			

Alpha to remove interaction term = 0.25

Two-Way ANOVA Table Without Interaction

Source	DF	SS	MS	F	P
Part	9	2.38374	0.264860	42.3191	0.000
Operator	1	0.02993	0.029927	4.7817	0.034
Repeatability	49	0.30667	0.006259		
Total	59	2.72034			

Gage R&R

Source	VarComp	%Contribution (of VarComp)
Total Gage R&R	0.0070476	14.05
Repeatability	0.0062586	12.48
Reproducibility	0.0007889	1.57
Operator	0.0007889	1.57
Part-To-Part	0.0431002	85.95
Total Variation	0.0501478	100.00

Source	StdDev (SD)	Study Var (6 * SD)	%Study Var (%SV)
Total Gage R&R	0.083950	0.50370	37.49
Repeatability	0.079112	0.47467	35.33
Reproducibility	0.028088	0.16853	12.54
Operator	0.028088	0.16853	12.54
Part-To-Part	0.207606	1.24564	92.71
Total Variation	0.223937	1.34362	100.00

Number of Distinct Categories = 3



LIS344ALH

MEMS inertial sensor
high performance 3-axis $\pm 2/\pm 6$ g ultracompact linear accelerometer

Features

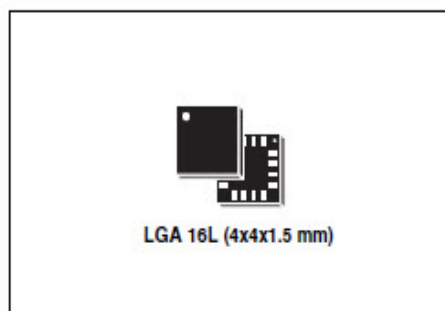
- 2.4 V to 3.6 V single supply operation
- ± 2 g / ± 6 g user selectable full-scale
- Low power consumption
- Output voltage, offset and sensitivity are ratiometric to the supply voltage
- Factory trimmed device sensitivity and offset
- Embedded self test
- RoHS/ECOPACK[®] compliant
- High shock survivability (10000 g)

Description

The LIS344ALH is an ultra compact consumer low-power three-axis linear accelerometer that includes a sensing element and an IC interface able to take the information from the sensing element and to provide an analog signal to the external world.

The sensing element, capable of detecting the acceleration, is manufactured using a dedicated process developed by ST to produce inertial sensors and actuators in silicon.

The IC interface is manufactured using an ST proprietary CMOS process with high level of integration. The dedicated circuit is trimmed to better match the sensing element characteristics.



The LIS344ALH has a dynamically user selectable full-scale of ± 2 g / ± 6 g and it is capable of measuring accelerations over a maximum bandwidth of 1.8 kHz for all axes. The device bandwidth may be reduced by using external capacitances. The self-test capability allows the user to check the functioning of the system.

The LIS344ALH is available in Land Grid Array package (LGA) manufactured by ST. It is guaranteed to operate over an extended temperature range of -40 °C to $+85$ °C.

The LIS344ALH belongs to a family of products suitable for a variety of applications:

- Mobile terminals
- Gaming and virtual reality input devices
- Antitheft systems and inertial navigation
- Appliance and robotics.

Table 1. Device summary

Order codes	Temp range [°C]	Package	Packaging
LIS344ALH	-40 to +85	LGA-16L	Tray
LIS344ALHTR	-40 to +85	LGA-16L	Tape and reel

2 Mechanical and electrical specifications

2.1 Mechanical characteristics

Table 3. Mechanical characteristics @ Vdd =3.3 V, T = 25 °C unless otherwise noted⁽¹⁾

Symbol	Parameter	Test condition	Min.	Typ. ⁽²⁾	Max.	Unit
Ar	Acceleration range ⁽³⁾	FS pin connected to GND	±1.8	± 2		g
		FS pin connected to Vdd	±5.4	± 6		
So	Sensitivity ⁽⁴⁾	Full-scale = ±2 g	Vdd/5 - 5%	Vdd/5	Vdd/5 + 5%	V/g
		Full-scale = ±6 g	Vdd/15 - 10%	Vdd/15	Vdd/15 + 10%	
SoDr	Sensitivity change Vs Temperature	Delta from +25 °C		± 0.01		%/°C
Voff	Zero-g level ⁽⁴⁾	Full-scale = ±2 g T = 25 °C	Vdd/2 - 5%	Vdd/2	Vdd/2 + 5%	V
OffDr	Zero-g level change Vs Temperature	Delta from +25 °C		±0.4		mg/°C
NL	Non linearity ⁽⁵⁾	Best fit straight line Full-scale = ±2 g		±0.5		% FS
CrossAx	Cross-axis ⁽⁶⁾			±2		%
An	Acceleration noise density	Vdd = 3.3 V; Full-scale = ±2 g		50		µg/√Hz
Vt	Self test output voltage change ^{(7),(8),(9)}	X axis T = 25 °C; Vdd=3.3 V	80	140	200	mV
		Y axis T = 25 °C; Vdd=3.3 V	-200	-140	-80	mV
		Z axis T = 25 °C; Vdd=3.3 V	100	230	350	mV
Fres	Sensing element resonant frequency ⁽¹⁰⁾	X, Y, Z axis	1.8			KHz
Top	Operating temperature range		-40		+85	°C
Wh	Product weight			0.040		gram

1. The product is factory calibrated at 3.3 V. The operational power supply range is from 2.4 V to 3.6 V. Voff, So and Vt parameters will vary with supply voltage.

2. Typical specifications are not guaranteed.

3. Guaranteed by wafer level test and measurement of initial offset and sensitivity.

4. Zero-g level and sensitivity are essentially ratiometric to supply voltage at the calibration level ±8%.

5. Guaranteed by design.

6. Contribution to the measuring output of an inclination/acceleration along any perpendicular axis.

7. "Self test output voltage change" is defined as $V_{out}(V_{st-Logic1}) - V_{out}(V_{st-Logic0})$.

8. "Self test output voltage change" varies cubically with supply voltage.

9. When full-scale is set to ±6 g, "Self test output voltage change" is one third of the specified value at ±2 g.

10. Minimum resonance frequency Fres=1.8 kHz. Sensor bandwidth= $1/(2 \cdot \pi \cdot 110k\Omega \cdot C_{load})$, with Cload>1 nF.

Mechanical and electrical specifications

LIS344ALH

2.2 Electrical characteristics

Table 4. Electrical characteristics @ Vdd =3.3 V, T = 25 °C unless otherwise noted⁽¹⁾

Symbol	Parameter	Test condition	Min.	Typ. ⁽²⁾	Max.	Unit
Vdd	Supply voltage		2.4	3.3	3.6	V
Idd	Supply current	Normal mode		680	850	μA
		Power-down mode		1	5	
Vfs Vst Vpd	Full-scale input Self-test input Power-down input	Logic 0 level	0		0.3*Vdd	V
		Logic 1 level	0.7*Vdd		Vdd	V
Rout	Output impedance of VoutX, VoutY, VoutZ		90	110	130	KΩ
Cload	Capacitive load drive ⁽³⁾ for VoutX, VoutY, VoutZ		1			nF
Ton	Turn-on time at exit of Power-down mode	Cload expressed in μF		550*Cload+0.3		ms
Top	Operating temperature range		-40		+85	°C

1. The product is factory calibrated at 3.3 V.

2. Typical specifications are not guaranteed.

3. Minimum resonance frequency $F_{res}=1.8$ kHz. Device bandwidth= $1/(2*\pi*110\text{ k}\Omega*Cload)$, with $Cload>1$ nF.

Gage R&R (ANOVA) for Response3

Gage name: Response3
 Date of study: 16.8.2010

Reported by:
 Tolerance:
 Misc:

