



Open your mind. LUT.

Lappeenranta University of Technology

**CAN-väyläliikenteisen kiihtyvyyssanturin CANopen-
toteutus ja liityntä PLC-laitteeseen**
**Implementation of a CANopen interfaced accelerometer
connected to a PLC device using CAN-bus**

Joona Pöyhä

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto
LUT School of Energy Systems
Sähkötekniikka

Joona Pöyhiä

CAN-väyläliikenteisen kiihtyvyyssanturin CANopen-toteutus ja liityntä PLC-laitteeseen

2018

Kandidaatintyö.

22 s.

Tarkastaja: TKT Tuomo Lindh

Tässä kandidaatin työssä suunnitellaan ja toteutetaan kiihtyvyyden mittausjärjestelmä, joka voidaan liittää PLC:hen CAN-väylää pitkin. Mittausjärjestelmä koostuu mikrokontrollerista ja kolmiakselisesta kapasitiivisesta MEMS-kiihtyvyyssanturimoduulista, joka on liitetty mikrokontrolleriin I²C-väylää pitkin. Mikrokontrollerin ja PLC:n välisessä kommunikaatiossa käytetään CANopen-protokollaa.

Mikrokontrollerille koodattiin C-kielinen ohjelma, joka lukee kiihtyvyydataa kiihtyvyyssanturilta halutulla näytteistystaajuudella ja lähettää datan edelleen PLC:lle. Järjestelmän toimintaa voidaan ohjata PLC:ltä CANopen:n NMT-protokollan mukaisilla käynnistys- ja pysäytyskomennoilla.

ABSTRACT

Lappeenranta University of Technology
LUT School of Energy Systems
Electrical Engineering

Joona Pöyhiä

Implementation of a CANopen interfaced accelerometer connected to a PLC device using CAN-bus

2018

Bachelor's Thesis.
22 p.

Examiner: D.Sc. Tuomo Lindh

In this bachelor's thesis an acceleration measurement system is designed and build. The measurement system can be connected to a PLC via CAN-bus. It consists of a microcontroller and a triple-axis MEMS accelerometer module which is connected to the microcontroller via I²C-bus. CANopen-protocol is used in the communication between the microcontroller and the PLC.

A program which reads the acceleration data from the accelerometer and sends it to the PLC was programmed for the microcontroller in the C-language. The measurement system can be controlled from the PLC using the start and stop commands of the CANopen NMT-protocol.

SISÄLLYSLUETTELO

Käytetyt merkinnät ja lyhenteet

Johdanto.....	7
1. Kiihtyvyyden mittaaminen	7
1.1 Kiihtyvyys	7
1.2 Kiihtyvyyssanturien ominaisuuksista	8
1.3 Yleisimmät kiihtyvyyssanturityypit ja niiden toiminta.....	9
1.3.1 Kapasitiivinen kiihtyvyyssanturi.....	9
1.3.2 Piezo-resistiivinen kiihtyvyyssanturi	10
1.3.3 Piezo-sähköinen kiihtyvyyssanturi.....	11
2. Järjestelmän suunnittelu	11
2.1 Käyttökohteet	12
2.1.1 Sähköajoneuvon identifiointi.....	12
2.1.2 Heiluriliike.....	12
2.2 Laitteiston valinta	13
2.2.1 Kehitysalusta	13
2.2.2 Kiihtyvyyssanturi	14
2.2.3 PLC ja CANopen Master.....	15
3. Järjestelmän toteutus	15
3.1 Mikrokontrollerin ohjelma	16
3.2 CAN-väylä.....	16
3.2.1 CANopen-protokolla	19
3.2.2 CAN-kommunikaatio PLC:n kanssa	19
3.3 I ² C-väylä.....	19
3.3.1 I ² C-kommunikaatio kiihtyvyyssanturin kanssa	21
4. Yhteenveto.....	21
Lähteet	22

KÄYTETYT MERKINNÄT JA LYHENTEET

CAN	Control Area Network, Alueellinen ohjausverkko
CANopen	CAN-protokollaan perustuva ylemmän tason protokolla
DSP	Digital Signal Processor, Digitaalinen signaali prosessori
eCAN	Enhanced Controller Area Network
FIFO	First In First Out, Ensimmäisenä sisään, ensimmäisenä ulos
I ² C/I2C	Inter-Integrated Circuit, Synkroninen sarjäväylä
LSB	Least Significant Bit, Vähiten merkitsevä bitti
LUT	Lappeenranta University of Technology, Lappeenrannan teknillinen yliopisto
MEMS	Micro Electro-Mechanical System, Sähkömekaaninen mikrosysteemi
NMT	Network Management, Verkon hallinta
ODT	Output Data Rate, Datan lähetysnopeus
OSI	Open Systems Interconnection, tiedonsiirtoprotokollien kerrosmalli
PDO	Process Data Object, CANopen-protokollassa datan siirtämiseen käytetty protokolla
PLC	Programmable Logic Controller, Ohjelmoitava logiikka
SDA	Serial Data, I ² C-sarjäväylän datalinja
SLC	Serial Clock, I ² C-sarjäväylän kellolinja
SPI	Serial Peripheral Interface, Synkroninen sarjäväylä
<i>a</i>	kiihtyvyys
<i>C</i>	kapasitanssi
<i>d</i>	etäisyys
<i>E</i>	jännite
<i>F</i>	voima
<i>g</i>	putoamiskiihtyvyys Maan pinnalla
<i>k</i>	jousivakio
<i>m</i>	massa
<i>r</i>	säde
<i>S</i>	pinta-ala
<i>s</i>	Laplace-muuttuja
<i>t</i>	aika
<i>V</i>	jännite
<i>v</i>	nopeus
<i>x</i>	paikka
α	kulmakiihtyvyys
θ	kulma
ω_0	luonnollinen kulmataajuus
ξ	vaimennusvakio

Alaindeksit

f	takaisinkytkentä (<i>feedback</i>)
g	painovoimasta johtuva
h	heiluri
m	mekaaninen
mc	massa-kansi (<i>mass-cap</i>)
mb	massa-pohja (<i>mass-base</i>)
min	minimi
out	lähtö
t	tangentiaalinen
x	x-akselin suuntainen
y	y-akselin suuntainen

JOHDANTO

Tässä kandidaatintyössä suunnitellaan ja toteutetaan kiihtyvyydenmittausjärjestelmä, joka voidaan kytkeä CAN-väylää (Control Area Network) pitkin PLC:hen (Programmable Logic Controller). Työssä toteutettu mittausjärjestelmä sisältää Texas Instrumentsin TMS320F28016 mikrokontrollerin, johon on kiinnitetty kapasitiivinen MEMS (Micro Electro-Mechanical System) kiihtyvyyssanturimoduuli I²C-väylän (Inter-Integrated Circuit) avulla. Mikrokontrolleri toimii mittausjärjestelmän aivoina välittäen kiihtyvyyssanturilta lukemansa kiihtyvyyssanan PLC:lle. PLC:n ja mikrokontrollerin välisessä tiedonsiirrossa käytetään CANopen-protokollaa.

1. KIIHTYVYYDEN MITTAAMINEN

Kappaleen kiihtyvyys voidaan määrittää kappaleen nopeuden, tai paikan muutoksen perusteella, mutta tämä vaatii nopeuden ensimmäisen ja paikan toisen derivaatan laskemista. Häiriöisessä ympäristössä derivaattojen laskeminen voi johtaa suuriin virheisiin ja luotettavan kiihtyvyyssarvon saaminen voi olla vaikeaa. Tämän takia kiihtyvyyden mittaamiseen kannattaa käyttää erillistä kiihtyvyyssanturia, joka kertoo suoraan kappaleen kokeman kiihtyvyyden.

Kiihtyvyystietoa hyödynnetään nykyään lukemattomissa eri sovelluksissa ja kiihtyvyyden mittaamiseen onkin saatavilla useita erilaisia järjestelmiä. Monet nykyaikaiset laitteet, koneet ja esimerkiksi ajoneuvot hyödyntävät kiihtyvyystietoa toiminnassaan. Esimerkiksi älypuhelimissa on kiihtyvyyssanturi, jolla voidaan tarkkailla puhelimen liikkeitä ja asentoa. Ajoneuvoissa kiihtyvyyssantureita käytetään puolestaan esimerkiksi törmäysten havainnointiin sekä osana luistoneston toimintaa.

1.1 Kiihtyvyys

Kiihtyvyys on fysikaalinen suure, joka kuvaa nopeuden muutosta. Kiihtyvyys voidaan esittää nopeuden ensimmäisenä derivaattana, tai paikan toisena derivaattana ajan suhteen

$$a = \frac{dv}{dt} = \frac{d^2x}{dt^2}, \quad (1.1)$$

missä a on kiihtyvyys, v on nopeus ja x on paikka.

Newtonin toisen lain mukaan kappaleeseen vaikuttava voima on verrannollinen kappaleen liikemäärän muutokseen ajan suhteen (Newton I. 1687)

$$F = \frac{d(m \cdot v)}{dt}, \quad (1.2)$$

missä F on voima, m on massa, v on nopeus ja t on aika.

Massan ollessa vakio tästä seuraa

$$F = m \frac{dv}{dt}, \quad (1.3)$$

ja edelleen

$$F = ma, \quad (1.4)$$

missä a on kappaleen kokema kiihtyvyys.

Elektronisten kiihtyvyyssanturien toiminta perustuu Newtonin toiseen lakiin. Kiihtyvyyssanturissa on massa, joka pääsee liikkumaan vapaasti anturin kotelon suhteen. Massan liikettä voidaan havainnoida erilaisilla menetelmillä ja massan kokema kiihtyvyys voidaan johtaa havaintojen perusteella.

1.2 Kiihtyvyyssanturien ominaisuuksista

Kiihtyvyyssanturi on mekaaninen värähtelijä, jonka ominaisuuksia voidaan kuvata matemaattisesti Laplace-tasossa yhtälöllä

$$ms^2X(s) + bsX(s) + kX(s) = -mA(s), \quad (1.5)$$

missä m on kiihtyvyyssanturin värähtelijän massa, s Laplace-muuttuja, b vaimennusvakio ja k jousivakio. $X(s)$ on systeemin lähtö ja $A(s)$ systeemin tulo, tässä tapauksessa anturiin kohdistuva kiihtyvyys. Kun yhtälö 1.5 ratkaistaan lähdön $X(s)$ suhteen saadaan

$$X(s) = \frac{mA(s)}{ms^2 + bs + k}. \quad (1.6)$$

Yhtälö 1.6 voidaan esittää muuttujien $\omega_0 = \sqrt{k/m}$ ja $2\xi\omega_0 = b/m$ avulla muodossa

$$X(s) = -\frac{A(s)}{s^2 + 2\xi\omega_0 s + \omega_0^2}, \quad (1.7)$$

missä ω_0 on systeemin luonnollinen kulmataajuus ja ξ vaimennusvakio. Yhtälöstä 1.7 voidaan johtaa systeemille siirtofunktio

$$G(s) = \frac{X(s)}{A(s)} = -\frac{1}{s^2 + 2\xi\omega_0 s + \omega_0^2}. \quad (1.8)$$

(Fraden, J. 2010)

Hyvin suunnitellulla kiihtyvyyssanturilla tulisi olla yksi selkeästi tunnistettava resonanssitaajuus ja tasainen taajuusvaste, jonka alueella tarkin mittaaminen voidaan suorittaa. Tämän tasanen kaistanleveyden sisällä värähtelytaajuuden muuttuessa anturin lähtö vastaa oikein taajuuden muutosta ilman, että anturi vahvistaisi signaalia taajuusvasteessa olevien vaihteluiden surauksena. (Fraden, J. 2010)

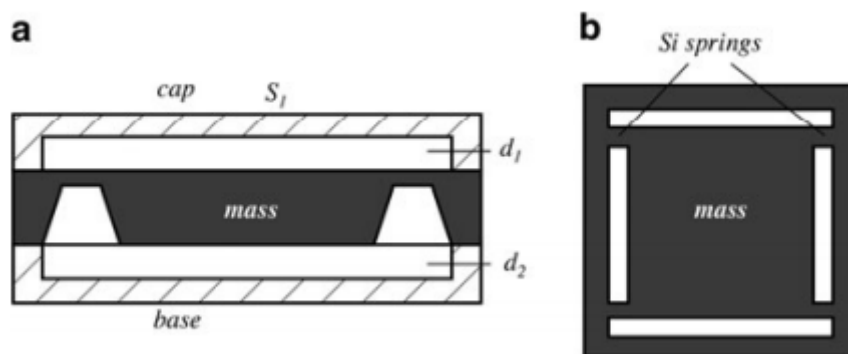
1.3 Yleisimmät kiihtyvyyssanturityypit ja niiden toiminta

Kiihtyvyyssiedon tarve eri sovelluksissa on vauhdittanut halpojen ja yksinkertaisten kiihtyvyyssanturien kehitystä ja markkinoilla onkin saatavilla useita eri teknologioihin perustuvia ratkaisuja. Tässä työssä käytetty kapasitiivinen MEMS-anturi on yksi yleisimmistä anturityypeistä.

Sähköinen kiihtyvyyssanturi tarvitsee toimiakseen massan, jonka liikettä voidaan mitata anturin kotelon suhteen. Tätä massaa kutsutaan yleensä seismiseksi, tai inertiaaliseksi massaksi. Massan liikettä voidaan havainnoida erilaisella menetelmällä. Menetelmille yhteistä on niiden tuottama sähköinen signaali, jonka perusteella voidaan määrittää massaan kohdistuva kiihtyvyys. Signaali syntyy, kun jokin mitattava sähköinen ominaisuus, kuten kapasitanssi, resistanssi, tai piezo-sähköinen jännite muuttuu. (Fraden, J. 2010)

1.3.1 Kapasitiivinen kiihtyvyyssanturi

Kuvassa 1.1 näkyy poikkileikkauksen kapasitiivisesta kiihtyvyyssanturista. Anturin sisällä on massa (*mass*), joka on anturin kotelon (*cap*) kanteen ja pohjan (*base*) välissä. Massa on kiinnitetty koteloon piijousilla (*Si springs*) ja se pääsee liikkumaan kohti kotelon kanta ja pohjaa. Kotelon kansi ja pohja ovat etäisyyksien d_1 ja d_2 etäisyydellä massasta. (Fraden, J. 2010)



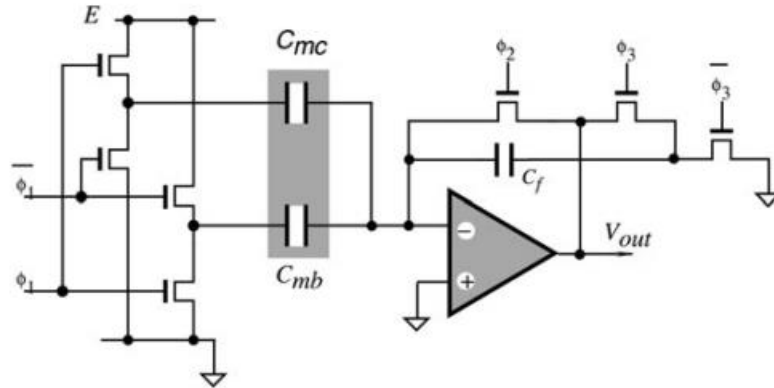
Kuva 1.1 Lämpileikkaus kapasitiivisesta kiihtyvyyssanturista, jossa on differentiaalinen kondensaattori (a) ja inertiaalimassa, jota kannattelee neljä pii joustaa (b) (Fraden, J. 2010).

Kotelon kansi ja massa muodostavat levykondensaattorin C_{mc} , jonka pinta-ala on S_1 ja levyjen etäisyys toisistaan d_1 . Kotelon pohja ja massa muodostavat puolestaan levykondensaattorin C_{mb} , jonka pinta-ala on eri kuin kondensaattorin C_{mc} ja levyjen etäisyys toisistaan d_2 . Kun massa liikkuu esimerkiksi kohti kotelon kanta välimatka d_1 pienenee, ja

välimatka d_2 kasvaa, etäisyyden Δ verran. Etäisyys Δ on verrannollinen massaan vaikuttavaan mekaaniseen voimaan F_m jaettuna piijousien jousivakiolla k

$$\Delta = \frac{F_m}{k}. \quad (1.9)$$

Kuvassa 1.2 on esitetty yksinkertainen vahvistinkytkentä, joka muuttaa kapasitanssin jännitteeksi.



Kuva 1.2 Vahvistinpiiri, joka muuttaa kapasitanssin jännitteeksi (Fraden, J. 2010).

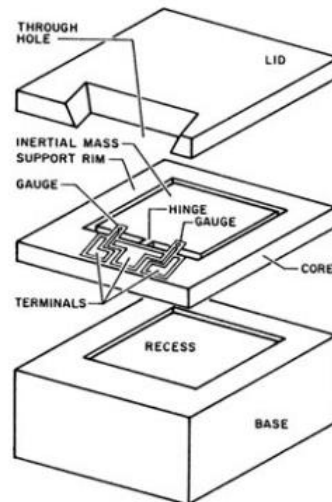
Kun kiihtyvyyssanturin kondensaattorit ovat kuvassa 1.2 esitetyn vahvistinpiirin tulona on vahvistimen lähtöjännite (V_{out}) verrannollinen kapasitanssin arvoon ja siten verrannollinen voimaan F_m

$$V_{out} = 2E \frac{C_{mc} - C_{mb}}{C_f}. \quad (1.10)$$

(Fraden, J. 2010)

1.3.2 Piezo-resistiivinen kiihtyvyyssanturi

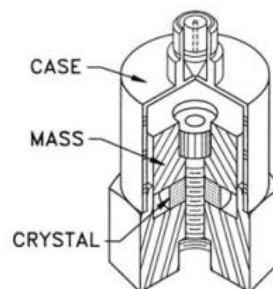
Kuvassa 1.3 esitetyn piezo-resistiivisen kiihtyvyyssanturin toiminta perustuu puolijohdemateriaalin resistanssin muutokseen, kun siihen vaikuttaa mekaaninen voima. Puolijohde on kiinnitetty massaa kannattelevaan jouseen, joka on puolestaan kiinni anturin kotelossa. Jousen taipuma voidaan suoraan korreloida massan liikkeen suuruuteen ja nopeuteen ja täten myös kiihtyvyyteen. (Fraden, J. 2010)



Kuva 1.3 Piezo-resistiivisen kiihtyvyyssanturin toimintaperiaate (Fraden J. 2010).

1.3.3 Piezo-sähköinen kiihtyvyyssanturi

Kuvassa 1.4 esitetty piezo-sähköinen anturi muuntaa mekaanisen energian suoraan sähköiseksi energiaksi kide materiaalissa. Käytetyimpiä kide materiaaleja ovat keraamiset piezo-elektroniset materiaalit, kuten bariumtitanaatti, lyijy-zirkoni-titanaatti ja lyijy-metaniobiitti. Kiteenä voidaan käyttää myös kvartsikidettä. Kide on asetettu sensorin kotelon ja massan väliin. Massa kohdistaa kiteeseen kiihtyvyyteen verrannollisen voiman. (Fraden, J. 2010)



Kuva 1.4 Piezo-sähköisen kiihtyvyyssanturin läpileikkaus (Fraden J. 2010).

2. JÄRJESTELMÄN SUUNNITELU

Kiihtyvyydenmittausjärjestelmää lähdettiin suunnittelemaan tiedossa olleen käyttökohteen perusteella. Kiihtyvyyssanturia halutaan käyttää ainakin LUT:n (Lappeenranta University of Technology) CAMBUS-projektissa sähköhybridibussin värähtelyn mittaamiseen, torninosturilla nostettavan esineen kiihtyvyyden mittaamiseen sekä LUT:n kursseilla heiluriin vaikuttavan voiman ja heilurin kulman mittaamiseen.

2.1 Käyttökohteet

Kiihtyvyyssanturi tulisi voida kytkeä ABB:n AC-500 PLC:hen CAN-väylää pitkin, koska sitä käytetään CAMBUS-projektissa bussin logiikan ohjaukseen. AC-500 on myös paljon opetuskäytössä LUT:n sähkötekniikan kursseilla.

2.1.1 Sähköajoneuvon identifiointi

LUT:n CAMBUS-projektissa kiihtyvyyssanturilla voidaan havainnoida bussin liikkeessä syntyvää värähtelyä. Värähtely liittyy sähköajoneuvoissa tyypilliseen ongelmaan, jossa voimansiirrossa tapahtuu värähtelyä, ajoneuvon rungon liitosten joustavuuden takia. Nämä värähtelyt ovat tyypillisesti 5 – 50 Hz:n välillä. Värähtelyt vaikuttavat ajoneuvon ajettavuuteen ja vaikeuttavat sen säätösuunnittelua. (Montonen et al., 2018)

Kiihtyvyydataa voidaan hyödyntää ajoneuvon käyttäytymisen identifioinnissa, mikä helpottaa mallipohjaisten säätömenetelmien kehittämistä. Identifioinnilla voidaan myös havaita ja ennaltaehkäistä ongelmia ajoneuvon mekaanisissa osissa. (Montonen et al., 2018)

2.1.2 Heiluriliike

Kiihtyvyyssanturin avulla voidaan tarkastella heilurin liikettä. Kiihtyvyyssanan avulla voidaan laskea esimerkiksi heilurin kulma, heiluriin vaikuttava voima sekä heilurin tangentialinen- ja kulmakiihtyvyys.

Heilurin kulma voidaan määrittää x- ja y-akselien kiihtyvyyksien suhteesta

$$\tan(\theta) = \frac{a_x}{a_y}, \quad (2.1)$$

ja edelleen

$$\theta = \tan^{-1}\left(\frac{a_x}{a_y}\right), \quad (2.2)$$

missä θ on heilurin kulma radiaaneina, a_x on x-akselin suuntainen kiihtyvyys ja a_y on y-akselin suuntainen kiihtyvyys.

Heiluriin vaikuttava voima voidaan laskea, kun tiedetään heilurin kulma ja painovoimasta aiheutuva voima

$$F_h = F_g \sin(\theta), \quad (2.3)$$

missä F_h on heiluriin vaikuttava voima ja F_g on painovoimasta aiheutuva voima.

Heilurin tangentialinen kiihtyvyys saadaan suoraan kiihtyvyyssanturilta, jos se on asennettu siten, että yksi anturin akseleista on heilurin tangentialisen liikkeen suuntainen.

Tangentiaalinen kiihtyvyys voidaan myös laskea, kun tiedetään heilurin kulma ja putoamiskiihtyvyys Maan pinnalla

$$a_t = \frac{F_g \sin(\theta)}{m}, \quad (2.4)$$

joka voidaan supistaa muotoon

$$a_t = g \sin(\theta), \quad (2.5)$$

missä a_t on heiluriin tangentiaalinen kiihtyvyys ja g on putoamiskiihtyvyys. Kulmakihtyvyys saadaan jakamalla tangentiaalinen kiihtyvyys heilurin säteellä r

$$\alpha_h = \frac{a_t}{r} \quad (2.6)$$

Pienin kulman muutos, joka kiihtyvyysanturilla voidaan havaita riipu anturin herkkyydestä. Pienin teoriassa havaittava kulman muutos on

$$\theta = \sin(\Delta a_{\min}), \quad (2.7)$$

missä Δa_{\min} on kiihtyvyysanturin herkkyys. Esimerkiksi 0.25 mg/LSB (Least Significant Bit) herkkyydellä pienin havaittavissa oleva kulman muutos olisi $\theta \approx 0.14^\circ$.

2.2 Laitteiston valinta

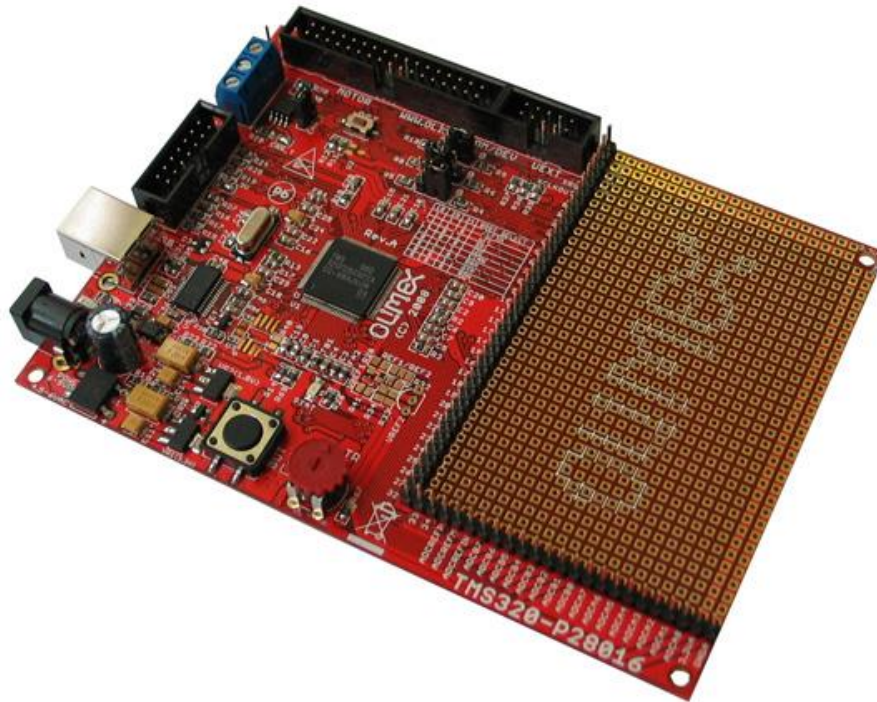
Järjestelmän laitteisto valittiin tiedossa olleiden käyttökohteiden perusteella. Käytetystä kehitysalustasta tulisi löytyä tuki CAN-väylälle sekä samalle sarjaväylälle, jota kiihtyvyysanturi käyttää kommunikointiin kehitysalustan kanssa.

2.2.1 Kehitysalusta

Kehitysalustaa valittaessa suurin huomio kiinnitettiin alustan tukemiin tiedonsiirtoprotokollisiin. Kehitysalustasta tulisi löytyä tuki ainakin CAN-protokollalle sekä kiihtyvyysanturista riippuen tuki I²C-, tai SPI-protokollalle (Serial Peripheral Interface). Alustan muut ominaisuudet, kuten prosessorin nopeus ja muistin määrä olivat toisarvoisia tekijöitä alustaa valittaessa, sillä kehitysalustalla ajettava ohjelma ei tarvitse suurta laskentatehoa.

Markkinoilla on lukuisia eri valmistajien kehitysalustoja, jotka ominaisuuksiltaan vastaisivat järjestelmän vaatimuksia. Suurin osa näistä kehitysalustoista on kuitenkin varsin kalliita ja niissä on myös paljon järjestelmän kannalta turhia ominaisuuksia, jotka osaltaan nostavat kehitysalustojen hintaa sekä tekevät niistä kookkaita.

Järjestelmän aivoiksi valittiin kuvassa 2.1 esitetty Olimex:n valmistama TMS320-P28016 kehitysalusta, joka vastaa mittausjärjestelmän asettamia vaatimuksia erittäin hyvin. TMS320-P28016 oli myös selvästi muita samantasoisia kehitysalustoja halvempi.



Kuva 2.1 Olimex:n valmistama TMS320-P28016 kehitysalusta. (Olimex 2018)

Kehitysalustan prosessorina toimii Texas Instrumentsin TMS320F28016P 32-bittinen DSP, joka tukee muun muassa CAN-, I²C- ja SPI-protokollia. Kehitysalustalla on myös Texas Instrumentsin SN65HVD32x CAN-lähetin-vastaanotin sekä reikälevyosa, johon kiihtyvyysanturi moduuli on helppo kiinnittää. CAN-väylää varten kehitysalustalla on kolminapainen ruuviterminaali sekä 120 Ω:n terminointivastus, joka on kytkettävissä hyppylangalla. I²C-väylän käyttöä varten prosessorin SDA- (Serial Data) ja SCL-pinnit (Serial Clock) on tuotu kehitysalusta reunalle. Kiihtyvyysanturimoduuli on täten helppo liittää kehitysalustan I²C-väylään. (Olimex 2018)

2.2.2 Kiihtyvyysanturi

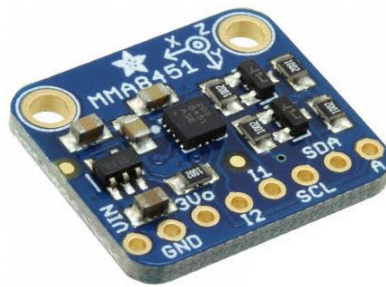
Kiihtyvyysanturiksi valittiin kuvassa 2.2 esitetty Adafruit:n valmistama MMA8451 kiihtyvyysanturi moduuli. Moduuli on käyttövalmis paketti, joka voidaan kytkeä suoraan mikrokontrolleriin I²C-väylän välityksellä. Moduuli sisältää NXP:n MMA8451Q kapasitiivisen kolmiakselisen kiihtyvyysanturin sekä tarvittavat oheiskomponentit, kuten 3v3 regulaattori kiihtyvyysanturin käyttöjännitteelle. Moduuli tarvitsee toimiakseen 3,3 – 5 voltin käyttöjännitteen, joka saadaan suoraan Olimex:n kehitysalustalta.

MMA8451Q kiihtyvyysanturi voidaan konfiguroida toimimaan 8-, tai 14-bitin resoluutiolla. Anturissa on myös kolme dynaamisesti valittavissa olevaa mittausaluetta: ±2, ±4 ja ±8 g:tä. Kaikkien mittausalueiden kaistanleveys on valittavissa 0,78 – 400 Hz:n väliltä. ±2 g

mittausalueella ja 14-bitin resoluutiolla anturin herkkyys on 0.25 mg/LSB. Anturin tarkemmat ominaisuudet eri mittausalueilla on eritelty taulukossa 2.1.

Taulukko 2.1 NXP:n MMA 8451Q kiihtyvyyssanturin tekniset ominaisuudet eri mittausalueilla 14-bittin resoluutiolla (NXP 2017).

	± 2 g	± 4 g	± 8 g
Herkkyys [mg/LSB]	0.25	0.5	1
Kaistanleveys [Hz]	0,78 – 400	0,78 – 400	0,78 – 400
ODT [Hz]	1,56 – 800	1,56 – 800	1,56 – 800



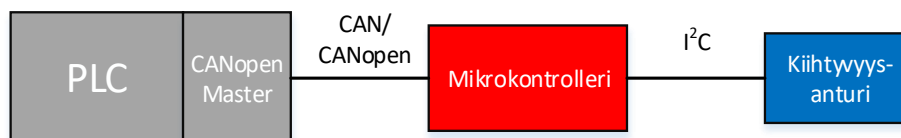
Kuva 2.2 Adafruit:n digitaalinen MMA8451 MEMS kiihtyvyyssanturimoduuli (Adafruit 2018).

2.2.3 PLC ja CANopen Master

Mittausjärjestelmä kytketään ABB:n AC-500 PLC:hen CAN-väylää pitkin. PLC:ssä on ABB:n CM598 CANopen Master-moduuli, jonka välityksellä kehitysalusta ja PLC kommunikoivat. PLC:n toiminnassa huomioitavaa on, että se lukee datan CAN-väylältä käännettyssä järjestyksessä vähiten merkitsevä tavu edellä. Kiihtyvyyssanturin tavujärjestys on tämän takia käännetty mikrokontrollerin puolella.

3. JÄRJESTELMÄN TOTEUTUS

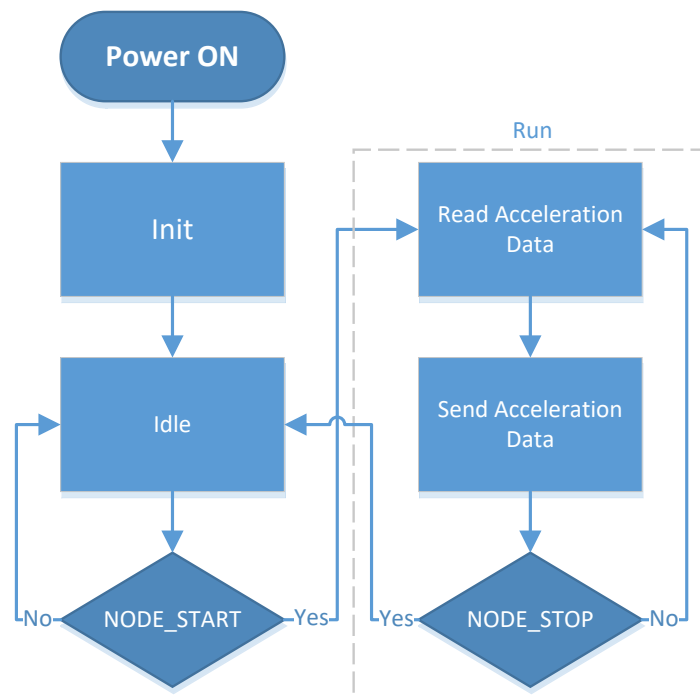
Mittausjärjestelmää varten koodattiin yksinkertainen C-kielinen ohjelma, joka lukee kiihtyvyyssanturin I²C-väylää pitkin kiihtyvyyssanturilta ja paketoit datan CAN-kehukseen, joka lähetetään CAN-väylää pitkin PLC:lle. Järjestelmän osat ja kytkennät on esitetty kuvassa 3.1.



Kuva 3.1 Koko mittausjärjestelmä, jossa PLC, mikrokontrolleri ja kiihtyvyyssanturimoduuli

3.1 Mikrokontrollerin ohjelma

Kiihtyvyyssdatan lukemista ja lähettämistä varten koodattiin yksinkertainen C-ohjelma. Ohjelman yksinkertaistettu toiminta on kuvattu vuokaaviolla kuvassa 3.2.



Kuva 3.2 Mikrokontrollerilla ajettavan ohjelman vuokaavio.

Ohjelman alussa alustetaan kaikki mikrokontrollerin tarvitsemat järjestelmät. Tämän jälkeen alustetaan ajastin, I²C- ja CAN-väylä. Alustusten jälkeen ohjelma siirtyy toimittomaan, eli Idle-tilaan odottamaan käynnistyskomentoa. Idle-tilassa ohjelma ei lue kiihtyvyyssdataa anturilta, eikä lähetä mitään PLC:lle.

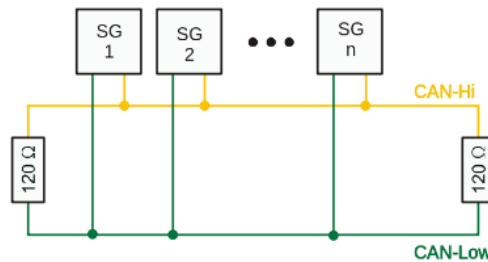
Ohjelma siirtyy toiminnalliseen, eli Run-tilaan, kun se vastaanottaa NMT-protokollan (Network Management) mukaisen käynnistyskomennon PLC:ltä. Run-tilassa ohjelma lukee kiihtyvyyssdataa I²C-väylää pitkin kiihtyvyyssanturilta ja lähettää datan CAN-väylää pitkin PLC:lle. Lähetyksen jälkeen ohjelma tarkistaa onko se vastaanottanut NMT pysäytys - komentoa PLC:ltä. Jos pysäytyskomento on vastaanotettu, ohjelma siirtyy takaisin Idle-tilaan odottamaan uutta käynnistyskomentoa. Muussa tapauksessa ohjelma jatkaa kiihtyvyyssdatan lukemista ja lähettämistä asetetulla näytteistystaajuudella. Järjestelmän näytteistystaajuus asetetaan ajastimella, joka luo keskeytyksen halutun ajanjakson välein.

3.2 CAN-väylä

CAN-väylä on Bosch:n vuodesta 1983 henkilöautojen ohjaukseen kehittämä sarjaväylä, joka on nykyisin laajalti käytössä ajoneuvojen ja koneiden ohjausjärjestelmissä sekä automaatiosovelluksissa (CAN in Automation). CAN-väylä perustuu solmupohjaiseen verkkoon, jossa kaikki viestit lähetetään kaikille verkkoon kytketyille laitteille. Laitteet ovat

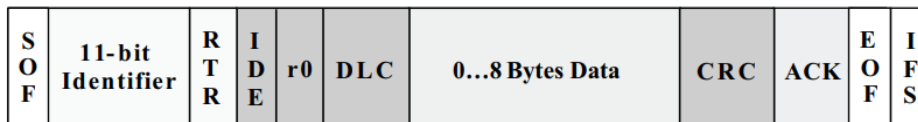
itse vastuussa siitä lukevatko ne viestin vai eivät. Näin dataa voidaan lähettää usealle laitteelle yhdellä viestillä.

CAN-verkko koostuu vähintään kahdesta laitteesta, jotka ovat kytketty samaan parikaapeliväylään. Verkossa olevia laitteita kutsutaan verkon solmuiksi. Väylä on terminoitu molemmista päistä $120\ \Omega$:n terminointivastuksilla. CAN-väylä on esitetty kuvassa 3.3.



Kuva 3.3 Fyysinen CAN-väylä (Wikipedia).

CAN-väylä voidaan konfiguroida käyttämään kahta viestiformaattia. Standardi 2.0A formaatti käyttää 11-bittistä tunnistetta, kun laajennetun formaatin 2.0B käyttäessä 29-bittistä tunnistetta. 29-bittinen tunniste mahdollistaa 536 870 912 uniikin osoitteen käytön, kun 11-bittisellä tunnisteella uniikkeja soitteita on maksimissaan 2048. CAN-väylään voidaan kuitenkin käytännössä kytkeä maksimissaan noin 100 laitetta. Yhden CAN-viestin kehys on esitetty kuvassa 3.4. (CAN in Automation)



Kuva 3.4 CAN-kehys 11-bittisellä tunnisteella (TI CAN 2016).

CAN-kehys koostuu seuraavista osista:

- SOF: Start of Frame, kehyksen aloitusbitti. Kertoo mistä kehys alkaa.
- ID: Identifier, yksilöllinen tunniste. 11 tai 29 bittiä.
- RTR: Remote Transmission Request, lähetyksen pyyntö toiselta laitteelta
- IDE: Identifier Extension, laajennettu tunnista. Kertoo, onko käytössä 11-, vai 29-bittinen tunniste
- r0: Reserved, varattu myöhempää käyttöä varten.
- DLC: Data Length Code, dataosan pituus.
- Data: 0 – 8 tavua.
- CRC: Cyclic Redundancy Check, tarkisteavain.
- ACK: Acknowledge, kuittausbitti. Kertoo, saiko vastaanottaja luettua viestin oikein.
- EOF: End of Frame, kehyksen lopetusosa. Kertoo missä kehys loppuu.

3.2.1 CANopen-protokolla

CANopen on yleensä CAN-protokollan päälle rakennettu ylemmän tason protokolla, joka käsittää sovellu kerroksen ja kommunikaatioprofiilin sekä sovellus-, laita- ja rajapintaprofiilit (can-wiki). CANopen on nykyään käytössä monissa ohjaus- ja automaatiojärjestelmissä, kuten ajoneuvoissa, laivoissa, junissa ja taloautomaatiossa (CiA CANopen).

CANopen rakentuu CAN-viestikehyksen päälle. CANopen käyttää 11-bittistä tunnistetta, jota kutsutaan CAN-ID:ksi. CAN-ID jakautuu kahteen osaan: 4-bittiseen funktiokoodiin, joka kertoo mitä protokollaa viesti edustaa ja 7-bittiseen tunnisteosaan, jota kutsutaan Node ID:ksi. Node ID kertoo mille laitteelle viesti on tarkoitettu. CANopen:n CAN-ID lähetetään CAN-kehysten tunniste kentässä ja data datakentässä.

3.2.2 CAN-kommunikaatio PLC:n kanssa

CANopen määrittää useita eri protokollia erilaisen tiedon siirtoon. Tässä työssä kehitettiin yksinkertainen CANopen-sovellus, joka käyttää CANopen:n määrittelemiä NMT- ja PDO-viestiprotokollia (Process Data Object). NMT-protokollaa käytetään esimerkiksi laitteiden alustamiseen, käynnistämiseen ja pysäyttämiseen. PDO-protokollaa käytetään puolestaan reaaliaikaiseen tiedonsiirtoon.

Mikrokontrolleri kuuntelee NMT-protokollan mukaisia käynnistys- ja pysäytyskomentoja, joiden mukaan laite joko aloittaa tai keskeyttää toimintansa. NMT-komennot voidaan lähettää joko Node ID:llä 0x00, tai mikrokontrollerille asetetulla 0x02 Node ID:llä. NMT-komennolla 0x01 mikrokontrolleri siirtyy toimettomasta tilasta toimivaan tilaan ja komennolla 0x02 pysäytettyyn tilaan.

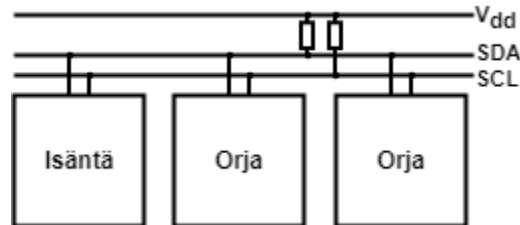
Mikrokontrolleri käyttää yhtä Transmit PDO:ta (TPDO) kiihtyvyyssanturilta lukemansa datan lähettämiseen PLC:lle. TPDO:t lähetetään Node ID:llä 0x182, missä 0x180 on CANopen:n ensimmäisen TPDO:n tunniste ja 0x02 mikrokontrollerin Node ID. kiihtyvyyssanturilta luettu data asetetaan TPDO:n datakenttään käännettyssä järjestyksessä, jossa se luettiin kiihtyvyyssanturilta, koska AC-500 PLC lukee datan vähiten merkitsevä tavu ensimmäisenä.

3.3 I²C-väylä

I²C-väylä on Philips Semiconductors:n (nyk. NXP Semiconductors) suunnittelema sarjaväylä, jota käytettiin alun perin samalla piirilevyllä olevien komponenttien välisessä kommunikoinnissa.

I²C-väylä toimii isäntä/orja -periaatteella, jossa isäntänä toimiva laite aloittaa kommunikoinnin orjana toimivan laitteen kanssa. Isäntälaitte myös generoi kommunikoinnissa tarvittavan kellosignaalin.

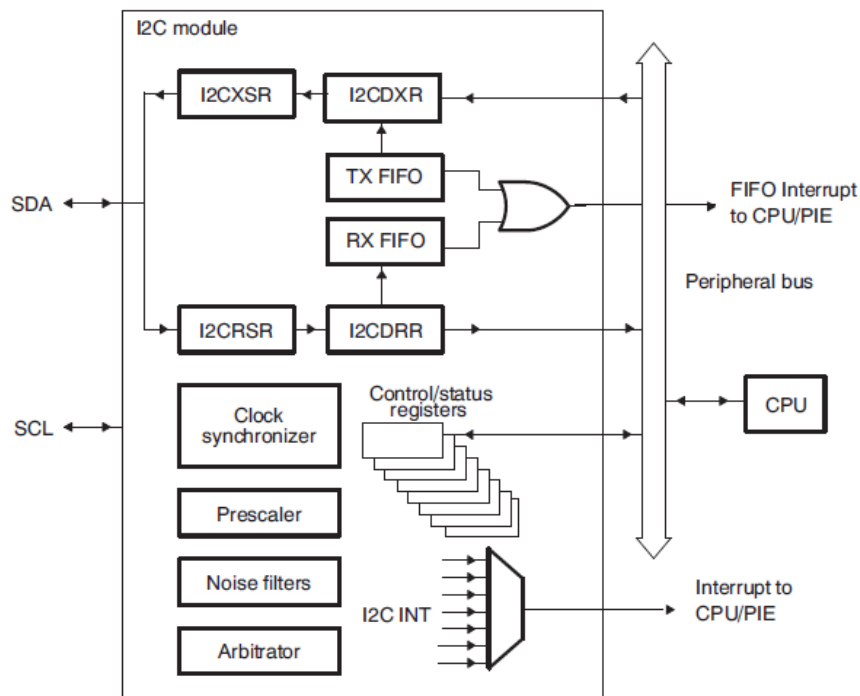
I²C-väylä koostuu kahdesta johtimesta. Datalinjasta (SDA) ja kellolinjasta (SCL). Väylä on half-duplex tyyppinen, eli datalinjalla voi kulkea dataa samanaikaisesti vain yhteen suuntaan. Data- ja kellolinjat tarvitsevat hyvin toimiakseen ylösvetovastukset. I²C-väylä on esitetty kuvassa 3.6.



Kuva 3.6 Fyysinen I²C-väylä.

Kommunikoinnissa voidaan käyttää 7-, tai 10-bittistä osoitteistoa, jolloin väylässä voi olla samanaikaisesti 112 laitetta 7-bittisellä osoitteistolla ja noin 10 kertaa enemmän laitteita 10-bittisellä osoitteistolla.

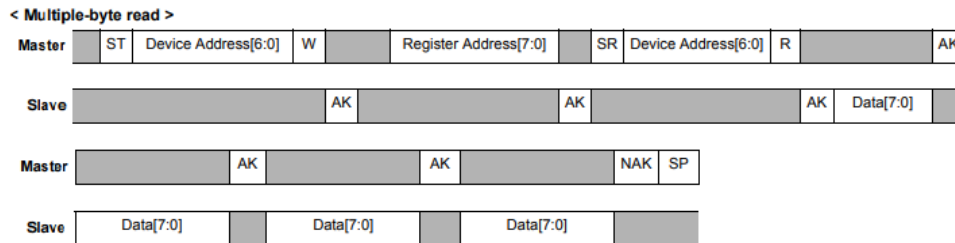
TMS320F28016 prosessorin I²C-moduuli tukee 7- ja 10-bittisiä osoitteita ja siinä on 16 viestin FIFO-puskuri (First In First Out) lähetetylle ja vastaanotetulle datalle. Moduuli kykenee luomaan keskeytyksen esimerkiksi silloin, kun vastaanottopuskuriin saapuu uusi viesti. I²C-moduuli lohkokkaavio on esitetty kuvassa 3.7. (TI I2C 2011)



Kuva 3.7 TMS320F28016P prosessorin I²C-lähetin-vastaanottimen lohkokkaavio (TI I2C 2011).

3.3.1 I²C-kommunikaatio kiihtyvyyssanturin kanssa

Kiihtyvyyssanturilta luetaan kaikkien kolmen akselin kiihtyvyydata yhdellä kertaa. Dataa luetaan siis yhteensä kuusi tavua ja anturi lähettää datan eniten merkitsevä bitti edellä. Usean tavun lukeminen kiihtyvyyssanturilta on esitetty kuvassa 3.9.



Kuva 3.9 Usean tavun lukeminen kiihtyvyyssanturilta I²C-väylää pitkin (NXP 2017).

Kiihtyvyyssanan lukeminen aloitetaan lähettämällä aloitusehto (ST, Start Condition) kiihtyvyyssanturin I²C-osoitteeseen 0x1D luku/kirjoitus bitti alhaalla. Anturi kuittaa tämän viestin (AK, Acknowledgement). Tämän jälkeen anturille lähetetään sen rekisterin soite, joka halutaan lukea. Kiihtyvyyssanaa luettaessa rekisterin muistiosoite on 0x02. Anturi kuittaa tämän viestin. Tämän jälkeen anturille lähetetään toistettu aloitus -ehto (SR, Repeated Start) ja anturin I²C-osoite luku/kirjoitus bitti ylhäällä. Anturi kuittaa viestin ja alkaa lähettää kiihtyvyyssanaa tavu kerrallaan. Mikrokontrollerin tarvitsee kuitata jokainen tavu vastaanotetuksi. Datan lukeminen loppuu, kun mikrokontrolleri ei enää kuittaa (NAK, Not Acknowledged) vastaanotettua dataa ja lähettää pysäytysehdon (SP, Stop) väylälle.

4. YHTEENVETO

Työssä suunniteltiin ja toteutettiin kiihtyvyyden mittausjärjestelmä, joka kykenee kommunikoimaan ABB:n AC-500 PLC:n kanssa CAN-väylän välityksellä CANopen-protokollaa käyttäen. CANopen-protokollasta toteutettiin tarvittavat osat mikrokontrollerille.

Järjestelmään valitut komponentit ovat suunniteltujen käyttökohteiden kannalta sopivia ja järjestelmän tarkkuus on käyttökohteiden kannalta vähintään riittävä. Nähtäväksi jää miten järjestelmää tullaan todellisuudessa hyödyntämään.

Järjestelmän toiminta jäi hieman vajaaksi, sillä kiihtyvyyssanturin asetusten kirjoittamista I²C-väylän välityksellä ei tuntemattomasta syystä saatu toimimaan. Järjestelmää voisi kehittää muuttamalla mikrokontrollerin ajastin pohjaisen ratkaisun keskeytyslähtöiseksi systeemiksi, jossa kiihtyvyyssanturi lähettäisi keskeytyksen mikrokontrollerille aina, kun uusi näyte on valmis. Tämä ratkaisu olisi hieman puhtaampi, eikä mikrokontrollerin tarvitsisi turhaan pyöriä loppumattomassa silmukassa. Järjestelmästä saisi tehtyä myös yksinkertaisemman ja kooltaan huomattavasti pienemmän, jos käytetty Olimex:n kehitysalusta korvattaisiin itse suunnitellulla piirilevyllä, jossa olisi vain järjestelmän kannalta kriittiset ominaisuudet.

LÄHTEET

Adafruit. [Verkkodokumentti]. [Viitattu 24.11.2017]. Saatavissa <https://www.adafruit.com/product/2019>

CAN in Automation (CiA). [Verkkodokumentti]. [Viitattu 24.11.2017]. Saatavissa <http://www.can-cia.org/can-knowledge>

CiA CANopen. [Verkkodokumentti]. [Viitattu 24.11.2017]. Saatavissa <https://www.can-cia.org/canopen/>

can-wiki. [Verkkodokumentti]. [Viitattu 10.3.2018]. Saatavissa <http://can-wiki.info/doku.php>

Fraden, J. 2010. Handbook of Modern Sensors: Physics, Designs, and Applications. Fourth Edition. New York: Springer

J.-H. Montonen, N. Nevaranta, T. Lindh, J. Alho, P. Immonen, O. Pyrhonen. 2018. Experimental Identification and Parameter Estimation of the Mechanical Driveline of a Hybrid Bus. IEEE Trans. Ind. Electron., vol. 65, no. 7, s. 5921–5930

Newton, I. 1687. Philosophiae Naturalis Principia Mathematica. First Edition. Lontoo

NXP, 2017, MMA8451Q, 3-axis, 14-bit/8-bit digital accelerometer. Datalehti

Olimes. [Verkkodokumentti]. [Viitattu 15.4.2017]. Saatavissa <https://www.olimes.com/Products/DSP/Development/TMS320-P28016/>

TI CAN, 2016, Texas Instruments Introduction to the Controller Area Network (CAN)

TI eCAN, 2009, Texas Instruments TMS320x280x/2801x Enhanced Controller Area Network (eCAN) Reference Guide. Datalehti

TI I2C, 2011, Texas Instruments TMS320x280x, 2801x, 2904x Inter-Integrated Circuit (I2C) Module Reference Guide. Datalehti

Wikipedia. [Verkkodokumentti]. [Viitattu 4.9.2017]. Saatavissa https://en.wikipedia.org/wiki/CAN_bus