

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Thomas Johann Seebecki elektroonikainstituut

Marek Aare

Piccolo F28069 A/D muunduri omaduste hindamine

Bakalaureuse lõputöö

IEE40LT

Juhendaja: vaneminsener Eero Haldre

Tallinn 2015

Autorideklaratsioon

Deklareerin, et käesolev lõputöö on minu iseseisva töö tulemus ning kinnitan, et esitatud materjalide põhjal ei ole varem akadeemilist kraadi taotletud. Kinnitan, et antud töö koostamisel on kõikide teiste autorite seisukohtadele, probleemipüstitustele, kogutud arvandmetele jmt viidatud.

Kuupäev:

Autor: Marek Aare

.....

/allkiri/

Lõputöö ülesanne

Lõputöö teema:

Piccolo F28069 A/D muunduri omaduste hindamine

Evaluation of Piccolo F28069's A/D Converter Properties

Teema päritolu: Tallina Tehnikaülikooli elektroonikainstituut.

Lõputöö eesmärgid: Õppelabori tingimustes muunduri testimine; anda ülevaade analoog-digitaalmuunduri tööst; kirjeldada erinevaid analoog-digitaalmuunduri arhitektuure ja parameetreid; mikrokontrolleri programmeerimine soovitud andmete saamiseks

Lahendatavad küsimused: Hinnang muunduri katsetamisest õppelaboris; kirjutada arvutis kood, millega on võimalik lugeda ja töötleda soovitud andmeid

Kuupäev:

Üliõpilane:

/allkiri/

Juhendaja:

/allkiri/

Kinnitaja:

/allkiri/

Annotatsioon

Bakalaureuse lõputööks on Piccolo F28069 analoog-digitaalmuunduri omaduste hindamine õppelabori keskkonnas. Töö sisaldab muunduri tööpõhimõtte kirjeldust, tehtud katsete mõõteskeeme ja arvutusi.

Töö eesmärk on hinnata kui edukalt on võimalik analoog-digitaalmuundurit uurida olemasolevate seadmetega. Muundurist oleks vaja saada soovitud andmeid, et veenduda seadme korrektsest töötamisest.

Töö käigus uuriti analoog-digitaalmuunduri tööpõhimõtet, arhitektuurilisi lahendusi ning testimis parameetereid. Andmete töötlemiseks oli vaja luua programm, mille abil oli võimalik mikrokontrollerit juhtida.

Töö tulemuseks oleks saada hinnang õppelabori võimalustest muunduri testimisel.

Bakalaureusetöö on kirjutatud eesti keeles ning sisaldab 37 lehte , 17 joonist , 12 graafikut, 3 fotot, 3 tabelit ja 1 lisa.

Abstract

The topic of this thesis is the evaluation of Piccolo F28069 analog-digital converter properties in a school laboratory setting. The thesis includes a description of the basic functions of a converter, schematics for the performed tests and calculations.

The aim was to evaluate the possibilities for converter testing in a school laboratory with existing equipment. Specific converter data was needed to make sure the device was working properly.

The working principles as well as architectural solutions and testing parameters of an analog digital converter were studied. A computer program was created to enable control of the microprocessor.

The outcome would be an assessment of the possibilities for testing a converter in the school laboratory.

This thesis was written in Estonian and contains 37 pages , 17 drawings , 12 graphs, 3 photos ja 3 tables and 1 appendix

LÜHENDITE SELETUS

- ADM- analoog-digitaal muundur , inglise keeles ADC(analog digital converter) või A/D
- AQCPs- inglise keeles Acquisition prescale, kontrollib S/H akna pikkust.
- DAM- digitaal-analoog muundur, inglise keeles DAC(digital analog converter)
- DMA- Direct Memory Access, on tehnika, mille abil on saab andmeid mällu edastada, ilma et nad protessorist läbi läheks
- EOC- end of conversion käsk, mis lõpetab muundamist.
- ePWM- inglise keeles enhanced Pulse width modulator, täiustatud impulss laius modulaator
- F28069 – Texas instrumentsi mikroprotsessor
- GND- ground, maandus
- IN- input, sisend
- LSB- inglise keeles least significant bit, vähima kaaluga bit
- MSB- inglise keeles most significant bit, suurima kaaluga bit
- MSPS- mega samples per second
- PC - Personal Computer, personaalarvuti
- RMS- Root mean square , ruutkeskmine
- SARAM- single-access memory, ühese juurdepääsuga mälu
- S/H- sample- and-hold, seade, mis samplib signaali ja hoiab seda.
- SOC- start of conversion, käsk, mis alustab muundamist
- USB- Universal Serial Bus, universaalne järjestik andmesiid

Jooniste ja Tabelite nimekiri

Joonis 1.1. Sample-and-Hold üldskeem.....	11
Joonis 1.2 Sampling ja kvantimine.....	11
Joonis 1.3 8-bitine Vilk-ADM.....	12
Joonis 1.4 Sammlähendus ADM.....	13
Joonis 1.5 Hierarhiline ADM , kus on neli 3-bitist astet.....	13
Joonis 1.6 Sigma-delta ADM.....	14
Joonis 2.1 3-bitise ADM-i ideaal karakteristik.....	15
Joonis 2.2 Positiivne Offset error.....	16
Joonis 2.3 Negatiivne Offset error.....	16
Joonis 2.4 Positiivne ja negatiivne Gain Error.....	16
Joonis 2.5 Differentiaalne mittelineaarsus.....	17
Joonis 2.6 Integraalne mittelineaarsus.....	17
Joonis 3.1 Piccolo F28069 Controlstick.....	19
Joonis 3.2 F28069 ADM-i plokskeem.....	19
Joonis 4.1 Mõõteskeem staatilise karakteristikü mõõtmiseks.....	21
Tabel 4.2 Staatilise karakteristikü tulemused kümnendkoodis.....	22
Graafik 4.3 Staatiline karakteristik.....	23
Foto 4.4 Foto staatika kontrolli katsest.....	23
Graafik 4.5 Müra karakteristik kui DAM-i ja ADM-i maandused on ühendatud.....	24
Graafik 4.6 Müra karakteristik kui DAM-i ja ADM-i maandused on lahti ühendatud.....	24
Graafik 4.7 Toitepinge müra.....	25
Joonis 4.8 Mõõteskeem patareiga testimiseks.....	26
Foto 4.9 ADM-i testimine patareiga.....	26
Tabel 4.10 ADM-i taktsagedust kontrollivate bitide väärtuste tabel.....	27
Tabel 4.11 ACQPS võimalikud väärtused.....	27
Graafik 4.12 ADM-i müra kui ACQPS=6.....	28
Graafik 4.13 ADM-i müra kui ACQPS=63.....	28
Graafik 4.14 ADM-i müra kui taktsagedus on 90 MHz.....	29
Graafik 4.15 ADM-i müra kui taktsagedus on 22,5 MHz.....	29

Joonis 4.16 ADM-i kanalite ülekostvuse mõõteskeem.....	30
Graafik 4.17 ADM-i müra, kui väline signaal on sama multipleksori kanalis, sagedus 2MHz.....	30
Graafik 4.18 ADM-i müra, kui väline signaal on teise multipleksori kanalis, sagedus 2MHz.....	30
Graafik 4.19 ADM-i müra, kui väline signaal on sama multipleksori kanalis, sagedus 1MHz.....	31
Graafik 4.20 ADM-i müra, kui väline signaal on teise multiplekstoru kanalis, sagedus 1 MHz.....	31
Foto 4.21 Signaaligeneraatori väljund.....	31

Sisukord

Sissejuhatus	10
1. Analoo-Digitaalmuunduri ülevaade	11
1.1 Analoo-digitaalmuunduri üldine tööpõhimõte	11
1.2 Analoo-digitaalmuunduri arhitektuurid	12
1.2.1 Vilk-ADM (Flash ADC).....	12
1.2.2 Sammlähendus ADM (Successive Approximation ADC)	12
1.2.3 Hierarhiline ADM (Subranging pipelined ADC)	13
1.2.4 Sigma-delta ADM (Sigma-delta ADC)	13
2. Analoo- Digitaalmuunduri parameetrid	15
2.1 Staatilised Parameetrid	15
2.1.1 Resolutsioon (Resolution)	15
2.1.2 Nullnihke viga (Offset Error)	15
2.1.3 Võimendusteguri viga (Gain Error).....	16
2.1.4 Differentiaalne mittelineaarsus (Differential Non-Linearity)	17
2.1.5 Integraalne mittelineaarsus (Integral Non-Linearity)	17
2.2. Dünaamilised Parameetrid.....	18
2.2.1 Total Harmonic Distortion (Täielik harmooniline moonutus).....	18
2.2.2 Signal-to-Noise ratio (Signaali-müra suhe).....	18
2.2.3 Signal-to-Noise and Distortion (signaali-müra ja moonutus suhe)	18
2.2.4 Effective Number of Bits (efektiivsete bitide arv)	18
3. Piccolo F28069 Analoo-digitaalmuundur	19
4. F28069 Analoo-digitaalmuunduri parameetrite mõõtmine	21
4. 1. Staatilise karakteristiku mõõtmine DAM-i abil	21
4.2. ADM-i ühendamine maandusega	25
4.3. ADM-i ühendamine oma toitepingega	25
4.4. ADM-i testimine patareiga	26
4.5. ADM-i kanalite ülekostvuse testimine	29
Kokkuvõte	32
Kasutatud kirjandus	33
Lisad.....	34

Sissejuhatus

Analoog-digitaal muundur on seade, mis eksisteerib peaaegu igas süsteemis, kus elektroonika omab mingit osa. Meie maailm on aina rohkem digitaalsemaks minemas ja ADM on selle digitaalse maailma südameks. Enamik andmevahetus ning töötlus toimub arvutis või mitme arvuti vahel. ADM-i on vaja, et meie looduslikku signaali esitata arvutile meelepärasel kujul.

Kuna ADM on nii tähtis meie tänapäeva maailmas, tekkis mul huvi ja tahtime seda seadet uurida. Soovisin teada kui lihtne oleks sellise digitaalmaailma fundamentaalse osage töötada. Kui lihtne on sellest seadest aru saada ning kui kerge on teda käsitleda, et soovitud tulemusi saada. Oleks vaja teada, kui keerukaid ning täpseid mõõteriistu on vaja, et saada ADM-ist objektiivne arvamus.

Katse all olev ADM peaks olema piisavalt lihtne, et sellega oleks võimalik tudengil õppelaboris katseid teha ja uurida. ADM-iks valisin Texas instrumendi mikrokontrolleri Piccolo F28069, kuna see oli väikse maksumusega seade ja teda oli piisavalt suures koguses ülikoolis olemas, et mõnes õppeaines on võimalik neid tudengitele laiali jagada ning katseid nendega teha.

Töö eesmärk oli anda hinnang kui lihtne on Piccolo F28069 ADM-i uurida õppelaboris. Esiteks tegin endale selgeks üldised ADM-i tööpõhimõtted, siis uurisin missugused arhitektuurilised lahendused esinevad ADM-ides. Teiseks kirjeldasin mõned levinumad parameetrid, millega ADM-i katsetamisel kokku puutub. Kolmandas peatükis tutvusin lähemalt F28069 ADM-iga ja tõin välja mõned tähtsamad tehnilised näitajad, mida see seade omab.

Töö käigus oli vaja õppida ka veidi mikrokontrolleri programmeerimist. Texas Instruments oli mikrokontrolleriga kaasa andnud CD plaadi, mille peal oli programmeerimis keskkonna installeerimis failid ning mõned näidisprogrammid.

Neljandas peatükis kirjeldan ma katseid, mida ma suutsin õppelaboris Piccolo F28069-ga teostada.

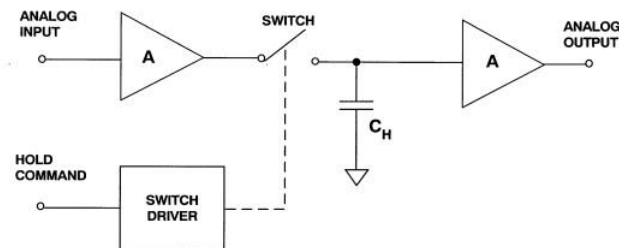
1. Analoo-Digitaalmuunduri ülevaade

1.1 Analoo-digitaalmuunduri üldine tööpõhimõte

ADM on seade, mis tõlgib meie analoog maailma väärtused digitaalseks. Selle abil on meil võimalik arvutitega signaali töötleda, arvutada ja kasutada juhtimis süsteemides. Selleks, et esitada analoog väärtused digitaalselt on vaja teha kolm funktsiooni: 1) Sampling 2) Kvantimine 3) Kodeerimine [3]

Sampling on signaali töötlemis protsess, mis muudab pideva signaali diskreetseks signaaliks. Samplimis ajal muudetakse ajatelg diskreetseks. Samplimis sagedus või kiirus on võrdne keskmise samplite arvuga ühe sekundi jooksul ehk $f_s = 1/T$, kus f_s on samplimis sagedus ja T on periood. Mida rohkem sampleid võetakse, seda täpsem on digitaalne esitus.[3]

Enamik ADM-id omavad Sample-and-Hold (S/H) seadet. Selle seadme ülesanne on samplida sisend signaali ja hoida seda kondensaatoris, et muundamist läbi viia. See seade koosneb tavaliselt sisendvõimendist, lülitist, kondensaatorist ja väljundvõimendist. Väljundvõimendi omab piisavalt suurt sisend impedantsi, et hoida kondensaatorit liiga kiiresti tühjaks jooksmast.[7]

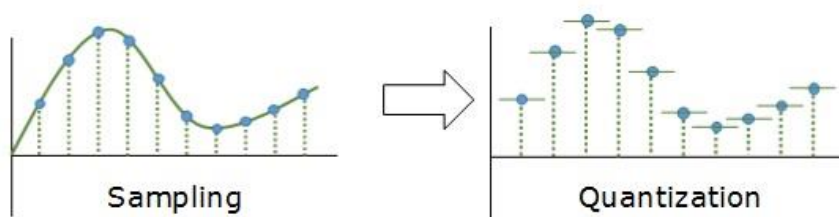


Joonis 1.1. Sample-and-Hold üldskeem. [7]

Signaali töötlemisel on vaja täita Shannon-Nyquisti teoreem, mis ütleb, et analoog signaali, mille ribalaius on f_a , tuleb samplida kiirusega mis on vähemalt kaks korda suurem ($f_s > 2f_a$), et ära hoida informatsiooni kadu [1]

Kvantimine on protsess, mis võtab pideva signaali amplituudi väärtused ja muudab nad diskreetseteks väärtusteks. Sisendiks võivad olla loendamatud piirkonnad või väärtused, mida tuleb kvantida loendatavaks väärtuseks. Esineb viga, kus tegelik analoog väärtus ja digitaalväärtus võivad erineda kuni $\frac{1}{2}$ LSB, mida nimetatakse kvantimis veaks või kvantimis määramatuseks.[3] Sellega kaasnevad olukorrad, kus mitmele erinevale sisenditele on ümardamise tagajärjel omandatud üks ainuke diskreetne väärtus, mis tähendab, et väljund väärtusest originaali on võimatu taastada.

Kui sampling ja kvantimine (vt joonis 1.2) on tehtud siis tulemused kodeeritakse sobivale kujule, mis tavaliselt on binaarkood.



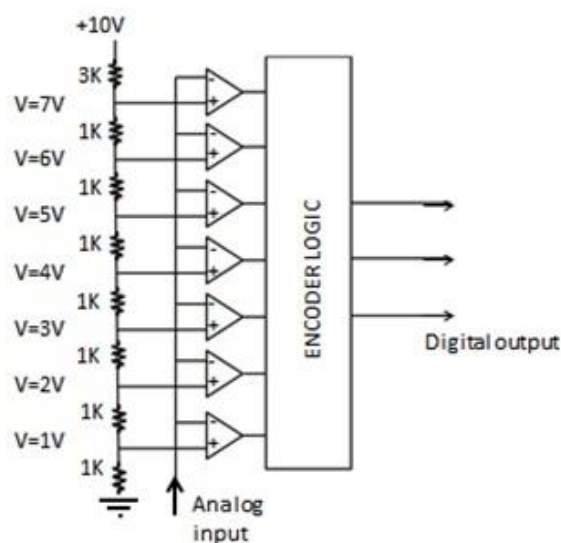
Joonis 1.2 Sampling ja kvantimine.

1.2 Analoo-digitaalmuunduri arhitektuurid

ADM-id omavad mitmeid erinevaid arhitektuure. Need arhitektuurid võimaldavad muunduritel saavutada soovitud parameetreid nagu kiirus, madal hind, kindel sagedusriba või väike võimsus. Esinevad neli tüüparhitektuuri, millest enamik arhitektuure on välja töötletud. Need on vilk, sammlähendus, hierarhiline ja sigma-delta ADM.

1.2.1 Vilk-ADM (Flash ADC)

Vilk-ADM on kõige kiirem muunduri tüüp. Tema kiirus tuleb sellest, et kõik biti valikud tehakse üheaegselt. N-bitine ADM koosneb 2^n takistist ja 2^{n-1} komparaatorist. Komparaatorid saavad takistite jadast tugipinge, mis on 1 LSB võrra kõrgem kui ahela alumisel. Kui sisendpinge on kõrgem kui tugipinge, siis komparaatori väljund on loogiline „1“. Kui sisendpinge on madalam kui tugipinge, siis on väljund loogiline „0“. Kuna 2^{n-1} andmeväljundit ei ole praktilised, siis dekodeer genereerib N-bitise kahendväljundi. Vilk-ADM-il on piiratud eraldusvõime, suur võimsusekadu ja suured mõõtmed. [2]



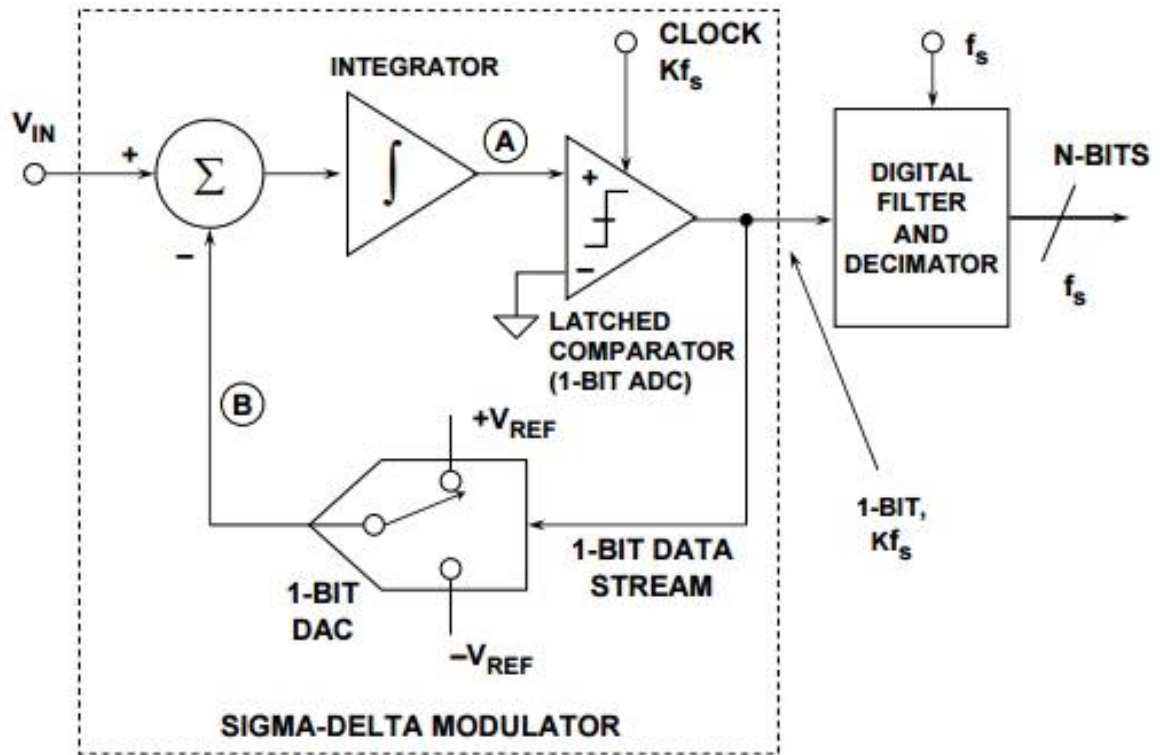
Joonis 1.3 8-bitine Vilk-ADM

1.2.2 Sammlähendus ADM (Successive Approximation ADC)

Sammlähendus ADM-iga võrreldakse tundmatut sisendit etalon tugipingetega. Võrdlemine toimub kindlas järjekorras alustades kõige suuremast. Sammlähendus ADM-i saab oma nime sammlähendus registrilt (SAR), mis juhib teisendusprotsessi. ADM saab käsu SOC (start of conversion), mis seab S/H seadme hoiderežiimi. Kõik registri bitid, välja arvatud MSB, on seatud loogika väärtusele „0“. Registri väljund käivitab sisemise DAM-i. [2]

Kui analoogsisendi väärtus on suurem kui DAM-i väljund, siis registri bit jäetakse loogilisele väärtusele „1“. ADM seab registri järgmise MSB loogilisele väärtusele „1“ ja võrdleb seda analoogsisendiga. Kui analoogsisendi väärtus on väiksem kui DAM-i väljund, siis registri teine MSB muudetakse loogilisele väärtusele „0“. Protsess kordub kõikide registri bittidega kordamööda, kuni on saavutatud lõpptulemus ning muundus on lõppenud. Muundamise lõppu näitab signaal EOC (end of conversion). [2]

Võrreldes vilk-ADM-iga on sammlähendus ADM madalama energiatarbega ja väiksema mõõtmeaga.



Joonis 1.6 Sigma-delta ADM

2. Analoo- Digitaalmuunduri parameetrid

ADM oma mitmeid parameetrid, mis pakuvad meile huvi. Need parameetrid on üldiselt jaotatud kahte kategooriasse : staatilised parameetrid ja dünaamilised parameetrid.

2.1 Staatilised Parameetrid

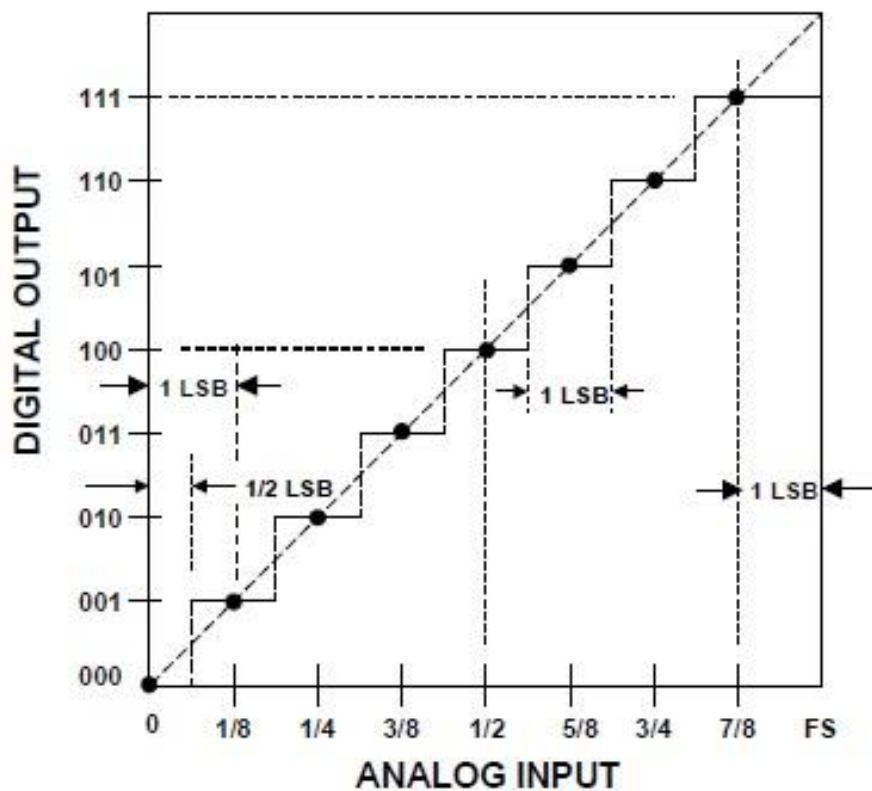
Staatilised parameetrid on sellised, mida on võimalik testida aeglastel kiirustel või konstantse pinges allikaga.

2.1.1 Resolutsioon (Resolution)

Resolutsioon näitab mitu diskreetset väärtust saab muundur väljastada. Resolutsiooni arvutatakse valemiga 2^n , kus n on bitide arv.

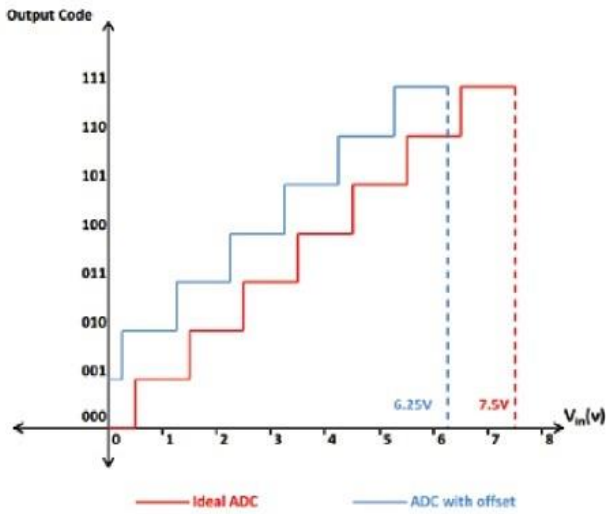
2.1.2 Nullnihke viga (Offset Error)

Offset error on viga, mis on määratud esimese ülemineku pinge ja ideaalse ülemineku pinge erinevusega. Ideaalselt toimub esimene üleminek pingel, mis on võrdne $\frac{1}{2}$ LSB-ga. [10]

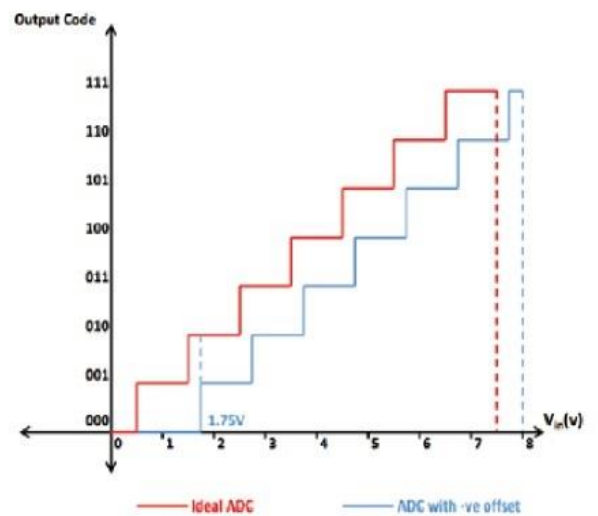


Joonis 2.1 3-bitise ADM-i ideaal karakteristik.

Offset error võib olla positiivne või negatiivne.



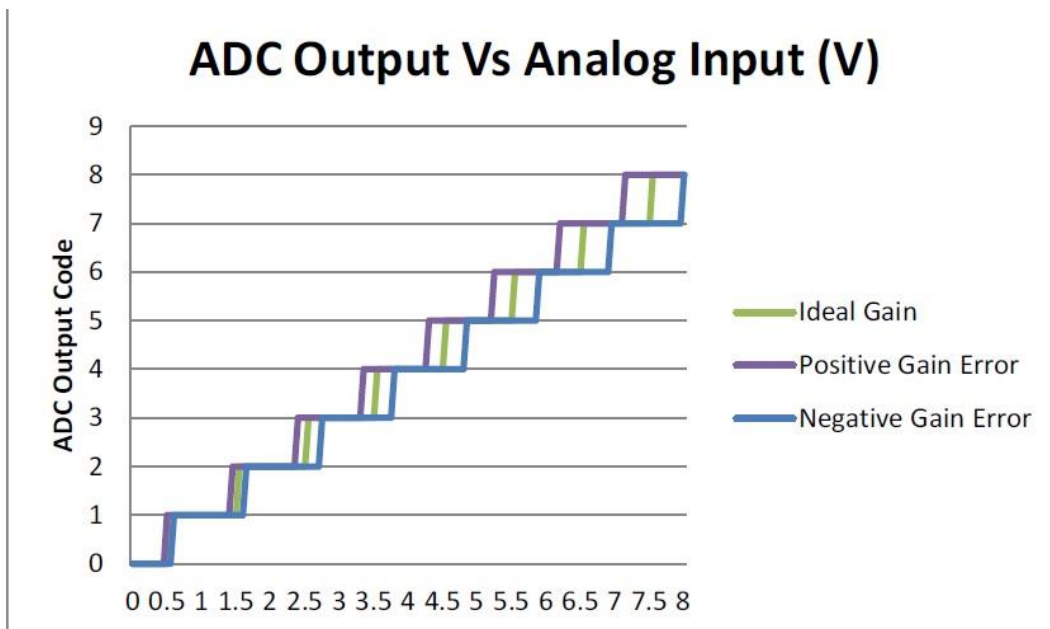
Joonis 2.2 Positiivne nullnihke viga



Joonis 2.3 Negatiivne nullnihke viga

2.1.3 Võimendusteguri viga (Gain Error)

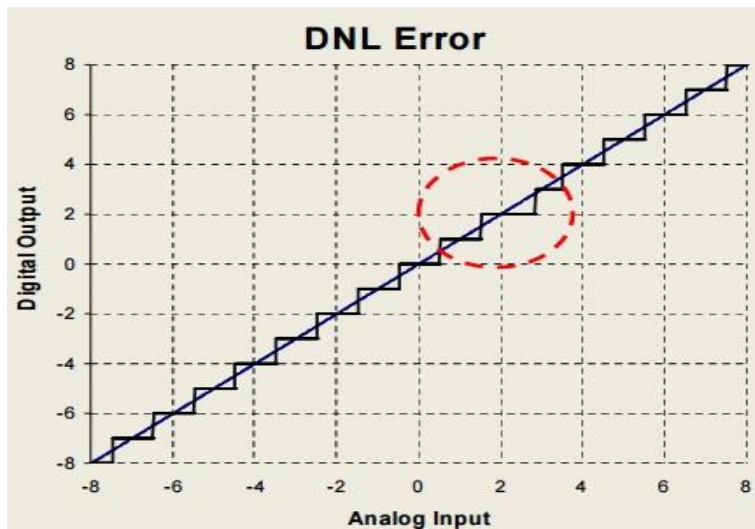
Gain error on viga, mis on määratud ideaalse ja tegeliku pinge vahega, mis tekitab täiemõdulise väljund koodi (Full scale output code ehk kuuteistkümnendsüsteemis 0xFFFF). See tekitab olukorra, kus ADM saab maksimum digitaalväärtuse liiga hilja või liiga vara.



Joonis 2.4 Positiivne ja negatiivne Gain Error.

2 1.4 Differentiaalne mittelineaarsus (Differential Non-Linearity)

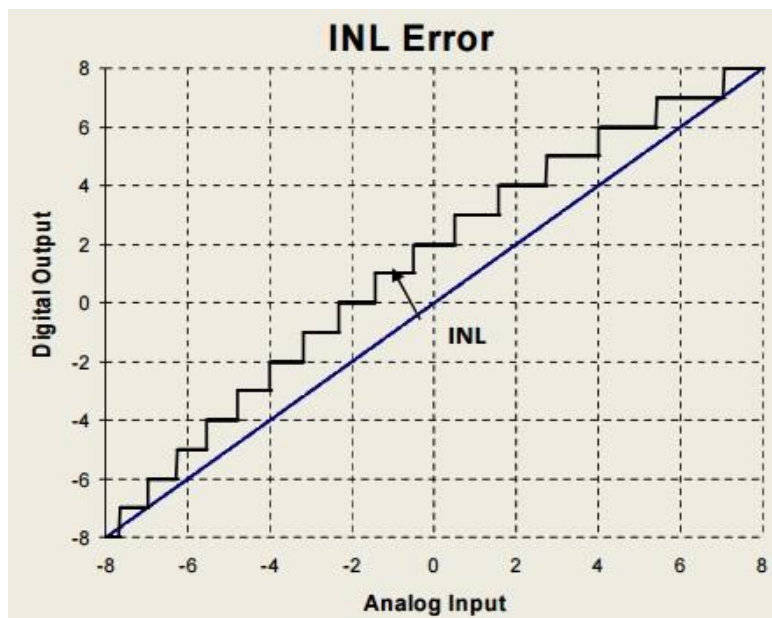
Differentiaalne mittelineaarsus(vt joonis 2.5) on koodi laiuse hälve ideaalsest laiusest (1 LSB) .



Joonis 2.5 Differentiaalne mittelineaarsus.

2 1.5 Integraalne mittelineaarsus (Integral Non-Linearity)

Integraalne mittelineaarsus(vt joonis 2.6) on määratud reaalse karakteristiku maksimaalse hälbe ga sirgjoonest. ADM-i puhul läbib see sirgjoon tavaliselt ideaal karakteristiku koodi keskpunkte.



Joonis 2.6 Integraalne mittelineaarsus.

2.2. Dünaamilised Parameetrid

Dünaamilisi parameetreid testitakse kiiremate signaalide puhul. Nende puhul kasutatakse enamasti FFT-d (Fast Fourier Transform).

2.2.1 Total Harmonic Distortion (Täielik harmooniline moonutus)

Täieliku harmoonilise moonutuse parameeter annab meile informatsiooni harmoonilisest energiast, mis asub kindla sisend sageduse spektril. Seda sagedus spektrit näidatakse tavaliselt kuni Nyquisti sageduseni (See on pool sãmplimissagedust). Muunduri mittelineaarsus tekitab harmooniaid, mida originaal signaalis ei olnud. Väljendatakse tavaliselt protsentuaalse suhtena või V_{rms} . [4]

$$THD = \frac{\sqrt{V_2^2 + V_3^2 + \dots + V_n^2}}{V_1}$$

V_1 – põhiharmooniline rms ping

V_n – kõrgemate harmooniliste rms ping

2.2.2 Signal-to-Noise ratio (Signaali-müra suhe)

Signaali-müra suhe annab informatsiooni sisend sageduse müra energiast. SNR arvutusel eiratakse signaali spektris olevat harmoonilist energiat. Ühikuks on dB, V_{rms} või %.[4]

$$SNR(dB) = 20 \log \left(\frac{V_{RMS(Signal)}}{V_{RMS(Noise)}} \right)$$

2.2.3 Signal-to-Noise and Distortion (signaali-müra ja moonutus suhe)

Signaali-müra ja moonutus suhe annab informatsiooni sisend signaali ja müra koos moonutusega suhtest. Ühikuks on tavaliselt dB. [4]

$$SINAD(dB) = 10 \log_{10} \left(\frac{P_S}{P_N + P_D} \right)$$

P_S – signaali võimsus

P_N - müra võimsus

P_D – moonutus võimsus

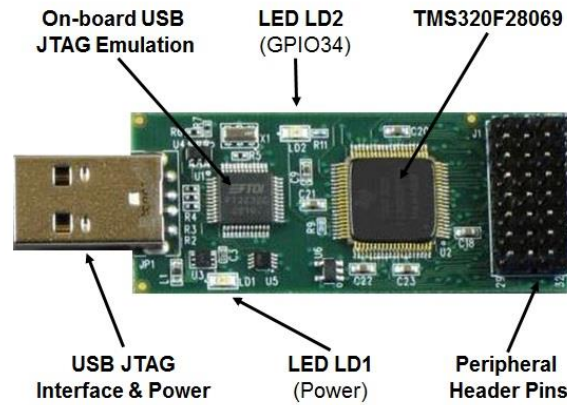
2.2.4 Effective Number of Bits (efektiivsete bitide arv)

Efektiivsete bitide arv on viis kuidas signaali-müra ja moonutus suhet kirjeldada bitides.[4]

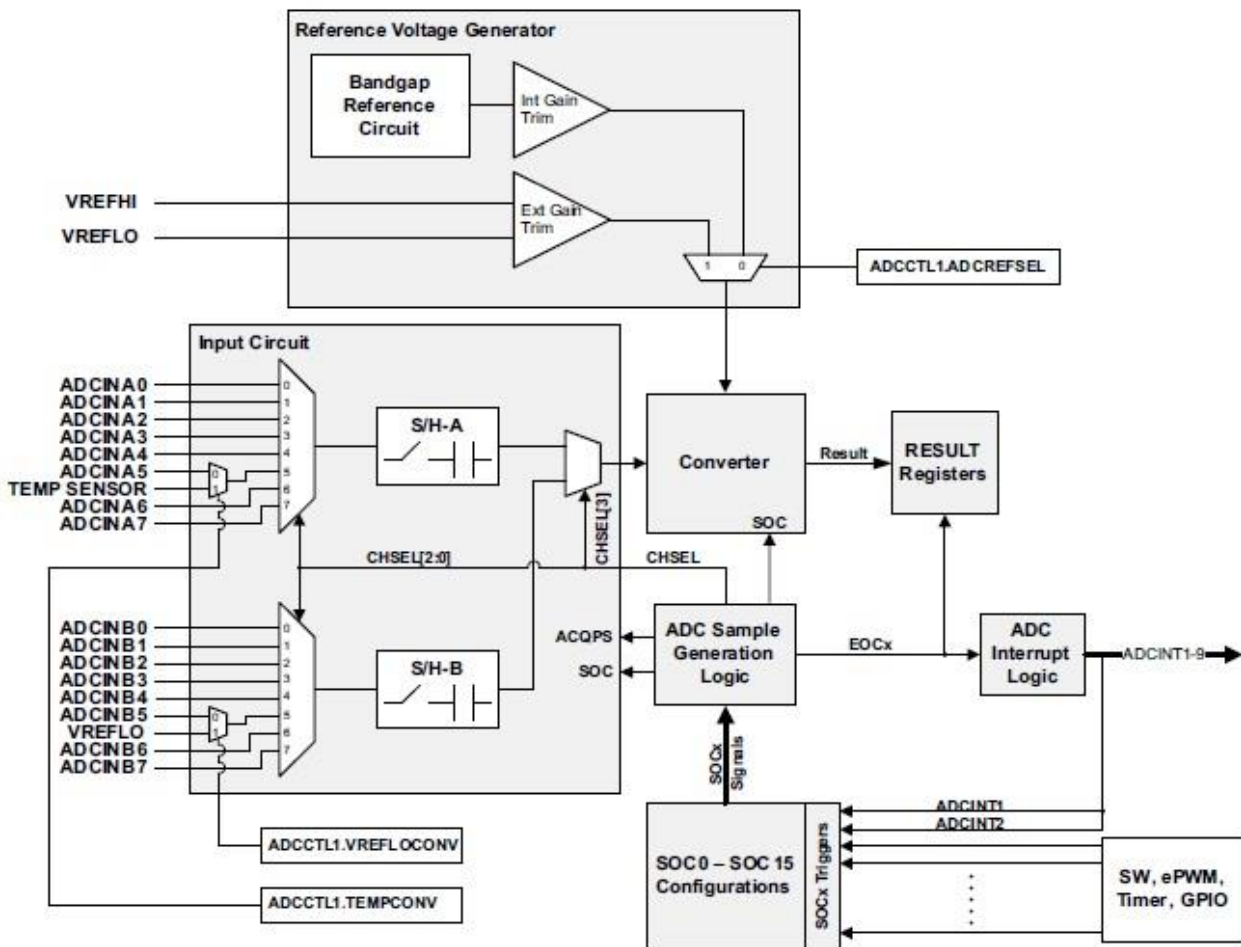
$$ENOB = \frac{SINAD - 10 \cdot \log \left(\frac{3}{2} \right)}{20 \cdot \log(2)} = \frac{SINAD - 1.76}{6.02}$$

3. Piccolo F28069 Analoog-digitaalmuundur

Antud bakalaureusetöös valisin ma Texas Instrumentsi Piccolo F28069 ADM-i oma katseseadmeks. F28069 mikrokontroller (vt joonis 3.1) omab 32-bitist keskprotsessorit ja 12-bitist ADM-i. Selle mikrokontrolleri ADM omab tsüklilist arhitektuuri (vt joonis 3.2). Tsükliline arhitektuur on kombinatsioon sammlähenduse ADM-ist ja hierarhilisest ADM-ist.



Joonis 3.1 Piccolo F28069 Controlstick



Joonis 3.2 F28069 ADM-i plokkskeem [5]

Piccolo F28069 mikrokontroller omab 50 kB SARAM-i ja 128 kB flash mälu, mida on võimalik läbi JTAG-i programmeerida ja debugida. See mikrokontroller omab ka 6 kanalist DMA moodulit, mis võimaldab andmete ülekandmist ilma keskprotsessori osaluseta. See võimaldab protsessoril tegeleda andmete töötlemise asemel teiste süsteemidega. F28069 omab ka ePWM-i, mis töötab DAM-i rollis. PWM-i töötükkel käitub kui DAM-i analoog väljund. ADM-i muundus võtab aega 13 takti. F28069 saavutab kuni 3,46 MSPS.

Piccolo F28069 sisaldab muu hulgas[5]:

- Kaks S/H kanalit; Kanal A ja kanal B, mis võimaldab samaaegseid sampleid ja
- 0 V – 3,3 V analoogsisendi piirkonda
- Kuni 16 sisendi kanalit
- 16 tulemuse registrit, mida on võimalik individuaalselt adresseerida

ADM-is saab üksikuid muundusi kontrollida SOC konfiguratsiooniga. F28069 omab 16 SOC konfiguratsiooni, mida on võimalik eraldi üles seada. SOC konfiguratsioon koosneb kanali valiku bitist, ajendi valiku bitist ning S/H akna pikkuse bitist (ACQPS).

F28069 LSB väärtuse saame, kui jagame ADM-i täis skaala pinget ja resolutsiooni

$$\text{LSB} = V_{\text{fs}} / 2^n = 3,3 \text{ V} / 4096 = 0,00080 \text{ V} = 0,8 \text{ mV}$$

Seega üks bit on võrdne 0,8 mV.

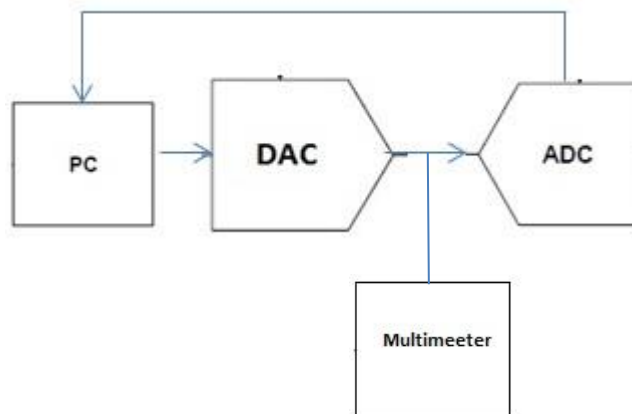
4. F28069 Analoo- Digitaalmuunduri parameetrite mõõtmine.

Tehnikaülikooli õppelaboris katsete tegemiseks Piccolo F28069 ADM-iga kasutasin ma järgmisi töövahendeid:

- Personaalarvuti, kuhu oli installitud Windows XP operatsioonisüsteem.
- Programmeerimis keskkond Code Composer Studio 4.2, millega sai programmeerida F28069 mikrokontrollerit.
- Disaini platformi National Instruments LabView Student edition, mille abil sai kontrollida personaalarvuti sees olevat 16 bitist digitaal- analoogmuundurit(DAM) ja selle muunduri abil genereerida soovitud analoog väljundeid.
- Ostsilloskoop Tektronix TDS 1001 signaalide kontrollimiseks.
- Multimeeter Hewlett-Packard 34401A
- Signaaligeneraator ITT TG550

4. 1. Staatilise karakteristiku mõõtmine DAM-i abil

Esimeseks katseks oli mul vaja luua olukord, kus arvuti sees olev DAM väljastaks kindla analoog väärtuse ning F28069 ADM loeks seda väärtust. Katse läbiviimiseks tinutasin 10 k Ω takisti juhtme otsa pistiku, millega ma saaks ühendada arvuti DAM-i väljundi ja F28069 ADM-i sisendi (vt foto 4.4). Takisti käitub voolu piirava ahelana, et kaitsta ADM-i.



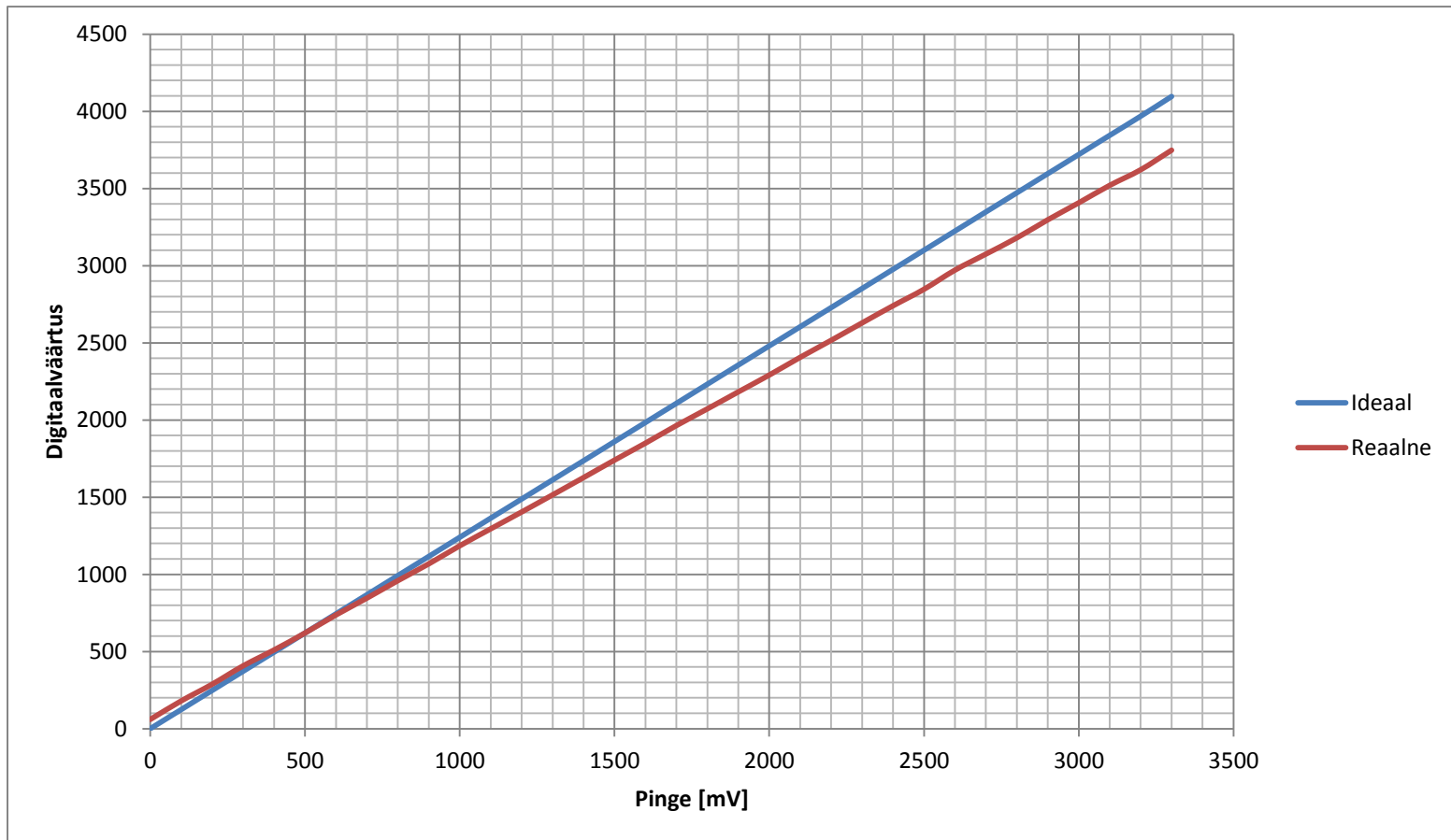
Joonis 4.1 Mõõteskeem staatilise karakteristiku mõõtmiseks

F28069 mikrokontroller on ühendatud arvutiga läbi USB liidese, mis täidab nii toite kui ka andmeedastus funktsiooni. Arvuti DAM-i kontrollisin ma National Instruments LabView programmiga „Gen Voltage Update.vi“, mis võimaldas mul genereerida kontsantset pinget. DAM-i pinget kontrollisin ma multimeetriga, et veenduda pinge väärtuses. Kasutades Code Composer Studio 4.2 näite projekti „ADC – Simple DAC to ADC“, sain ma lugeda ADM-i digitaalset väärtust. Code Composer Studio debugger omab mugavat akent nimega „Watch“, millega saab jälgida ning muuta soovitud parameetreid nagu näiteks registri või muutuja väärtusi. Vaadeldavaid parameetreid saab ka selle akna abil vormistada sobivas formaadis nagu binaar-, kümnend- või kuueteistkümnendkoodis. Kui kõik ühendused olid saavutatud, hakkasin ma DAM-i pinget 100 mV haaval muutma ja salvestasin vastavad ADM-i väärtused tabelisse. Mõõtepiirkond oli 0 V kuni 3,3 V. Tulemused on esitatud tabelis 4.2.

Tabel 4.2 Staatilise karakteristiku tulemused kümnendkoodis.

Pinge[mV]	Digitaalneväärtus (ideaalne)	Digitaalneväärtus (reaalne)
0	0	60
100	124	180
200	248	288
300	372	407
400	496	510
500	620	620
600	744	737
700	868	846
800	992	959
900	1116	1068
1000	1240	1185
1100	1364	1295
1200	1488	1404
1300	1612	1515
1400	1736	1627
1500	1860	1740
1600	1984	1850
1700	2108	1964
1800	2232	2072
1900	2356	2182
2000	2480	2291
2100	2604	2406
2200	2728	2516
2300	2852	2629
2400	2976	2740
2500	3100	2847
2600	3224	2971
2700	3348	3075
2800	3472	3180
2900	3596	3297
3000	3720	3407
3100	3844	3520
3200	3968	3621
3300	4095	3748

Tabelist võib näha, et katse all oleva ADM-i tulemused erinevad märgatavalt ideaalsest. Erinevus on ilmekamalt illustreeritud graafikul 4.3. Graafikult võib näha, et ADM omab selget negatiivset võimendusteguri viga (Gain error) ning nullnihke viga (offset error), mida õppelaboris on keeruline välja arvutada, kuna see nõuaks iga biti eraldi mõõtmist ehk siis 4096 mõõtmise tegemist.



Graafik 4.3 Staatiline karakteristik

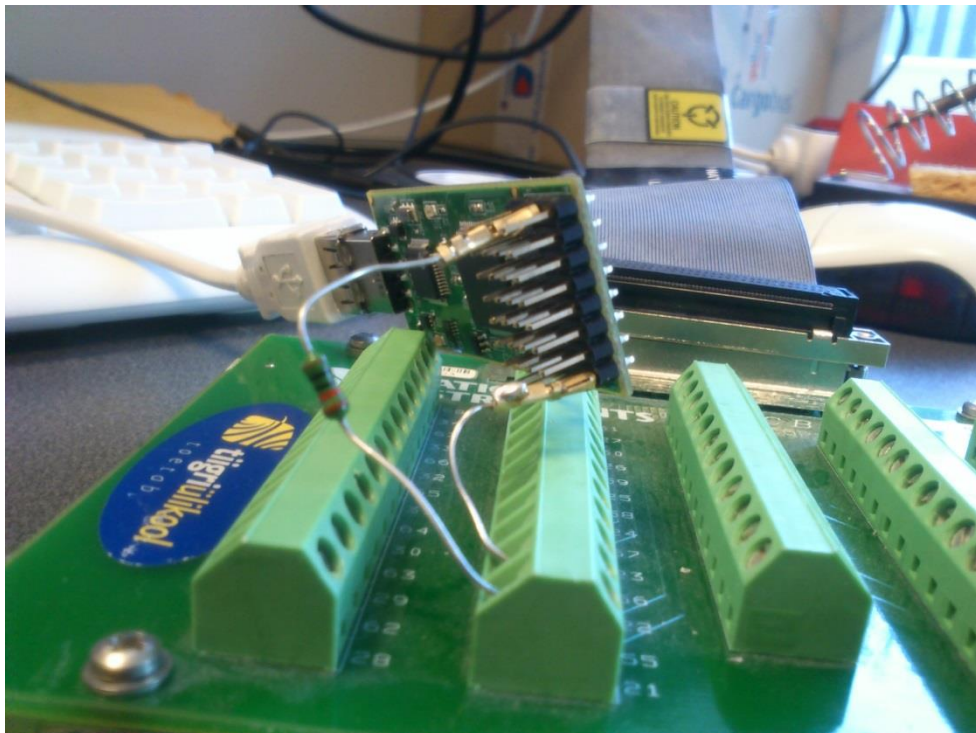
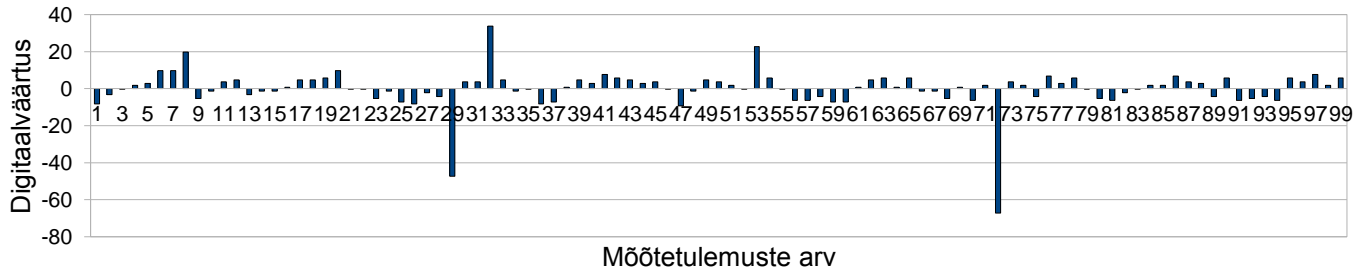
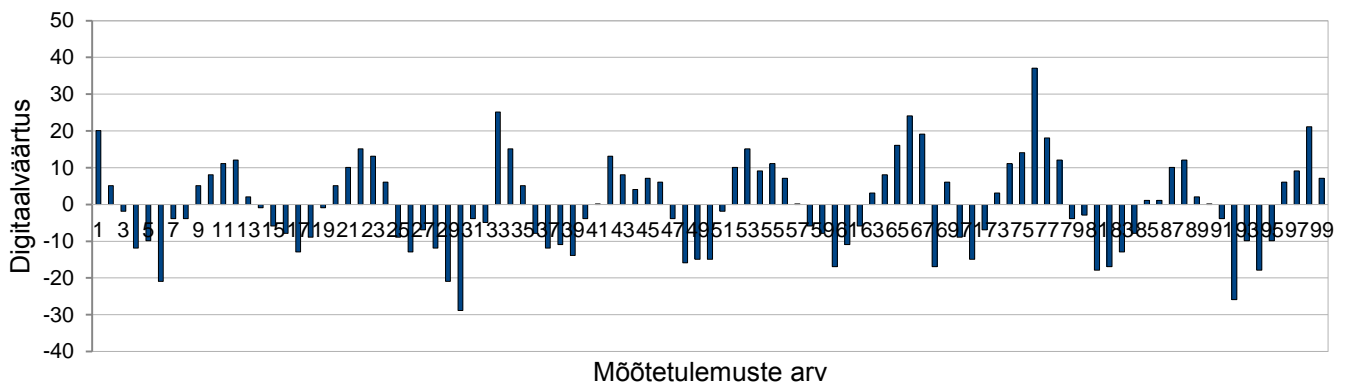


Foto 4.4 Foto staatika kontrolli katses

Katse läbiviimisel selgus, et DAM-i ja ADM-i ühendamisel tekib umbes 30 mV nihkepinge, mida võis lugeda multimeetrilt. Probleem võib olla selles, et mõõteskeem ei oma ühist maanduspunkti, vaid ta on maandatud kahest erinevast kohast. Üks maandus jookseb mööda USB liidest ja teine maanduse kontuur tuleb DAM-i väljundist. Uurisnin maanduste ühenduse mõju mürale kahe lisa katsega, kus ühel juhul jätsin ma maandused ühendatud ning teisel juhul ühendasin ma DAM-i maanduse ADM-ist lahti, jättes ainult USB maanduse. Personaalarvutis oleva DAM-i abil genereerisin konstantse pinge väärtusega 1.5 V ja lasin programmil teha 100 mõõtmist ning arvutasin standardhälve. Tulemused on esitatud graafikul 4.5 ja 4.6.



Graafik 4.5 Müra karakteristik kui DAM-i ja ADM-i maandused on ühendatud



Graafik 4.6 Müra karakteristik kui DAM-i ja ADM-i maandused on lahti ühendatud

Standardhälve valem
$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$$

x_i - muutuja väärtus kohal i

μ - keskvärtus

N – muutujate arv

Keskvärtus, kui maandused on ühendatud: AVERAGE= 1830,3

Keskvärtus, kui maandused on lahti ühendatud: AVERAGE= 1832,9

Standardhälve, kui maandused on ühendatud : $STDEV=10,5$

Standardhälve, kui maandused on lahti ühendatud: $STDEV=12,4$

Tuleb välja, et maanduste ühendamisel on müra väiksem. Seega õppelaboris on parem kasutada mõtteskeemi, kus DAM-i ja ADM-i maad on ühendatud, kuid väga täpseid tulemusi sellega ei saa.

Ebakõla reaalsete ja ideaalsete digitaalväärtuste vahel võib tuleneda ka kaablist, mis väljastab personaalarvuti DAM-i analoog signaali. Kaabel võib püüda õhust igasuguseid mürasid, mis moonutavad ADM-i tulemusi.

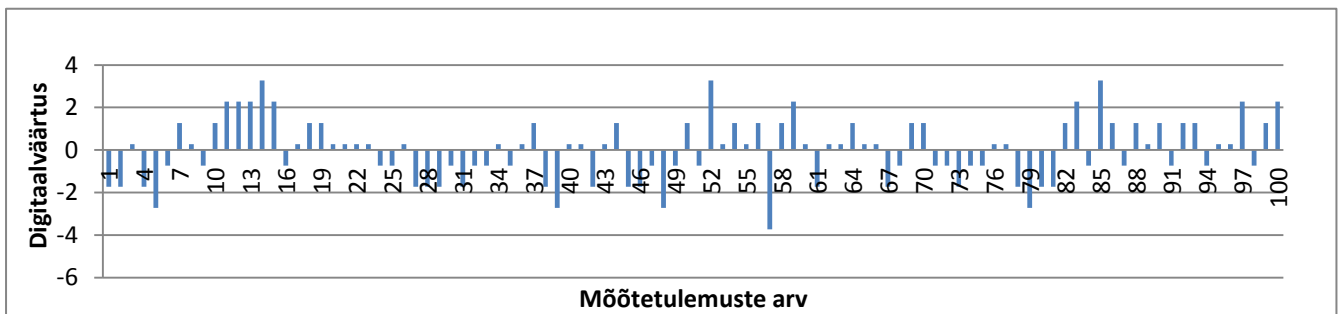
ADM-i staatiliste karakteristikute täpsemateks hindamiseks oleks õppelaboris vaja näiteks patarei või aku toitel DAM-i, mida saaks ühendada katse all olevasse ADM-i väheste või väga lühikeste juhtmetega, mis vähendaks keskkonnast tulenevat müra teket. Paraku sellist seadet meie õppalaboris ei leidu.

4.2. ADM-i ühendamine maandusega

Tegin katse, kus ADM-i sisend oli vaja ühendada maaga. Katse eesmärk oli tuvastada, kas esineb mingit müra maa ja sisendi vahel. Kasutades Texas Instrumenti poolt kaasa antud kaablit, ühendasin sisendi ja maanduse. Lasin programmil teha 1000 mõõtmist ja tuvastasin, et maa suhtes müra puudub. ADM-i väljastas konstantse 0 koodi oma tulemuse registris.

4.3. ADM-i ühendamine oma toitepingega

Selles katses ühendasin ma muunduri tema enda toitepingega, et leida sisendi ja toite vahel olev müra. Selleks on mikrokiibil toitepinge väljaviik, mis andis välja 3,3 V alalispinget. Veendusin multimeetriga, et pingeväärtus oli korrektselt 3,3 V. Pärast toitepinge kontrollimist ühendasin ma toite ja ADM-i sisendi. Lasin programmil teha 1000 mõõtmist ning nendest on 100 tulemust esitatud graafikus 4.7.



Graafik 4.7 Toitepinge müra

Graafikult võib näha, et toitepinge müra on võrdlemisi väike. Tulemustest arvasin ka keskvaertuse ja standardhälve.

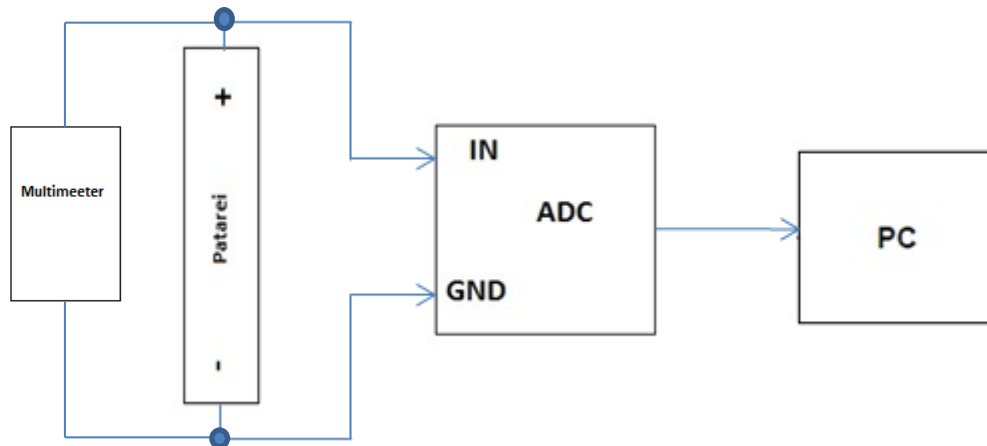
AVERAGE= 4002

STDEV= 1,5

Tulemustest võib näha, et ADM ei saavutanud oma 3,3 V toitepinge teoreetilist digitaalväärtust, mis on 4095. Selle asemel on ta keskvaertus 4002, mis korrutades ühe biti pingeväärtusega (LSB-ga) 0,8 mV annab meile umbes 3,2 V. See võib tuleneda toitepinge ebastabiilsusest või ADM-i kalibreeringu häirest, kuid õppelabori vahenditega ei ole võimalik viga tuvastada.

4.4. ADM-i testimine patareiga

Katse eesmärk oli leida viis, kuidas kontrollida ADM-i stabiilsust konstantse pingesallikaga, mis ei tuleks personaalarvuti DAM-ist ning muuta ADM-i taksagedust ja sampling akna kestvust. Selleks valisin pingesallikaks õppelaboris oleva patarei (vt joonis 4.8). Kontrollisin multimeetriga patarei pingeväärtust ja leidsin, et see oli 1,2 V. Selleks, et patareid kasutada ADM-i katsetamisel, oli vaja selle külge joota sobiva pistikuga juhtmeid. Juhtmed lõikasin nii lühikeseks kui võimalik, et vähendada müra, mida nad võiks tekitada (vt foto 4.9).



Joonis 4.8 Mõõteskeem patareiga testimiseks.

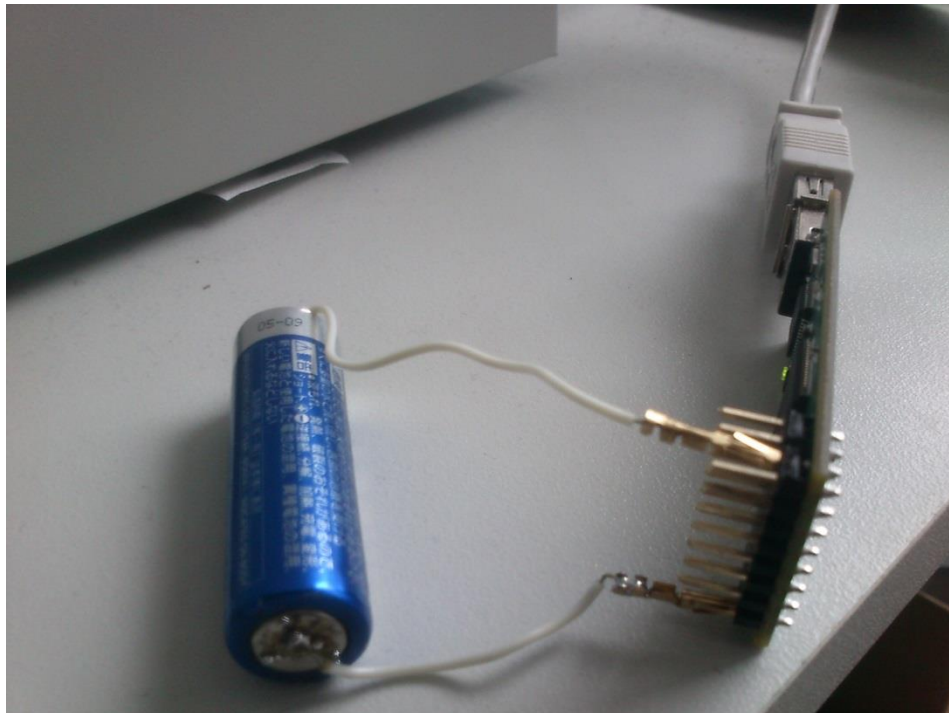


Foto 4.9 ADM-i testimine patareiga

Selles katses huvitas meid kuidas ADM-i taktsagedus ja sample and hold pikkus muudab tulemusi. See nõudis näidisprogrammi „ADC- Simple DAC to ADC“ modifitseerimist. Näidisprogrammi lisati käsuread, mille abil oli võimalik ADM-i taktsagedust muuta. Taktsagedust muudeti CLKDIV2EN ja CLKDIV4EN registri väärtustega. Registritesse kirjutati väärtus 0 või 1 vastavalt soovitud tulemusele. Võimalikke kiiruste kombinatsioone on näha tabelist 4.10.

Tabel 4.10 ADM-i taktsagedust kontrollivate bitide väärtuste tabel [5]

Bit	Field	Value	Description	
2	CLKDIV4EN	0	When enabled, divides the ADC input clock by 4 ADC clock = CPU clock	
		1	ADC clock = CPU clock/4	
0	CLKDIV2EN	0	ADC clock = CPU clock	
		1	ADC clock = CPU clock/2	
		Combined with the CLKDIV4EN bit, this determines the ADC Clock to CPU Clock ratio.		
		CLKDIV2EN	CLKDIV4EN	ADCCLK
		0	0	SYSCLK
	0	1	SYSCLK	
	1	0	SYSCLK / 2	
	1	1	SYSCLK / 4	

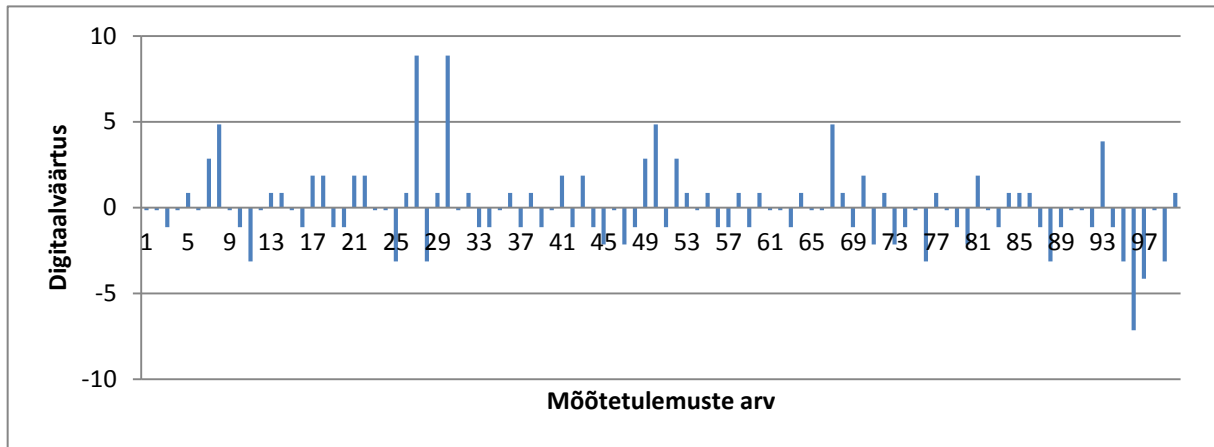
Sample-and-hold (S/H) akna pikkust saab muuta registri bitiga, mille nimi on ACQPS (Acquisition Prescale). See muutuja kirjeldab mitu ADM-i takti tsüklit S/H kestab. Miinimum ACQPS väärtus on 6, mis kokku on 6+1 ADM-i tsüklit ehk 7 takti. Tabelis 4.11 on toodud mõned võimalikud ACQPS väärtused.

Tabel 4.11 ACQPS võimalikud väärtused [5]

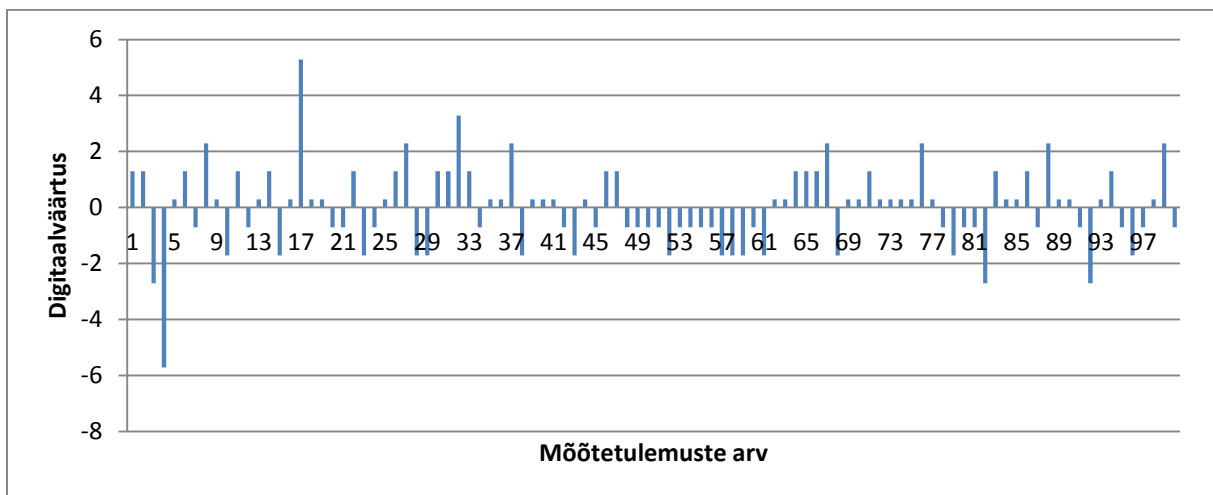
Bit	Field	Value	Description
5-0	ACQPS		SOCx Acquisition Prescale. Controls the sample and hold window for SOCx. Minimum value allowed is 6.
		06h	Sample window is 7 cycles long (6 + 1 clock cycles).
		07h	Sample window is 8 cycles long (7 + 1 clock cycles).
		08h	Sample window is 9 cycles long (8 + 1 clock cycles).
		09h	Sample window is 10 cycles long (9 + 1 clock cycles).
	
		3Fh	Sample window is 64 cycles long (63 + 1 clock cycles).

Programmis käivitati ka DMA moodul, mis võimaldab muunduri tulemused otse mällu kirjutada nii, et protsessor ei tegele andmete üleviimisega. Selline meetod eemaldab võimaliku müra, mis võib tekkida protsessori töötamisest ning tagab kiire andmete edastuse. Loodi DMA buffer, mille suuruseks määrasin 1024, mis tähendab, et muundur teeb 1024 mõõtmist.

Esimese ja teise katse tingimusteks olid ADM-i taktsagedus 45 MHz (SYSCLOCK/2) ja vastavalt S/H aken 7 ja 64 tsükli. Mõõtetulemused on graafikul 4.12 ja 4.13.



Graafik 4.12 ADM-i müra kui ACQPS=6



Graafik 4.13 ADM-i müra kui ACQPS=63

Kui ACQPS=6, siis tulemuste AVERAGE= 1535,15 ehk 1,228 V ja STDEV= 2,25 ehk 1,8 mV

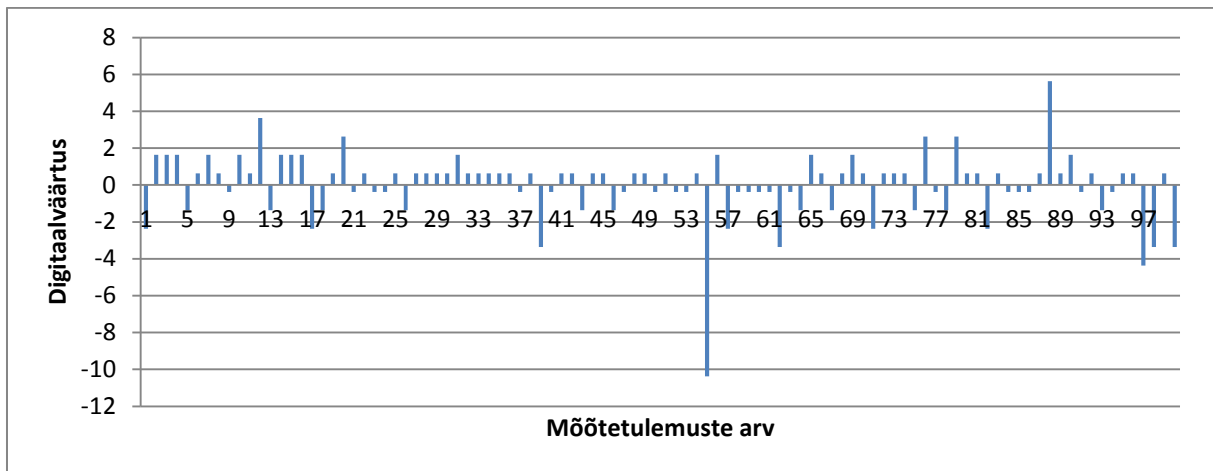
$$\text{SNR} = 20 \log(1228/1,8) = 56,7 \text{ dB}$$

Kui ACQPS=63, siis tulemuste AVERAGE= 1537,71 ehk 1,230 V ja STDEV= 1,49 ehk 1,2 mV

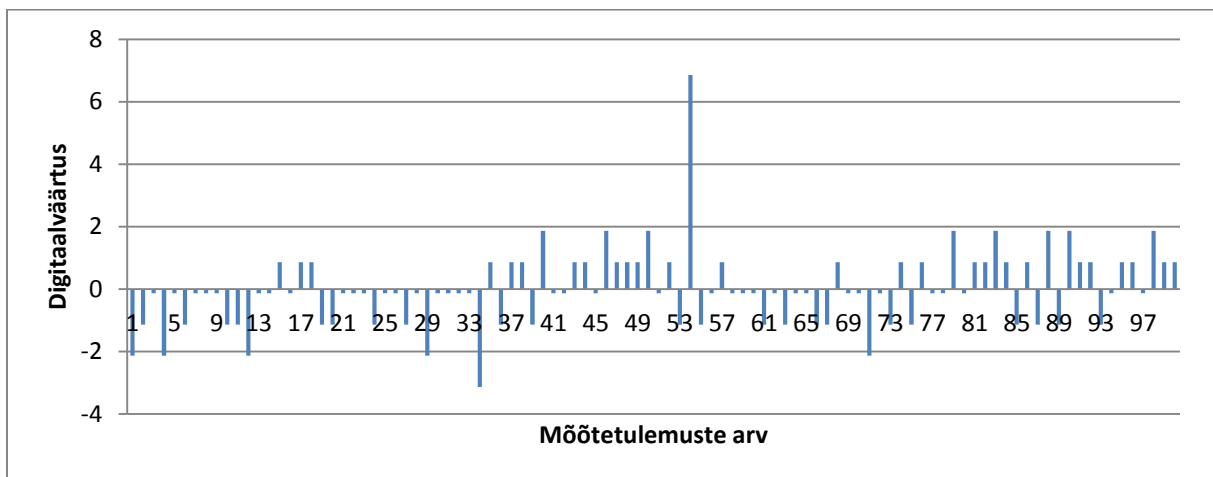
$$\text{SNR} = 20 \log(1230/1,2) = 60 \text{ dB}$$

Saadud andmetest võib eeldada, et mida kauem S/H kestab, seda täpsem on ADM-i väljund ja seda väiksem on müra.

Kolmanda ja neljanda katse tingimusteks on, et ACQPS omab konstantset väärtust, milleks on 6, ning me teeme mõõtmised kui ADM taktsagedus on 90 MHz (SYSCLOCK) ja 22,5 MHz (SYSCLOCK/4). Tulemused on esitatud graafikul 4.14 ja 4.15.



Graafik 4.14 ADM-i müra kui taktsagedus on 90 MHz



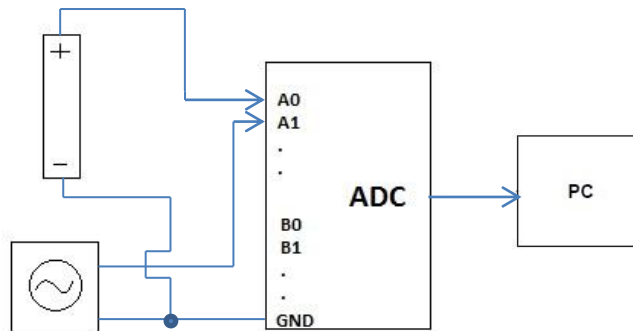
Graafik 4.15 ADM-i müra kui taktsagedus on 22,5 MHz

Kui taktsagedus on 90 MHz, siis AVERAGE= 1530,37 ehk 1,224 V ja STDEV= 1,85

Kui taktsagedus on 22,5 MHz, siis AVERAGE= 1538,14 ehk 1,230 V ja STDEV= 1,25

4.5. ADM-i kanalite ülekostvuse testimine

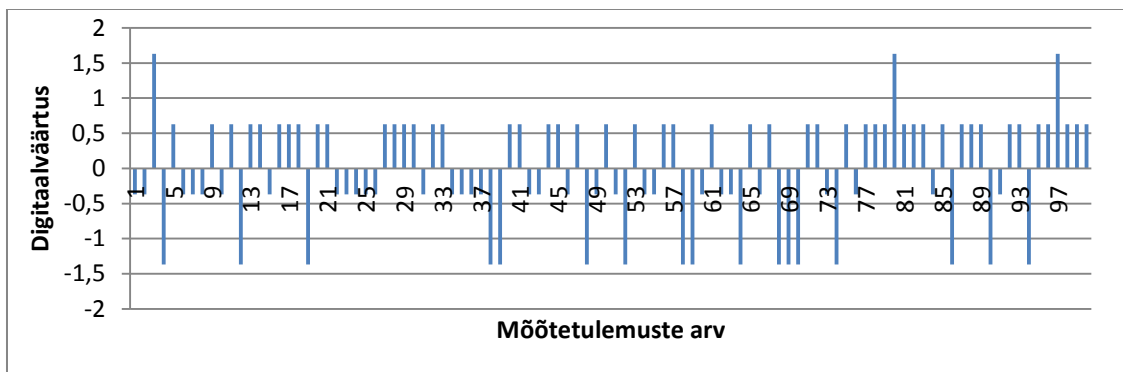
Ülekostvuse testimiseks oli mul vaja ühe ADM-i sisendisse genereerida siinus signaal(vt foto 4.21) ja teise sisendisse alalispingeallikas, milleks ma kasutasin eelnevalt testitud 1,2 V patareid. Katse eesmärk oli uurida, kas välise signaali lisamine muudab ADM-i tulemusi tekitades müra või anomaaliaid. Kõik eelmised katsed olid läbiviidud ühe sisend kanali peal, milleks oli kanal A. Piccolo F28069 omab kahte sisend kanalit, mida on võimalik katsetada. Katse käigus tuli siinus signaali sisestada esialgu samasse multipleksori kanalisse, kuhu patarei väljund on sisestatud, ning hiljem teise kanali multipleksorisse, kuhu pole ühendatud alalispinge allikat.



Joonis 4.16 ADM-i kanalite ülekostvuse mõõteskeem

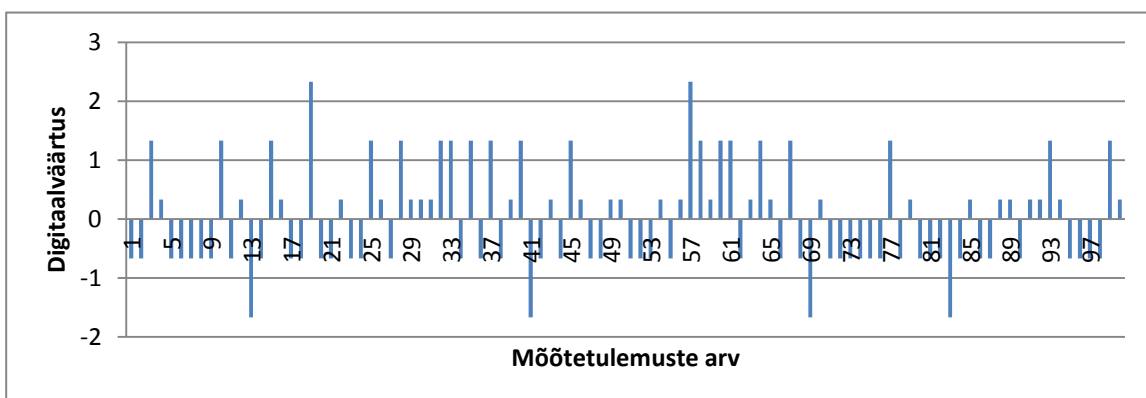
Esiteks genereerisin signaaligeneraatoriga 2 MHz siinus signaali, mille peak-to-peak pingeks määrasin 2V. Kuna ADM ei võimalda negatiivse pinge muundamist, siis tuli generaatoriga lisada alalispinge offset, et tõsta siinus signaali nullist kõrgemale. Nyquisti kriteeriumi täitmiseks muutsin ma ADM-i programmi nii, et selle samplimis sagedus oleks 4MHz. Patarei ühendasin muunduriga nagu eelmises katses.

Generaatori väljundi ühendamisel ADM-iga oli mul tarvis samu juhtmeid koos pistikutega, mida ma kasutasin DAM-ist ADM-i tulemuste katsetamisel. Katse tulemuste saamiseks oli nüüd vaja ühendada generaatori väljund ADM-i multiplexsori sisendisse A1 (patarei on ühendatud sisendisse A0). Lasin programmil teha 1024 mõõtmist ja 100 tulemuse müra tõin välja graafikus 4.17.



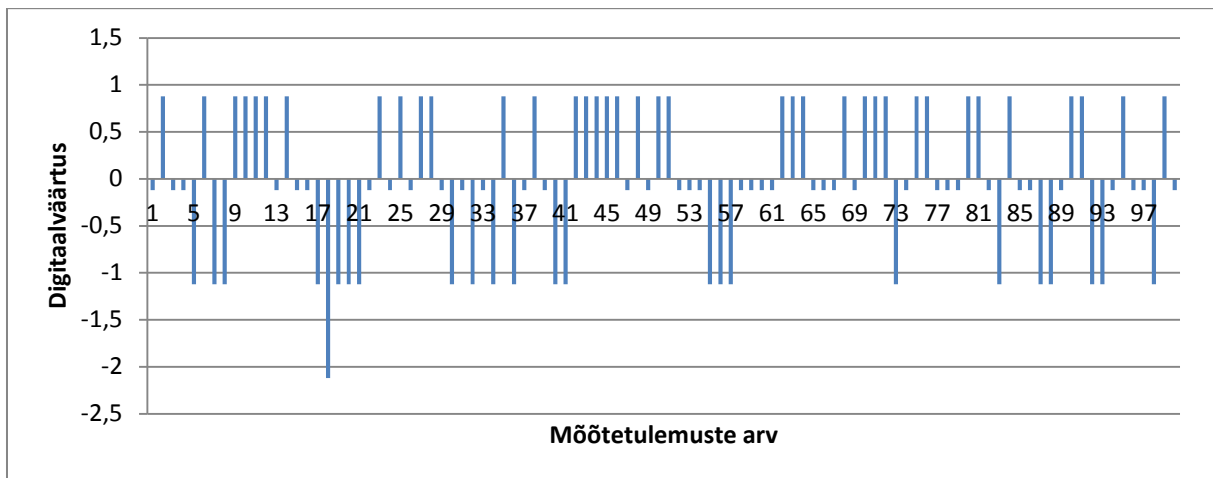
Graafik 4.17 ADM-i müra, kui väline signaal on sama multiplexsori kanalis, sagedus 2MHz

Järgmisena ühendasin signaaligeneraatori väljundi ADM-i teise kanali multiplexsori sisendisse B0. Lasin programmil teha 1024 mõõtmist ja 100 tulemuse müra tõin välja graafikus 4.18.

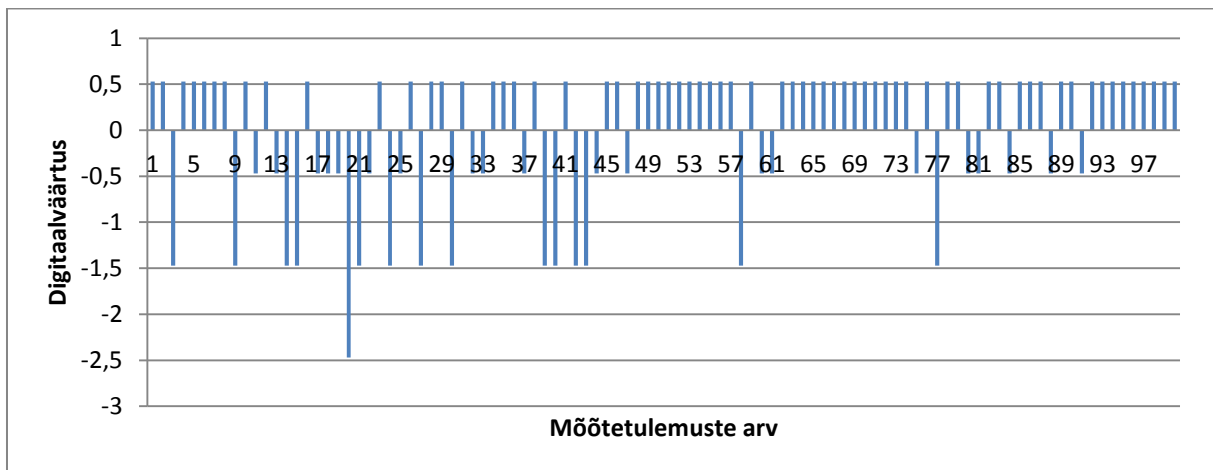


Graafik 4.18 ADM-i müra, kui väline signaal on teise multiplexsori kanalis, sagedus 2 MHz

Kordasin sama katset ka 1 MHz siinuse signaaliga, mille peak-to-peak pingeks seadsin 2,4 V. Tulemused tõin välja graafikus 4.19 ja 4.20.



Graafik 4.19 ADM-i müra, kui väline signaal on sama multipleksori kanalis, sagedus 1MHz



Graafik 4.20 ADM-i müra, kui väline signaal on teise multipleksori kanalis, sagedus 1 MHz

Graafikutelt võib näha, et müra on küllaltki väike, millest võib järeldada kõrvalise signaali vähest mõju ADM-i tulemustele.

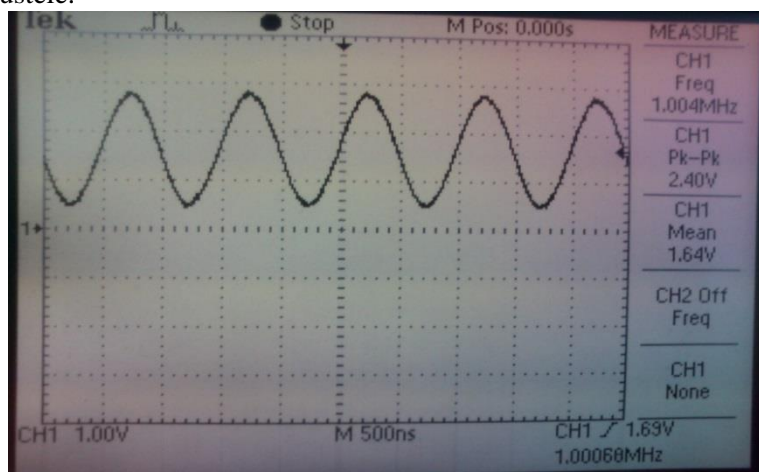


Foto 4.21 Signaaligeneraatori väljund

Kokkuvõte

Bakalaureusetöö eesmärgiks oli hinnata Piccolo F28069 ADM-i omadusi õppelabori keskkonnas. Laboris oli võimalik teha mõned katsed, kuid enamik katseid vajas täpismõõteseadeid, mida paraku õppelaboris kohapeal ei olnud.

Staatiliste parameetrite katseid oli suures osas võimalik teha, kuid tulemustes oli näha silmapaistvat müra, nullinihet ning võimendusteguri viga. Need hoidsid ära oodatud muunduste tulemuste saabumist. Staatilise karakteristiku paremaks uurimiseks oleks vaja olnud väga täpset iseseisvat DAM-i, mis ei oleks ühendatud sama personaalarvutiga, kus ADM oli ühendatud.

Dünaamilisi parameetreid oli keeruline hinnata, kuna õppelaboris signaali analüsaatoreid ei olnud saada. Dünaamiliste parameetrite hindamiseks oleks tarvis ka lisa filtreid ning väga täpseid signaaligeneraatoreid.

Töö käigus tutvusin ka mikrokontrolleri programmeerimisega. Kõige suurem vaev oli näidisprogrammi läbi töötlemine ja käima saamine, kuna Code composer studio programmeerimis keskkonnaga polnud ma enne tutvunud. Selle bakalaureuse töö käigus sain ma palju teada kontrollerite programmeerimisest. Siin kohal võiks natuke kritiseerida Texas Instrumentsi F28069 dokumentatsiooni, mis tihti ei olnud kõige selgemini arusaadav. Mõnele terminile ja käsule polnud kindlat seletust ja nende käskude otsing teistest allikatest võttis aega.

Töö tulemuseks ütleks, et Piccolo F28069 ADM saaks praktikumides kasutada üldise ADM-i töö illustreerimiseks. Tema tehnilised näitajad on minu hinnangul liiga head, et õppelabori keskkonnas süvitsi uurida. Õppimise eesmärgil oleks vaja lihtsama arhitektuuriga, aeglasemat või vähema bittidega ADM-i, mis võimaldaks täpsemate omaduste hinnangu.

Kasutatud kirjandus

1. Walt Kester .(2004). Analog-Digital Conversion. 9200 lk
2. Vodovozov, V. , Jansikene, R. (2008). Elektroonika ja jõupooljuhttehnika. 139 lk
3. Nihal Kularatna. (2003). Digital and Analogue Instrumentation testing and measurement. 645 lk
4. F28069 datasheet tms320f28069.pdf 177 lk
5. F28069 technical reference manual spruh18e.pdf 1192 lk
6. Analog-to-digital converter [www] http://en.wikipedia.org/wiki/Analog-to-digital_converter (05.06.2015)
7. Sample-and-Hold Amplifiers [www] <http://www.analog.com/media/cn/training-seminars/tutorials/MT-090.pdf> (05.06.2015)
8. Understanding Pipelined ADC [www] <http://www.maximintegrated.com/en/app-notes/index.mvp/id/1023> (05.06.2015)
9. ADC Architectures III: Sigma-Delta ADC Basics [www] <http://www.analog.com/media/en/training-seminars/tutorials/MT-022.pdf> (05.06.2015)
10. Lundberg,K. Analog-to-Digital Converter Testing. (2002). A2Dtesting.pdf 20 lk

Mikrokontrolleri programmeerimise kood

```
#include "DSP28x_Project.h"

void DeviceInit(void);
void InitFlash();
void MemCopy(Uint16 *SourceAddr, Uint16* SourceEndAddr, Uint16* DestAddr);
extern Uint16 RamfuncsLoadStart, RamfuncsLoadEnd, RamfuncsRunStart;
#define BUF_SIZE 1024
#pragma DATA_SECTION(ADCA0, "DMARAML5");
Uint16 adc_result;
Uint16 duty_cycle = 20;
Uint16 ADCA0[BUF_SIZE];
Uint16 Voltage[1000];
Uint16 i;
// MAIN CODE - starts here
void main(void)
{
    Uint16 j = 0;
    for (i=0; i<BUF_SIZE; i++)
    { ADCA0[i] = 0; }

    DeviceInit(); // Device Life support & GPIO mux settings
#ifdef FLASH
MemCopy(&RamfuncsLoadStart, &RamfuncsLoadEnd, &RamfuncsRunStart);
InitFlash(); // Call the flash wrapper init function
#endif //(FLASH)

    EALLOW;

    SysCtrlRegs.PCLKCR3.bit.DMAENCLK = 1;

    EDIS;

    EALLOW;

    PieVectTable.DINTCH2= &DINTCH2_ISR;
```

```

EDIS;

IER = M_INT7 ;                               //Enable INT7 (7.1 DMA Ch1)

PieCtrlRegs.PIECTRL.bit.ENPIE = 1;         // Enable the PIE

PieCtrlRegs.PIEACK.all = 0xFFFF;

EINT; // Enable Interrupts at the CPU level

EALLOW;

SysCtrlRegs.PCLKCR0.bit.TBCLKSYNC = 1;

GpioCtrlRegs.GPAPUD.bit.GPIO0 = 1; // Disable pull-up on GPIO0 (EPWM1A)

GpioCtrlRegs.GPAPUD.bit.GPIO1 = 1; // Disable pull-up on GPIO1 (EPWM1B)

/* Configure EPWM-1 pins using GPIO regs*/

GpioCtrlRegs.GPAMUX1.bit.GPIO0 = 1; // Configure GPIO0 as EPWM1A

GpioCtrlRegs.GPAMUX1.bit.GPIO1 = 1; // Configure GPIO1 as EPWM1B

EDIS;

#define period 40                             // 2MHz when PLL is set to 0x10 (80MHz) (80MHz / 40 = 2MHz )

        duty_cycle = 20;                       // Initialize to 50% duty cycle

// Time-base registers

EPwm1Regs.TBPRD = period;                     // Set timer period, PWM frequency = 1 / period

EPwm1Regs.TBPHS.all = 0;                      // Time-Base Phase Register

EPwm1Regs.TBCTR = 0;                          // Time-Base Counter Register

EPwm1Regs.TBCTL.bit.PRDLN = TB_IMMEDIATE; // Set Immediate load

EPwm1Regs.TBCTL.bit.CTRMODE = TB_COUNT_UP; // Count-up mode: used for asymmetric PWM

EPwm1Regs.TBCTL.bit.PHSEN = TB_DISABLE;      // Disable phase loading

EPwm1Regs.TBCTL.bit.SYNCSEL = TB_SYNC_DISABLE;

EPwm1Regs.TBCTL.bit.HSPCLKDIV = TB_DIV1;

EPwm1Regs.TBCTL.bit.CLKDIV = TB_DIV1;

// Setup shadow register load on ZERO

EPwm1Regs.CMPCTL.bit.SHDWAMODE = CC_SHADOW;

EPwm1Regs.CMPCTL.bit.LOADAMODE = CC_CTR_ZERO; // Load on CTR=Zero

// Set compare values

EPwm1Regs.CMPA.half.CMPA = duty_cycle;       // Set duty 50% initially

```

```

// Set actions

EPwm1Regs.AQCTLA.bit.ZRO = AQ_SET;      // Set PWM1A on Zero event

EPwm1Regs.AQCTLA.bit.CAU = AQ_CLEAR;    // Clear PWM1A on Up-Count CompareA event

// ADC INITIALISATION

EALLOW;

AdcRegs.ADCCTL2.bit.CLKDIV2EN = 1;

AdcRegs.ADCCTL2.bit.CLKDIV4EN = 0;

    EDIS;

EALLOW;

    AdcRegs.ADCCTL1.bit.ADCREFSEL = 0;      // use internal band gap reference

AdcRegs.ADCCTL1.bit.ADCBGPWD= 1;        // power up band gap

AdcRegs.ADCCTL1.bit.ADCREFPWD= 1;       // power up reference

    AdcRegs.ADCCTL1.bit.ADCPWDN = 1;      // power up rest of ADC

    AdcRegs.ADCCTL1.bit.ADCENABLE = 1;    // enable ADC output

    for(j=0; j<5000; j++){                // wait 60000 cycles = 1ms (each iteration is 12
cycles)

AdcRegs.ADCCTL2.bit.ADCNONOVERLAP = 1; // Enable non-overlap mode

    AdcRegs.ADCCTL1.bit.INTPULSEPOS = 1;   // create int pulses 1 cycle prior to output latch

//EOC = end of conversion event, SOC = start of conversion event

AdcRegs.INTSEL1N2.bit.INT1SEL = 0;        // ADCCH0 (ADC-A0) EOC causes ADCInterrupt1

AdcRegs.INTSEL1N2.bit.INT1CONT = 1;       // set ADCInterrupt 1 to auto clr (continuous conversion)

AdcRegs.INTSEL1N2.bit.INT1E = 1;         // enable ADCInterrupt1; 0=none, 1=ADCInt1, 2=ADCInt2

AdcRegs.ADCSAMPLEMODE.bit.SIMULEN0 = 0; // none-Simultaneous sample for SOC0 and SOC1

AdcRegs.ADCINTSOCSEL1.bit.SOC0 = 1;

    AdcRegs.ADCSOC0CTL.bit.CHSEL = 0;     // set SOC0 channel select to ADCINA0

AdcRegs.ADCSOC0CTL.bit.TRIGSEL = 7;      // ePWM2, ADCSOCA

    AdcRegs.ADCSOC0CTL.bit.ACQPS = 10;    // set SOC0 S/H Window to 7 ADC Clock Cycles, (6 ACQPS plus 1) ADC
clk 60MHz :: Total Time to Process Analog Voltage 333.34ns eg. 3MSPS

AdcRegs.ADCSOCFRC1.all = 0x0001;        // kick start ADC by causing an ADCInterrupt1 event

```

```

EDIS;

DMAInitialize();

EALLOW;

DmaRegs.CH2.SRC_BEG_ADDR_SHADOW = (Uint32)&AdcResult.ADCRESULT0; // Point to beginning of source
buffer

DmaRegs.CH2.SRC_ADDR_SHADOW = (Uint32)&AdcResult.ADCRESULT0;

// Set up DESTINATION address:

DmaRegs.CH2.DST_BEG_ADDR_SHADOW = (Uint32)&ADCA0[0]; // Point to beginning of destination buffer

DmaRegs.CH2.DST_ADDR_SHADOW = (Uint32)&ADCA0[0];

EDIS;

DMACH2BurstConfig(0,0,0); //BURST_SIZE ; SRC_BURST_STEP ; DST_BURST_STEP
(1,BUFFER_SIZE,4);

DMACH2TransferConfig((BUF_SIZE-1),0,1); // TRANSFER_SIZE ; SRC_TRANSFER_STEP ;
DST_TRANSFER_STEP ((PER_CNT-1),0,0);

DMACH2WrapConfig(0,0,0,1); // SRC_WRAP_SIZE ; SRC_WRAP_STEP ;
DST_WRAP_SIZE ; DST_WRAP_STEP (0,1,0,0);

DMACH2ModeConfig(1,PERINT_ENABLE,ONESHOT_DISABLE,CONT_DISABLE,SYNC_DISABLE,SYNC_SRC,OV
RFLOW_DISABLE,SIXTEEN_BIT,CHINT_END,CHINT_ENABLE);

StartDMACH2();

// Forever LOOP

//for(;;) //infinite loop

for(j=0; j<5000; j++){

}

interrupt void DINTCH2_ISR(void) // DMA Channel 2

{

PieCtrlRegs.PIEACK.bit.ACK7 = 1;

EALLOW;

DmaRegs.CH2.CONTROL.bit.RUN = 1;

EDIS;

}

```