

MCD tiedostojen lukeminen ja konvertointi

19.12.2017, Lauri Kantola (Jyväskylän yliopisto), lauri.t.kantola@jyu.fi

Tämä ohje koskee seuraavia MATLAB -skriptejä, jotka on tehty helpottamaan MCD -tiedostojen lukua MATLAB:iin ja niiden konvertoimista binääritiedostoiksi jatkokäsittely varten (esim. KlustaKwik).

validateMCDMAT2BIN.m = Validoi loadMCDtoStruct skriptillä ladatun MCD -tiedoston annetun binääritiedoston sisältöön.

validateMCDBIN.m = Validoi kaksi annettua binääritiedostoa.

saveBinaryMCS.m = Apufunktio binääritiedostojen tallentamiseen.

reorderChannels.m = Apufunktio kanavien järjestämiseen rautapohjaisen ID -tunnuksen perusteella. Eri MCS -järjestelmät tallentavat kanavat eri järjestyksessä MCD -tiedostoon. Kanavien järjestäminen ajetaan aina oletuksena.

remapChannels.m = Apufunktio kanavien järjestämiseen 'remap' -tiedoston avulla.

loadStreamData.m = Apufunktio MCD -tiedostojen datavirtojen lukemiseen.

loadRecordingInfo.m = Apufunktio mittaukseen ja dataa liittyvien tietojen lukemiseen.

loadMCDtoStruct.m = Lataa MCD -tiedoston MATLAB:iin. Varsinainen ohjelma.

convertMCDtoDat.m = Muuntaa MCD -tiedoston binääritiedostoksi. Arvot tallennetaan 16 bit kokonaislukuina.

loadInChunks.m = Apufunktio datan lataamiseen palasissa. Nopeuttaa lataamista ja korjaa ongelmat pitkien mittauksen kanssa. Mikäli mittaus on lyhyempi, kuin 10 000 ms, pitää tätä muokata.

extractRippleChAndAnalogCh.m = Tallentaa .loadch -tiedoston perusteella valitut ripplekanavat ja kaikki analogikanavat .mat tiedostoon. Käyttää samoja argumentteja, kuin 'loadMCDtoStruct'. Muista käyttää remap tiedostoa ja argumenttia kanavien järjestämiseen.

Sisällysluettelo

Esivalmistelut	2
MCD -tiedoston lukeminen MATLAB:iin	2
MCD -tiedoston konvertointi binääritiedostoksi	5
MC_DataTool:illa	5
MATLAB:issa	6
Datan validointi	6

Esivalmistelut

Asenna seuraavat Multi Channel Systemsin ohjelmat:

- MC_Rack (<http://www.multichannelsystems.com/software/mc-rack>). MC_Rack sisältää MCD -tiedostojen lukemiseen tarvittavia työkaluja.
- MC_Rackin tarvittavat työkalut tulee lisätä MATLAB:in polkuun. Sen voi tehdä antamalla komento 'addpath' komentoikkunaan. Oikea 'mcintfac' -kansio löytyy yleensä 'Program Files' -kansioista (esim. C:\Program Files (x86)\Multi Channel Systems\MC_Rack\MCStreamSupport\Matlab\meatools\mcintfac).

Esimerkki komento:

```
addpath 'C:\Program Files (x86)\Multi Channel Systems\MC_Rack\MCStreamSupport\Matlab\meatools\mcintfac'
```

Huom!

Lisäys ei säily seuraavaan MATLAB:in käynnistykseen, ellei sitä syötetä järjestelmänvalvojan oikeuksilla.

- Lisää MATLAB:in polkuun myös tätä ohjetta koskevien MATLAB -skriptien kansio samalla tavoin, kuin lisäsit MC_Rack -kansion. Tämän jälkeen käsiteltävät tiedostojen ja skriptien ei tarvitse olla samassa työkansiossa.
- MC_DataTool (<http://www.multichannelsystems.com/software/mc-datatool>). MC_DataToolin avulla voi muuntaa MCD -tiedostoja binäärimuodossa. MC_DataTool:illa konvertoituja binääritiedostoja käytetään MATLAB:illa luettujen tiedostojen validointiin.

MCD -tiedoston lukeminen MATLAB:iin

loadMCDtoStruct -skripti lukee MCD -tiedoston MATLAB:iin 'struct' -tyyppiseen muuttujaan.

MCD -tiedosto ladataan esim. seuraavasti:

```
data = loadMCDtoStruct('testData.mcd');
```

Esimerkissä 'data' on muuttuja, johon tiedoston sisältö ladataan. 'testData.mcd' on ensimmäinen argumentti, jonka tulee olla ladattavan tiedoston nimi.

Lisäksi voidaan syöttää erilaisia optioita esim. seuraavasti:

```
data = loadMCDtoStruct('testData.mcd','selectStream','Electrode Raw Data','startPoint',100);
```

Esimerkissä tiedostonimi on ensimmäinen argumentti. Seuraavat argumentit voivat olla missä järjestyksessä tahansa, kunhan ensin tulee argumentin nimi ja sitten argumentin arvo. Esimerkissä 'selectStream' on argumentin nimi ja 'Electrode Raw Data' on argumentin arvo. Ko. argumentti lataa vain 'Electrode Raw Data' datavirran ja muut datavirrat jätetään pois.

Lista käytettävistä argumenteista:

- **'loadStream'** = Määrittää ladataanko datavirtoja vai pelkästään mittaukseen ja tiedostoon liittyviä tietoja. Oletuksena 'true'. Kytetään pois syöttämällä 'false'.
- **'startPoint'** = Määrittää ladattavan datan alkupisteen millisekunteina. Oletuksena on 0, eli mittauksen alku.
- **'endPoint'** = Määrittää ladattavan datan loppupisteen millisekunteina. Oletuksena on mittauksen loppu. Mittauksen pituuden voi selvittää lataamalla ensin mittauksen tiedot syöttämällä 'loadStream' kohtaan 'false'. Loppupiste löytyy kentästä 'endMs' (esim. 'data.endMs').
- **'selectStream'** = Tämän avulla voidaan valita vain haluttu datavirta (esim. 'Electrode Raw Data' tai 'Avg Trigger' jne.). Huom! Älä syötä datavirran numeroa.
- **'orderByHwID'** = Määrittää järjestetäänkö kanavat rautapohjaisten id-numeroiden perusteella. Eri järjestelmät tallentavat kanavat eri järjestyksessä. Tämän avulla kanavat ovat aina järjestyksessä. Kanavien oikea numero ja nimi löytyvät kentistä 'correctedChIDs' ja 'correctedChNames'. Tämä on oletuksena päällä, mutta voidaan disabloida syöttämällä 'false'.
- **'toMuVolt'** = Muuntaa arvot mikrovoltteiksi. Oletuksena 'false'. Muuntaa arvot myös double-muuttujaksi, jolloin pitkien mittausten kanssa voi loppua muisti kesken. Lisäksi, jos lataaja ilmoittaa käyttävänsä oletusarvoja, ei muunnos välttämättä pidä paikkaansa.
- **'remap'** = Järjestää kanavat '.remap' -tiedoston mukaiseen järjestykseen. Remap-tiedoston tulee olla samanniminen, kuin ladattava tiedosto. Kanavat tulee olla eroteltuna ';' tiedostossa.

Esim. remap -tiedostosta:

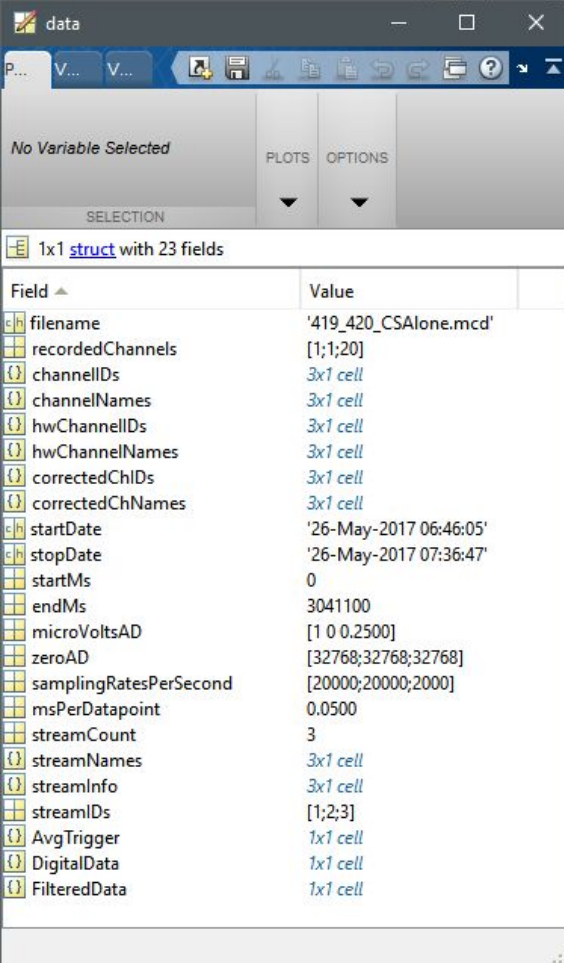
```
9;7;11;5;13;3;15;1;16;8;14;6;10;4;12;2;26;18;28;20;30;24;32;22;25;23;27;21;29;19;31;17
```

Muuttujaan ladatut tiedot:

- **'filename'** = Ladatun MCD -tiedoston nimi.

- **'recordedChannels'** = Kunkin datavirran kanavien määrä listana datavirran id-numeron mukaisessa järjestyksessä.
- **'channelIDs'** = Kunkin datavirran kanavien id-numerot listana datavirran id-numeron mukaisessa järjestyksessä. KANAVIDEN JÄRJESTYKSET EIVÄT PIDÄ PAIKKAANSA, AINOASTAAN VARALTA MUKANA.
- **'channelNames'** = Kunkin datavirran kanavien nimet listana datavirran id-numeron mukaisessa järjestyksessä. KANAVIDEN JÄRJESTYKSET EIVÄT PIDÄ PAIKKAANSA, AINOASTAAN VARALTA MUKANA.
- **'hwChannelIDs'** ja **'hwChannelNames'** = Kanavien rautapohjaiset id-numerot ja nimet listana kunkin datavirran id-numeron mukaisessa järjestyksessä. JÄRJESTYKSESSÄ AINOASTAAN SILLOIN, KUN **'orderByHwID'** on **'false'**.
- **'correctedChIDs'** ja **'correctedChNames'** = Kanavien id-numerot ja nimet listana kunkin datavirran id-numeron mukaisessa järjestyksessä. Tämä kanavajärjestys pitää yleensä paikkansa, kun **'orderByHwID'** ei ole **'false'**. Muista validoita data kun lataat eri kokeiden mittauksia! Katso ohjeen kohta **'Datan validointi'**.
- **'startDate'** = Mittauksen aloitus aika ja päivämäärä.
- **'stopDate'** = Mittauksen lopetus aika ja päivämäärä.
- **'startMs'** = Mittauksen aloituspiste.
- **'endMs'** = Mittauksen lopetuspiste ja myös mittauksen pituus, jos ei manuaalisesti määritely.
- **'microVoltsAD'** = Kunkin datavirran kerroin datavirran id-numeron mukaisessa järjestyksessä. Kertoimen avulla voi muuntaa arvot Mikrovolteiksi (μV).
- **'zeroAD'** = Kunkin datavirran nollakohta datavirran id-numeron mukaisessa järjestyksessä. Nollakohta on laskettu automaattisesti ladatusta datasta.
- **'samplingRatesPerSecond'** = Kunkin datavirran näytteenottotaajuus per sekunti datavirran id-numeron mukaisessa järjestyksessä.
- **'msPerDatapoint'** = Kahden datapisteen välinen ajallinen etäisyys millisekuntein (ms).
- **'streamCount'** = Datavirtojen lukumäärä.
- **'streamNames'** = Kunkin datavirran nimi datavirran id-numeron mukaisessa järjestyksessä.
- **'streamInfo'** = Datavirtoihin liittyviä muita tietoja.
- **'streamIDs'** = Datavirtojen id-numerot.
- **'AvgTrigger', 'DigitalData', 'FilteredData', 'ElectrodeRawData', 'AnalogRawData'** jne. sisältävät ladatut datat. Kenttien sisältö riippuu datavirtojen tyypistä. Esim. **'AvgTrigger'** sisältää merkkien aikapisteen useammassa muodossa, kun taas **'ElectrodeRawData'** sisältää aloitus ja lopetuspisteen, sekä kunkin kanavan datan valitulta aikaväliltä.

Esimerkki ladatusta tiedostosta.



Field	Value
filename	'419_420_CSAlone.mcd'
recordedChannels	[1;1;20]
channelIDs	3x1 cell
channelNames	3x1 cell
hwChannelIDs	3x1 cell
hwChannelNames	3x1 cell
correctedChIDs	3x1 cell
correctedChNames	3x1 cell
startDate	'26-May-2017 06:46:05'
stopDate	'26-May-2017 07:36:47'
startMs	0
endMs	3041100
microVoltsAD	[1 0 0.2500]
zeroAD	[32768;32768;32768]
samplingRatesPerSecond	[20000;20000;2000]
msPerDatapoint	0.0500
streamCount	3
streamNames	3x1 cell
streamInfo	3x1 cell
streamIDs	[1;2;3]
AvgTrigger	1x1 cell
DigitalData	1x1 cell
FilteredData	1x1 cell

MCD -tiedoston konvertointi binääritiedostoksi

Binääritiedostoja tarvitaan viemään dataa muihin ohjelmiin analysoitavaksi. Lisäksi binääritiedostolla voidaan validoida MATLAB:iin vietyjä ja sillä luotuja tiedostoja.

MC_DataTool:illa

Käynnistä MC_DataTool. Avaa MCD -tiedosto (Ctrl + O). Valitse ylävalikosta 'Tools' ja sieltä 'Convert MCD to BIN'. Valitse avautuneesta ikkunasta haluttu datavirta ja halutut kanavat. Vaihda tallennuskansio kohdasta 'Browse path'. Varmista, että 'Write header' ei ole valittu ja valitse 'Signed 16bit'. Lopuksi paina 'Save'.

MATLAB:issa

`convertMCDtoDat` -skripti on tarkoitettu MCD -tiedoston muuntamiseen 16 bittiseksi binääritiedostoksi esim. KlustaKwikiä varten. Toiminta on hyvin samanlainen, kuin `loadMCDtoStruct` -skriptissä. Argumentteja on käytettävissä vain 'selectStream' ja 'orderByHwID', 'startPoint', 'endPoint' ja 'remap'. Skripti luo '.dat' päätteisen tiedosto samaan kansioon, kuin alkuperäinen MCD -tiedosto on.

Datan validointi

Voi verrata MC_DataTool:illa ja MATLAB:illa luotua binääritiedostoa keskenään `validateMCDBIN` -skriptillä. Skripti ilmoittaa ovatko ladatut tiedostot identtisiä. Skripti ottaa argumentteina kaksi tiedostonimeä.

Esimerkkikomento: `'validateMCDBIN('file1.raw','file2.bin');'`

Kahta tiedostoa voi myös verrata keskenään Windowsin komentokehotteessa 'FC' -komennolla. FC:llä voi vertailla mitä tiedostoja tahansa.

Esimerkkikomento: `'FC file1.bin file2.bin'`

`validateMCDMAT2BIN` -skriptillä voi verrata `loadMCDtoStruct` -skriptillä luettua MCD -tiedostoa ja binääritiedostoa. Skripti ilmoittaa ovatko ladatut tiedostot identtisiä. Skripti ottaa argumentteina kaksi tiedostonimeä ja MCD -tiedoston osalta halutun datavirran (oletuksena on 'Electrode Raw Data' datavirta). Ensin tulee syöttää MCD -tiedoston nimi ja sen haluttu datavirta. Lopuksi binääritiedoston nimi.

Esimerkkikomento: `'validateMCDMAT2BIN('file1.mcd','Analog Raw Data','file2.bin');'`