

BILAG 6

Implementering & Test

Indholdsfortegnelse

Kapitel 1 Indledning	3
Kapitel 2 Implementering	4
2.1 Hardware	4
2.1.1 Instrumentationsforstærker 1	4
2.1.2 Strømgenerator	4
2.1.3 Instrumentationsforstærker 2	5
2.1.4 OP-AMP	6
2.1.5 Spektrumanalyse	6
2.1.6 AA filter	7
2.2 Software	8
2.2.1 Funktioner	9
2.2.1.1 Synkerefleksmonitor_OpeningFcn	9
2.2.1.2 Btn_Start_Measurements	9
2.2.1.3 Btn_Save_Measuerments	9
2.2.1.4 Start_GUI	9
2.2.1.5 Generate_SineWave	10
2.2.1.6 Read_Measurements	10
2.2.1.7 Process_Measurements	10
2.2.1.8 Show_Measurements	11
2.2.1.9 Save_Measurments	12
Kapitel 3 Modultest	13
3.1 Hardware	13
3.1.1 Instrumentationsforstærker 1	13
3.1.2 Strømgenerator	14
3.1.3 Instrumentationsforstærker 2	15
3.1.4 OP-AMP	16
3.1.5 Spektrumanalyse	17
3.1.6 AA filter	18
3.2 Software	19
3.2.1 Funktioner	19
3.2.1.1 Synkerefleksmonitor_OpeningFcn	19
3.2.1.2 Btn_Start_Measurement	19
3.2.1.3 Btn_Save_Measurement	19
3.2.1.4 Start_GUI	19
3.2.1.5 Start_GUI	20
3.2.1.6 Generate_SineWave	20
3.2.1.7 Read_Measurements	22

Indholdsfortegnelse

3.2.1.8	Process_Measurements	26
3.2.1.9	Show_Measurements	29
3.2.1.10	Save_Measurments	29
Kapitel 4	Integrationstest	30
4.1	Indledning	30
4.1.1	Hardware Top-down test	30
Kapitel 5	Accepttest	40
5.1	Indledning	40
5.2	Accepttest af Usecases	40
5.2.1	Use Case 1	40
5.2.2	Use Case 2	41
5.2.3	Use Case 3	41
5.3	Accepttest af ikke-funktionelle krav	43
5.4	Problemrapport	46
Litteratur		47

Indledning 1

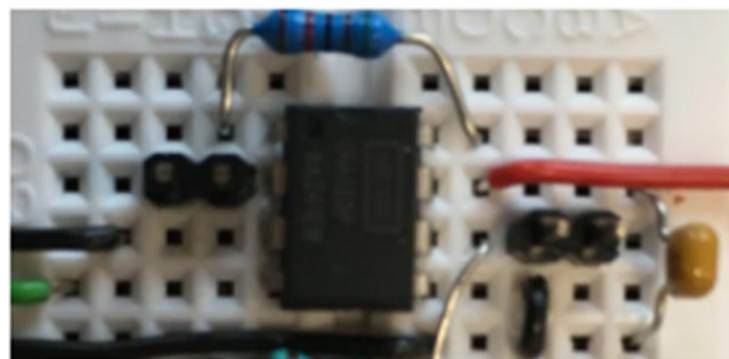
I dette bilag beskrives, hvordan de designede hardware og software-komponenter er implementeret og modultestet. På baggrund af disse designede enheder i designfasen foretages der nu implementering og modultest. Til implementering af hardware komponenter er der valgt at opbygge testkredsløbet på et fumlebræt. Hver af disse komponenter skal igennem en modultest for at verificere om de kan bruges til det tiltænkte formål. For softwarens vedkommende dokumenteres, hvordan kodeimplementeringen er foregået, samt resultatet af disse implementeringer. Formålet med modultesten er at forberede produktet til integrationstest.

Implementering 2

2.1 Hardware

2.1.1 Instrumentationsforstærker 1

Denne instrumentationsforstærker er tiltænkt til at forstærke et AC signal på 2V til 4V. Til dette formål er der anvendt instrumentationsforstærkeren INA128 med de begrundelser, som er angivet i designafsnittet. Figur 2.1 viser INA128 og en eksterne modstand, som bruges til at fastsætte den ønskede forstærkning.

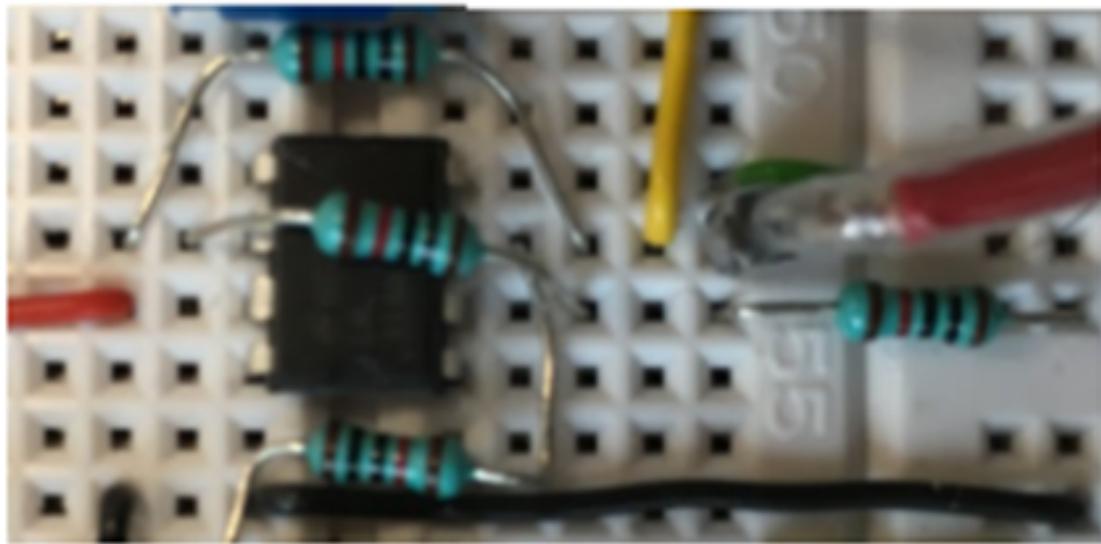


Figur 2.1: Figuren viser, hvordan INA128 er implementeret på et fumlebræt

2.1.2 Strømgenerator

Strømgeneratorens funktion er at levele en konstant strøm, som sendes til et måleobjektets væv. Til implementering af denne strømgenerator er der anvendt operationsforstærkeren LM318. Figur 2.2 viser komponenten LM318 med de tilhørende modstande.

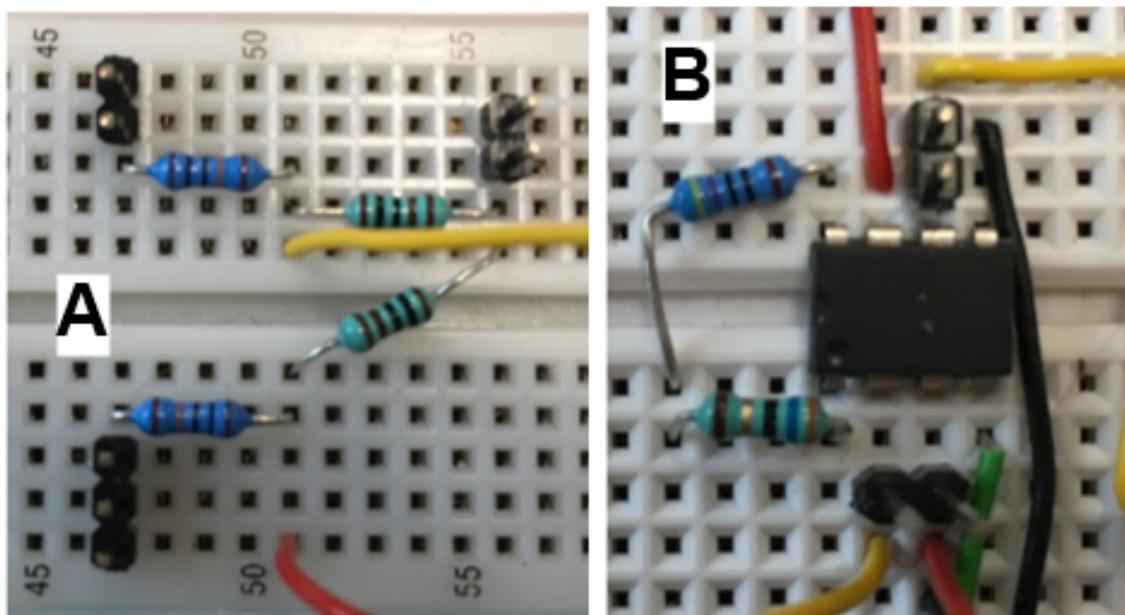
2.1. Hardware



Figur 2.2: Figuren viser, hvordan LM318 er implementeret på et fumlebræt

2.1.3 Instrumentationsforstærker 2

Instrumentationsforstærker 2 bliver brugt til at forstærke biosignal fra måleobjektets og undertrykkelse støj . Til implementering af denne Instrumentationsforstærker er der igen brugt INA128. Figur 2.3 B viser komponenten INA128 med dens tilhørende eksterne modstande. Figur 2.3 A viser en spændingsdeler kredsløb, der er benyttet til at teste instrumentationsforstærkeren. Hvorfor der anvendes en spændingsdeler til test af INA128 henvises der til "bilag 6 - Design".

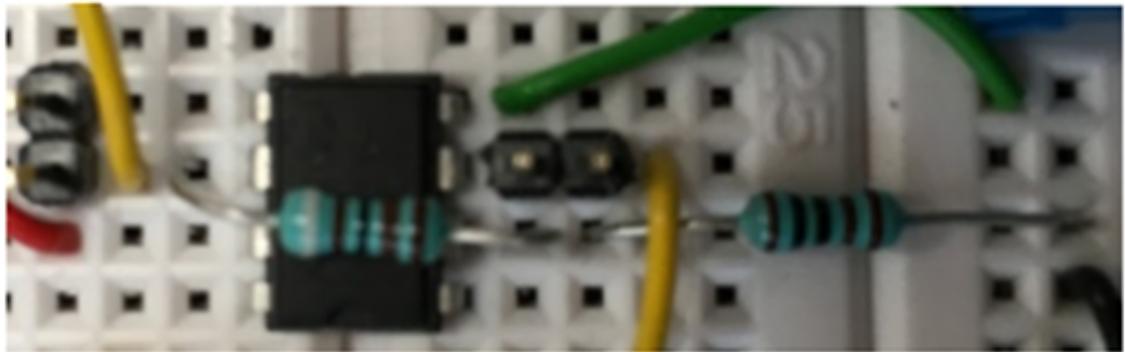


Figur 2.3: Figuren viser, hvordan LM318 er implementeret på et fumlebræt

2.1. Hardware

2.1.4 OP-AMP

Biosignalet fra måleobjektet forstærkes op i to trin. Det første trin benyttes Instrumentationsforstærker 2 og det andet trin anvendes operationsforstærkeren LM318, som forstærker signalet fra Instrumentationsforstærker 2 yderligere. Figur 2.4 viser implementering af LM318 og de to modstande, som fastsætter, hvor meget forstærkning man kan få ud af operationsforstærkeren.

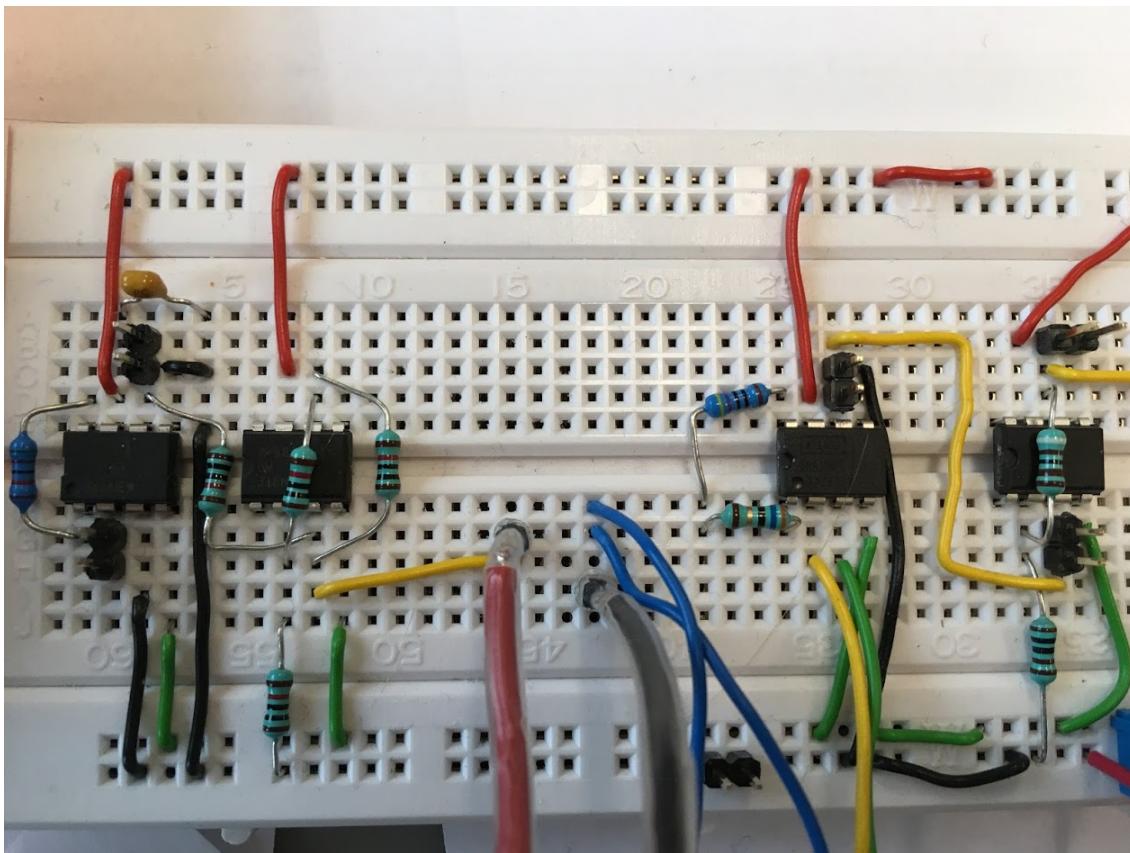


Figur 2.4: Figuren viser, hvordan LM318 er implementeret på et fumlebræt

2.1.5 Spektrumanalyse

Der er behov for at foretage en spektrumanalyse, med alle komponenter sat sammen, for at vurdere hvilket krav der skal være til AA filterets dæmpning.

2.1. Hardware

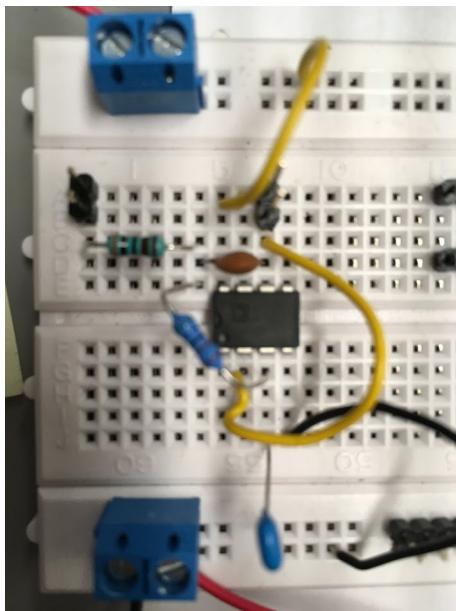


Figur 2.5: Billede over den samlede opstilling med komponenterne: Instrumentationsforstærker 1 & 2, strømgenerator, OP-AP og monteret elektroder. Disse komponenter repræsentere det samlet frekvensspektrum.

2.1.6 AA filter

For at undgå aliasering i det optagte signal implementeres et anti-aliaseringsfilter foran Analog Discovery. AA filteret modtager et forstærketsignal fra OPAMP, filtrere det og sender det videre til Analog Discovery.

2.2. Software



Figur 2.6: Implementering af AA filter.

2.2 Software

I dette afsnit beskrives de vigtigste funktioner for systemet software-del. Til implementering af disse funktioner er der brugt udviklingsværktøjet Matlab pga. følgende fordele:

- Matlab understøtter styring af Analog Discovery
- Matlab forører databehandlingsfunktioner som ligger klar til anvendelse
- Matlab er god til at indlæse store mængde data med få kodelinjer
- I Matlab kan man udvikle en brugergrænseflade nemt og hurtigt
- Ved brug af Matlab med Analog Discovery, kan man betjene funktionsgeneratoren, dataopsamlingsenhed og brugergrænsefladen fra ét sted

Alternativet til overnævnte fordele er at man bruger forskellige “single purpose” programmer og enheder, når man skal måle et biomedicinsk signal.

I det følgende gennemgås de kritiske funktioner for programmet.

2.2. Software

2.2.1 Funktioner

Funktionernes nærmere beskrivelser henvises der til "bilag 6 - Design".

2.2.1.1 Syncerefleksmonitor_OpeningFcn

```
1 function Syncerefleksmonitor_OpeningFcn(hObject, eventdata, handles, varargin)
2 % This function has no output args, see OutputFcn.
3 % hObject    handle to figure
4 % eventdata reserved - to be defined in a future version of MATLAB
5 % handles    structure with handles and user data (see GUIDATA)
6 % varargin   command line arguments to Syncerefleksmonitor (see VARARGIN)
7
8 % Choose default command line output for Syncerefleksmonitor
9 handles = Start_GUI(handles);
10 % Update handles structure
11 guidata(hObject, handles);
```

2.2.1.2 Btn_Start_Measurements

```
1 function Btn_Start_Measurements_Callback(hObject, eventdata, handles)
2 % hObject    handle to Btn_Start_Measurements (see GCBO)
3 % eventdata reserved - to be defined in a future version of MATLAB
4 % handles    structure with handles and user data (see GUIDATA)
5 handles = Generate_SineWave(handles);
6 handles = Read_Measurements(handles);
7 handles = Process_Measurements(handles);
8 guidata(hObject,handles);
9 handles = Show_Measurements(handles);
10 set(handles.txtSynk, 'String', length(handles.locs_synk));
11 set(handles.txtSynk,'Visible','On');
```

2.2.1.3 Btn_Save_Measuerments

```
1 function Btn_Save_Measurements_Callback(hObject, eventdata, handles)
2 % hObject    handle to Btn_Save_Measurements (see GCBO)
3 % eventdata reserved - to be defined in a future version of MATLAB
4 % handles    structure with handles and user data (see GUIDATA)
5 handles = Save_Measurements(handles);
6 guidata(hObject,handles);
```

2.2.1.4 Start_GUI

```
1 function handles = Start_GUI(handles)
2 %% Denne funktion har til opgave at sætte tilføje objekter til GUI op når programmet startes.
3 set(handles.txtDate, 'String', (datestr(datestr(now)))); %Dato og tid vises
4
5 axes(handles.axes3); %Visning af billede i GUI
6 myImage = imread('SyncBillede.PNG');
7 image(myImage);
8 axis off
9 axis image
10
11 set(handles.Btn_Save_Measurements,'Visible','Off'); %Save Measurements knappen er ikke synlig
12 end
```

2.2. Software

2.2.1.5 Generate_SineWave

```
1 function handles = Generate_SineWave(handles)
2 % For at faa Analog Discovery til at fungere med Matlab skal denne saettes op med
3 % følgende indstillinger:
4 s = daq.createSession('digilent'); % Opretter forbindelse til Analog Discovery
5
6 fgenBI = addFunctionGeneratorChannel(s, 'AD1', 1, 'Sine'); % Opsætning af funktionsgenerator og
7 % waveform
8 fgenBI.Gain = 2; % Amplitude paa 2V
9
10 fgenBI.Frequency = 20000; % Frekvens paa 20 kHz
11
12 handles.GS = s; % Giver mulighed for starte en maaling vha. handles.GS
13 end
```

2.2.1.6 Read_Measurements

```
1 function handles = Read_Measurements(handles)
2 % Her bliver malingen oprettet og startet.
3 s = handles.GS; %Analog Discovery instillinger fra "Generate_SineWave" funktionen
4 aiBI = addAnalogInputChannel(s, 'AD1', 1, 'Voltage'); %Oprettelse af analog ind 1 og 2
5 aiEMG = addAnalogInputChannel(s, 'AD1', 2, 'Voltage');
6 s.Rate = 500000; % Samplingrate
7
8 sekunder = get(handles.popSec,'string');
9 selectedIndex = get(handles.popSec,'Value');
10 s.DurationInSeconds = str2double(sekunder{selectedIndex});
11
12 %s.DurationInSeconds = 2; % Tid pr. maaling
13
14 % Visning af message boksen "Measurements running..."
15 % h = figure('units','pixels','position',[300 300 300 80],'windowstyle','modal');
16 % uicontrol('FontSize',12,'style','text','string','Measurements
17 % running...','units','pixels','position',[40 30 200 20]);
17 set(handles.txtMeasure,'Visible','On');
18
19 [data, timestamps] = startForeground(s); % Maaling startes og genererer data og tid
20 %close(h); % Lukker message boksen "Measurements running..."
21 set(handles.txtMeasure,'Visible','Off');
22 % Malingerne gemmes i handles til videre brug
23 handles.BI = data(:,1);
24 handles.EMG = data(:,2);
25 %handles.data = data;
26 handles.timestamps = timestamps;
27
28 % Rydder op og slukker forbindelsen til Analog Discovery
29 s.release();
30 delete(s);
31 clear s;
32 end
```

2.2.1.7 Process_Measurements

```
1 function handles = Process_Measurements(handles)
2 % Digital envelope af BI signalet.
3
4
5 %% BI signal
6
7 % Dobbeltensret BI signal
```

2.2. Software

```
8 BIabs = abs(handles.BI);
9
10 % Udglattr BI signal med lavpas filter.
11 % Knaekfrekvens = 500 Hz
12 % Daemper en dekade frem til 5000 Hz
13 lpFilt = designfilt('lowpassfir', 'PassbandFrequency', 0.001, ...
14                      'StopbandFrequency', 0.01, 'PassbandRipple', 0.2, ...
15                      'StopbandAttenuation', 40, 'DesignMethod', ...
16                      'kaiserwin');
17
18 BIsignal = filter(lpFilt,BIabs); %BI signalet filteret
19
20
21 [p,s,mu] = polyfit((1:numel(BIsignal))',BIsignal,6);
22 f_y = polyval(p,(1:numel(BIsignal))',[],mu);
23
24 BI_data = BIsignal - f_y;
25
26 strom = str2double(get(handles.txtstrom,'String'));
27
28 BI_dataImpedans = (BI_data/(strom*10^-5));
29
30 downBI = downsample(BI_dataImpedans,20000);
31
32 smoothBI = smooth(downBI);
33
34 BI_inverted = -smoothBI;
35 [~,locs_synk] = findpeaks(BI_inverted,'MinPeakHeight',4.5,...
36                           'MinPeakDistance',10);
37
38 handles.locs_synk = locs_synk;
39
40 handles.BIsignal = smoothBI; %BI signalet gemmes i handles til senere visning
41
42 %% EMG
43
44 [p,s,mu] = polyfit((1:numel(handles.EMG))',handles.EMG,6);
45 f_y = polyval(p,(1:numel(handles.EMG))',[],mu);
46
47 EMG_data = handles.EMG - f_y;
48
49 downEMG = downsample(EMG_data,20000);
50 smoothEMG = smooth(downEMG);
51
52 handles.EMGsignal = smoothEMG;
53 %% TID
54
55 handles.TID = downsample(handles.timestamps,20000);
56
57 end
```

2.2.1.8 Show_Measurements

```
1 function handles = Show_Measurements(handles)
2 % De optaget signaler BI og EMG plottes
3
4 axes(handles.axes1); % Vises i axes1
5 subplot(2,1,1)
6 plot(handles.TID, handles.BIsignal,'k'); % Det behandlet BI signal
7 hold on
8 plot(handles.TID(handles.locs_synk),handles.BIsignal(handles.locs_synk),'o',...
9       'MarkerFaceColor','b');
9 title('Bioimpedance')
10 ylabel('Impedance (ohm)');
```

2.2. Software

```
11 axis([1 12 -10 5])
12
13
14 subplot(2,1,2)
15 plot(handles.TID, handles.EMGsignal,'k'); % EMG signalet
16 title('EMG')
17 xlabel('Time in seconds')
18 ylabel('Voltage (V)');
19 axis([1 12 -0.2 0.2])
20 end
```

2.2.1.9 Save_Measurments

```
1 function handles = Save_Measurements(handles)
2 % Maalinger gemmes i csv-fil.
3 [FileName,PathName] = uiputfile('.csv'); % Oprettelse af prompt til brugeren, hvor filnavn og sti
4 % skal bestemmes
5 filepath=fullfile(PathName,FileName); % Sti og filnavn sammensættes
6
7 % Message boksen "Saving measurements..." vises
8 % f = figure('units','pixels','position',[300 300 300 80],'windowstyle','modal');
9 % uicontrol('FontSize',12,'style','text','string','Saving
10 % measurements...','units','pixels','position',[40 30 200 20]);
11 % pause(1); % Pauser i 1 sek for at message boksen kan naa at blive vist under gemmeprocessen
12 % starter
13 set(handles.txtSave,'Visible','On');
14 pause(2);
15 csvwrite(filepath,{{handles.BI,handles.EMG,handles.timestamps}}); % Behandlet BI signal, EMG signal
16 % og tid samles i en csv-fil.
17 %close(f); %Lukker message boksen igen efter csv-filen er gemt
18 set(handles.txtSave,'Visible','Off');
19 h = msgbox('Measurements saved!'); % Der informeres om at csv-filen er gemt
20 end
```

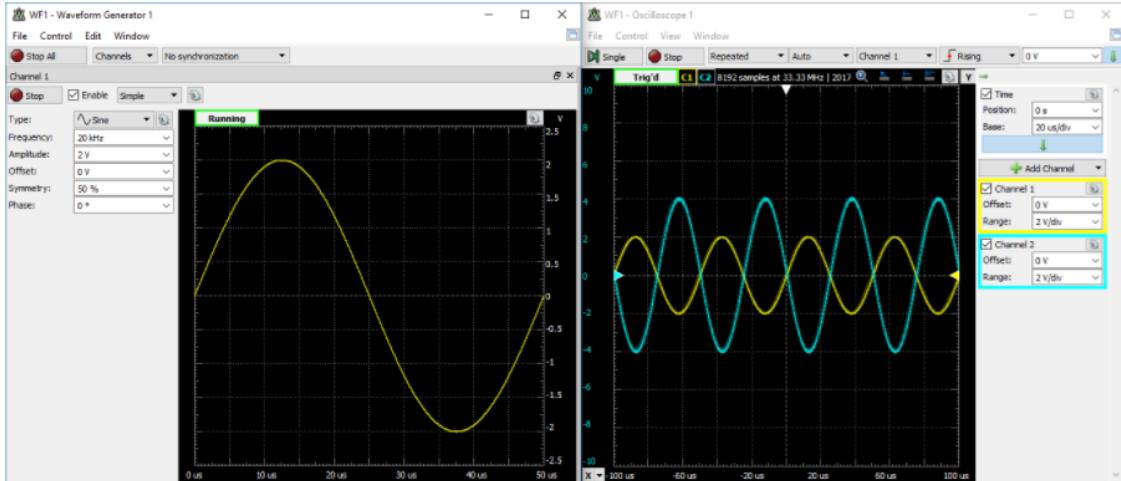
Modultest 3

Modultest af hardware-delen består af en simuleret test vha. Multisim og en praktisk test. Nogle komponenterne eksister ikke i Multisim og kræver at blive oprettet, men det er valgt at ikke bruge tid på det, da det er tidskrævende. Derfor præsenteres kun praktiske resultater for disse komponenter. I det følgende præsenteres testresultaterne for instrumentationsforstærker 1, 2 , strømgeneratoren, operationsforstærkeren og AA filteret.

3.1 Hardware

3.1.1 Instrumentationsforstærker 1

Dette modul er testet ved at sende 2V/20kHz fra Analog Discovery (den gule kurve) igennem INA128. Det ses på figur 3.1 at de 2V bliver forstærket til 4V (turkis kurve) ved udgangen af INA128. Dette resultat stemmer overens med det beregnede resultat i designfasen.

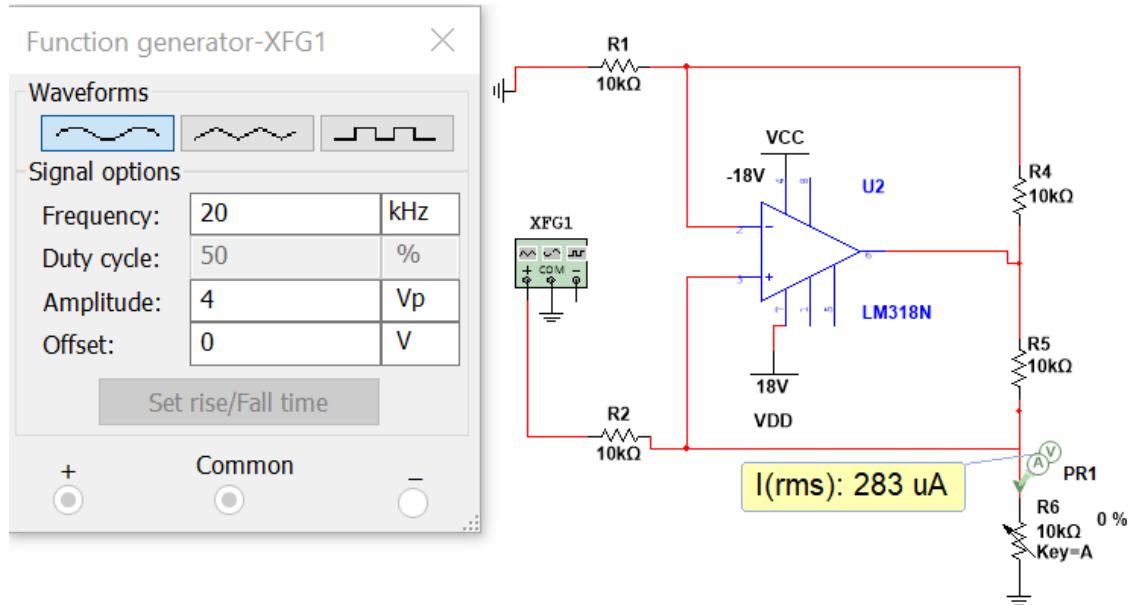


Figur 3.1: Figuren viser resultatet af INA128, som forstærker 2V til 4V

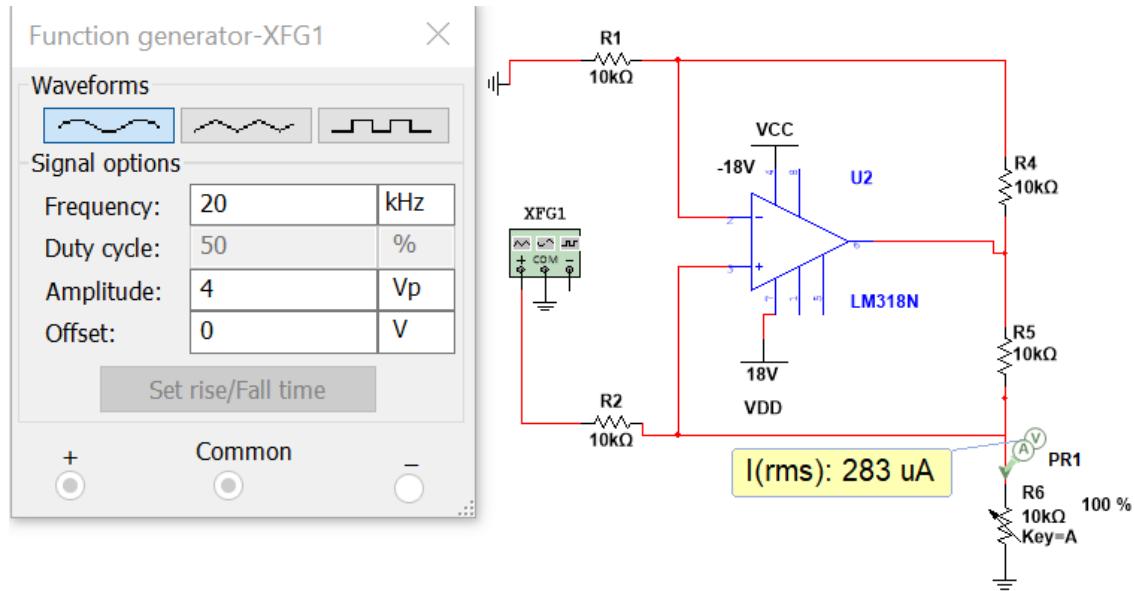
3.1. Hardware

3.1.2 Strømgenerator

Strømgeneratoren er simuleret ved at den får 4V fra en funktionsgenerator. På baggrund af denne spænding genereres der 283 μ A ud af strømgeneratoren. Bemærk at figur 3.2 viser strømudgangen ved no-load, hvorimod figur 3.3 viser når man belaster strømgeneratoren med 10k Ω .



Figur 3.2: Figuren viser det simuleret resultat for strømgeneratoren ved no-load

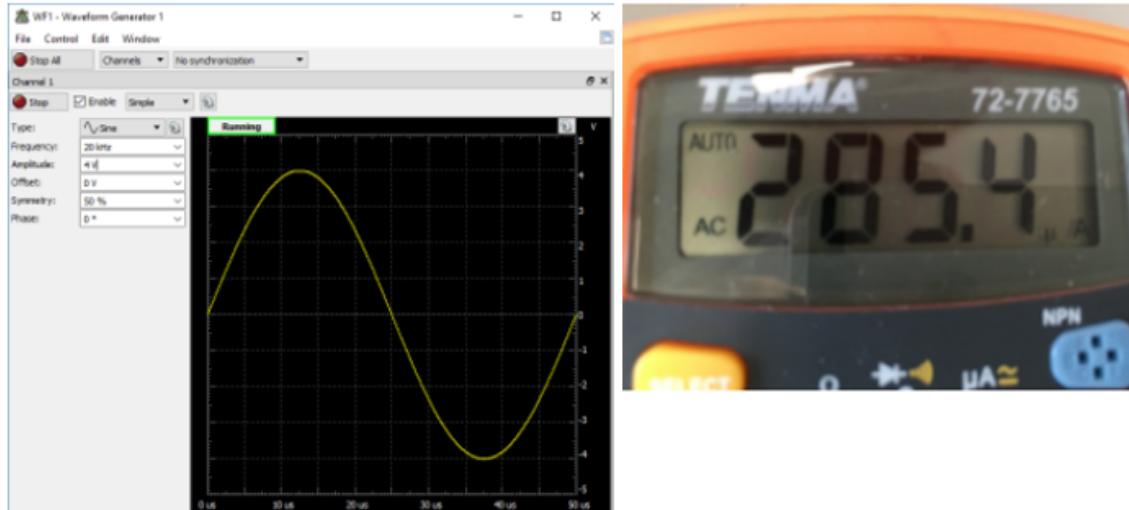


Figur 3.3: Figuren viser det simuleret resultat for strømgeneratoren med load på 10k Ω

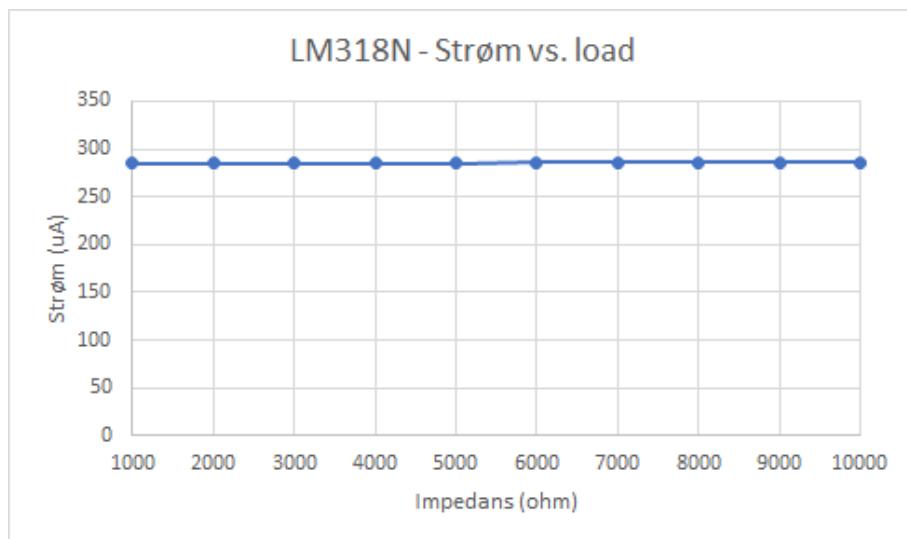
Det ses på figur 3.3 at strømmen ikke ændrer sig, selvom man belaster kredsløbet med 10k Ω .

3.1. Hardware

Tilsvarende sendes der 4V ind i strømgeneratoren, når der skal foretages den praktiske test på fumlebræt. De 4V genereres fra Analog Discovery. Den producerede strøm måles vha. et amperemeter i serie på udgangen af den anvendte operationsforstærker, LM318. Det ses på figur 3.4 at udgangsstrømmen, som genereres af strømgeneratoren er målt til 285 μ A. Dette resultat afviger lidt fra det beregnede og simulerede resultat. Det vurderes at afvigelsen er så lille at den ingen betydning har for måleobjektets sikkerhed. Figur 3.5 viser strømmens stabilitet fra 1 k Ω til 10 k Ω , hvilket svarer til området for load impedansen for et biologisk væv [1]. Derfor er det besluttet at arbejde videre med den målte strøm.



Figur 3.4: Figuren viser det praktisk strømresultat som er målt ved udgangen af strømgeneratoren.



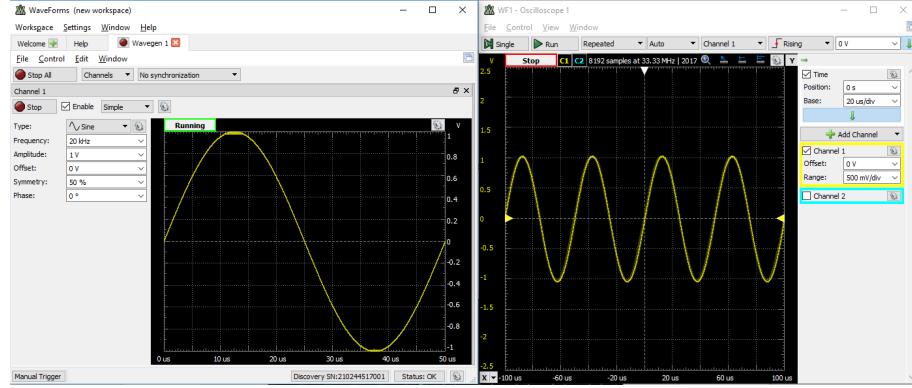
Figur 3.5

3.1.3 Instrumentationsforstærker 2

Til test af instrumentationsforstærker 2 er der benyttet en spændingsdeler, som får 1V fra en funktionsgeneratoren, Analog Discovery. Spændingsdelerens funktion er at reducere 1V til 10mV, som efterfølgende forstærkes af instrumentationsforstærker 2 med faktor 100

3.1. Hardware

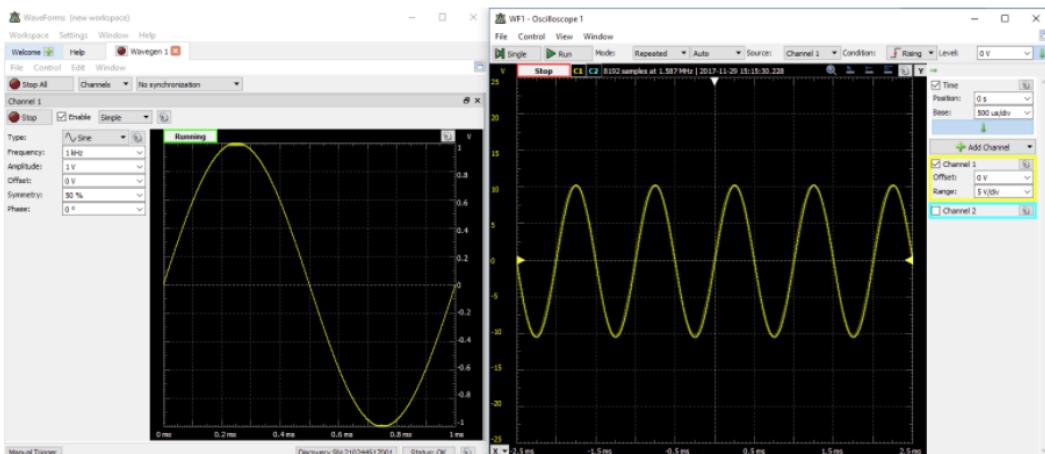
gange. Som figur 3.6 viser, kan man med instrumentationsforstærkeren INA128 opnå en forstærkning på 100. Dette resultat stemmer overens med det teoretiske resultat, som er beregnet i designfasen.



Figur 3.6: Figuren viser det praktiske udgangsspænding for INA128. Denne INA128 yder en forstærkning på 100 gange.

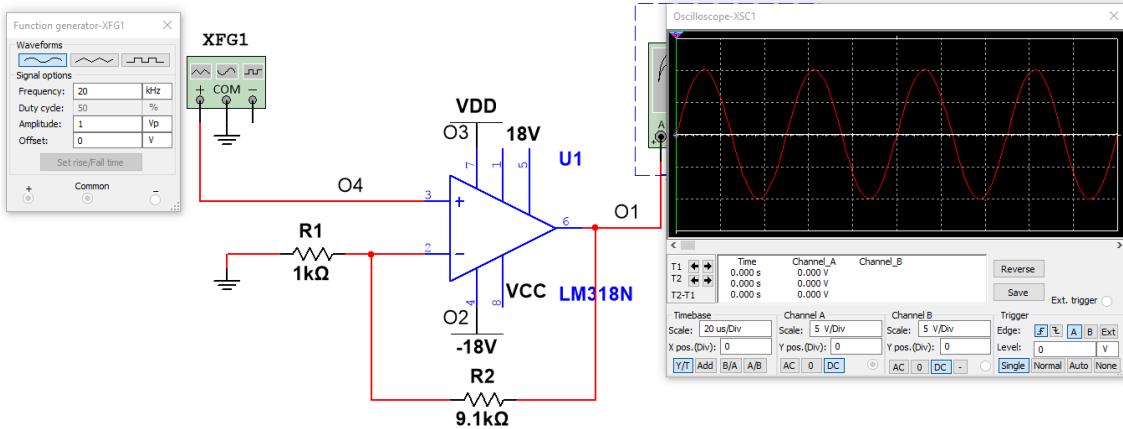
3.1.4 OP-AMP

Denne operationsforstærkers opgave er at forstærke 1 V til 10 V, dvs. en forstærkning ved faktor 10. Der sendes 1V fra funktionsgeneratoren, Analog Discovery, som derefter bliver forstærket til 10 V. Figur 3.7 viser at signalet fra funktionsgeneratoren bliver forstærket 10 gange, dvs. den målte udgangsspænding er på 10V. Hermed stemmer den målte og den teoretiske udgangsspænding af operationsforstærkeren LM318 med hinanden.



Figur 3.7: Figuren viser det praktisk udgangsspænding som er målt ved udgangen af operationsforstærkeren.

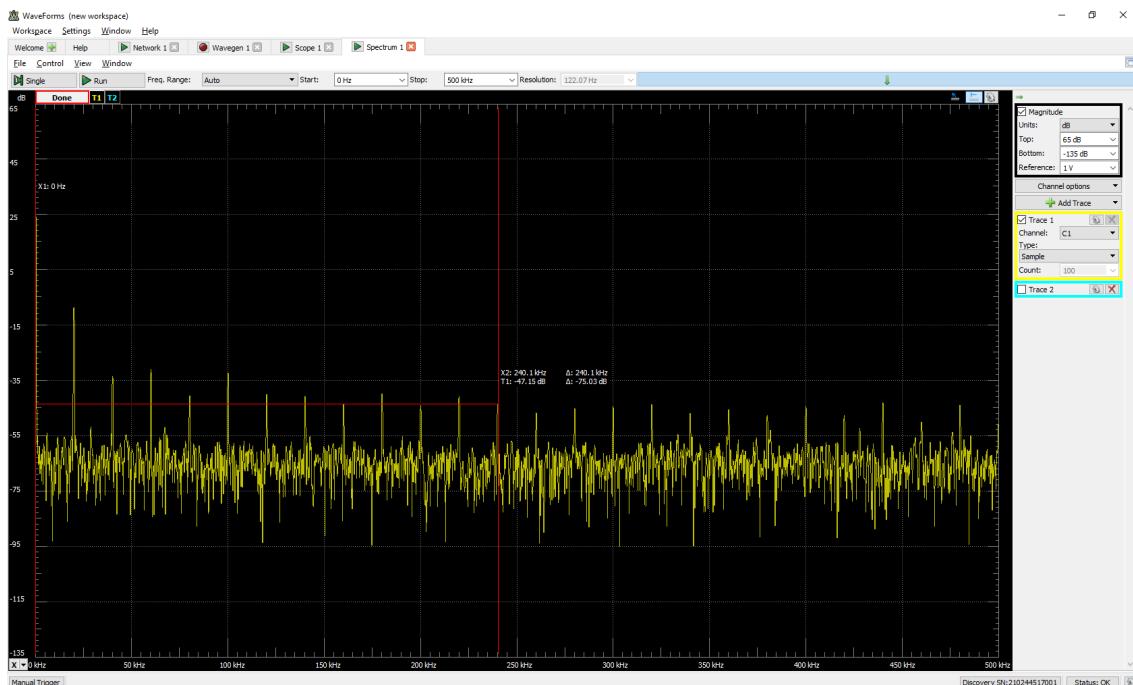
3.1. Hardware



Figur 3.8: Figuren viser det simuleret udgangsspænding som er målt ved udgangen af operationsforstærkeren.

3.1.5 Spektrumanalyse

Ved at sammensætte alle komponenter inkl. elektroder, kan spektrumanalysen blive optaget vha. Analog Discovery. Som det kan aflæses på figur 3.9, er der en dæmpning fra det samlet kredsløb på ca. -75 dB. Dette aflæses fra den højeste peak i passbåndet til ca. 250 kHz, som er den halve samplingfrekvens. Da Analog Discovery arbejder med 14bit (skal dæmpe ned til ca. -90 dB) kræver det en yderligere dæmpning på ca. 15 dB. Dette kunne realiseres med et 1. ordens filter, men der vælges at benytte et 2. ordens filter, for at undgå de variationer der måtte være i passbåndet i det målte signal.

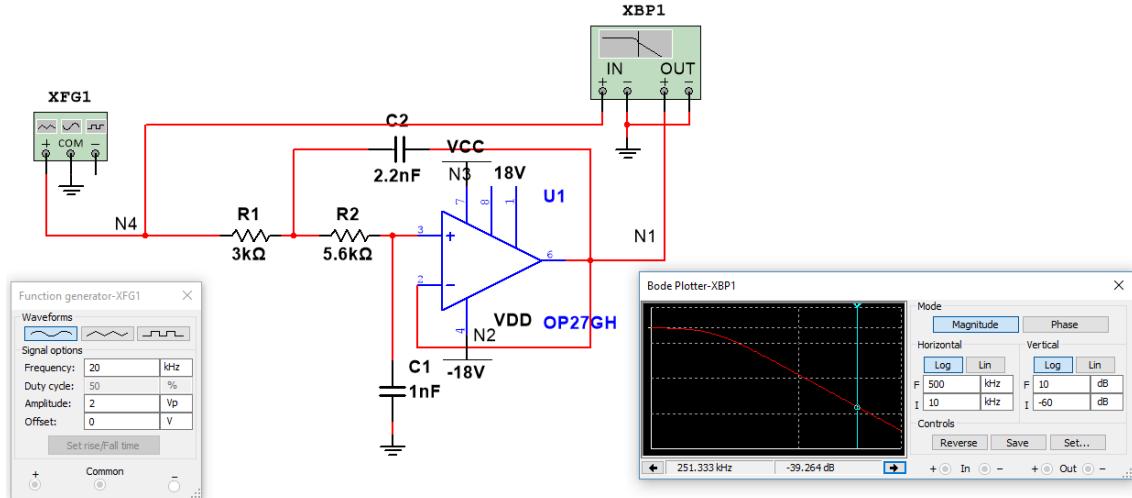


Figur 3.9: Det implementeret frekvensspektrum, som viser at der er en samlet dæmpning på ca. 75dB som skal yderligere dæmpes til 90dB med et 2.ordens lavpasfilter.

3.1. Hardware

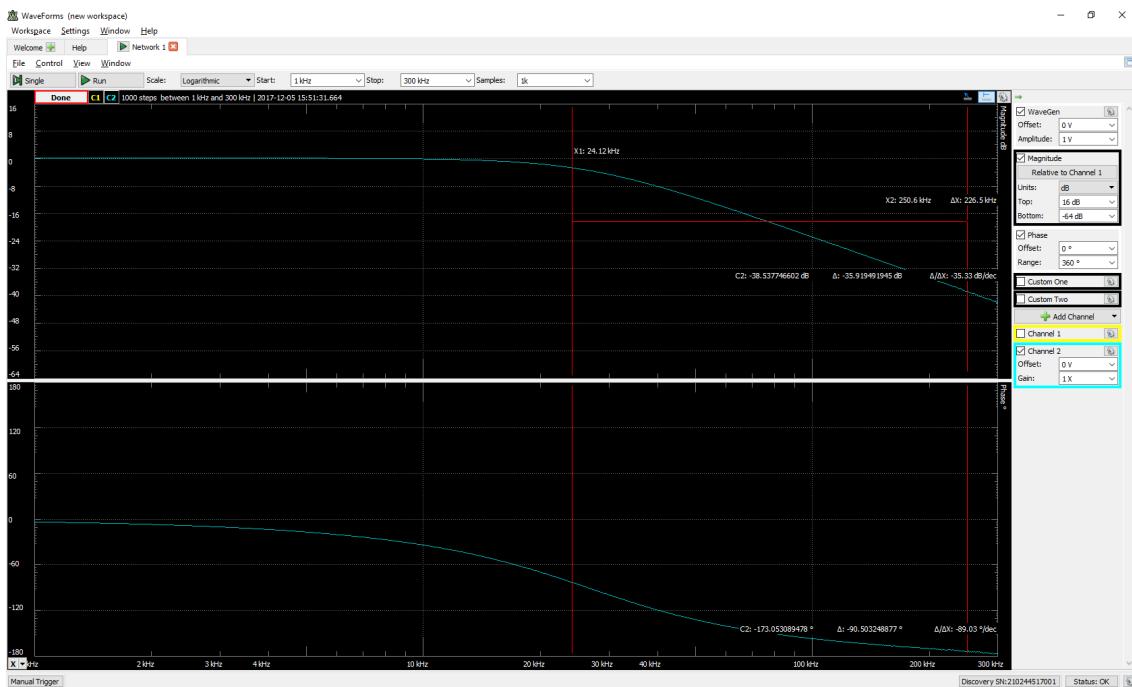
3.1.6 AA filter

Først er AA filteret simuleret i Multisim. Ved hjælp af Bode Plotter i Multisim, kan det aflæses ved 250 kHz at der er en dæmpning på 40 dB.



Figur 3.10: Resultat om filterets virkning simuleret i Multisim

Der er brugt værktøjet Network Analyzer i programmet Waveforms, for at teste filters virkning, se figur 3.15. Det kan konstateres at ved knækfrekvensen (25 kHz) er faseforskydningen ca. 90°. Efterfølgende kan det aflæses, at ved knækfrekvensen og en dekade frem er der en dæmpning på ca. 35 dB.



Figur 3.11: Resultat om filterets virkning fra Network Analyzer i waveforms.

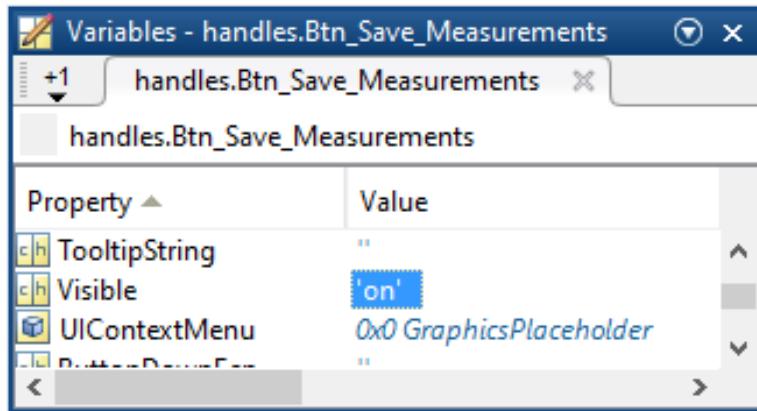
3.2. Software

3.2 Software

3.2.1 Funktioner

3.2.1.1 Synkerefleksmonitor_OpeningFcn

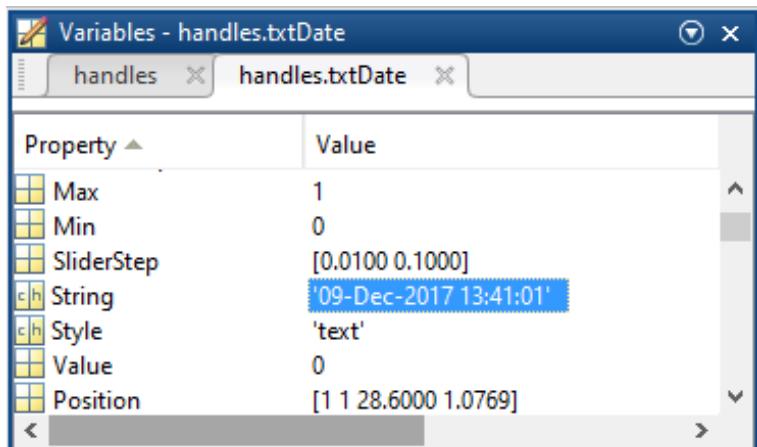
3.2.1.2 Btn_Start_Measurment



Figur 3.12: Oprettelse af dato og tid.

3.2.1.3 Btn_Save_Measurment

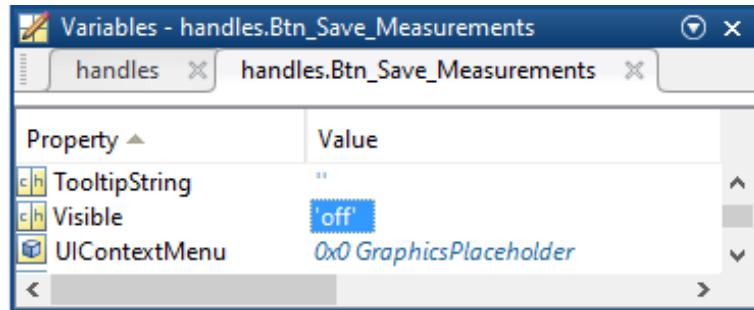
3.2.1.4 Start_GUI



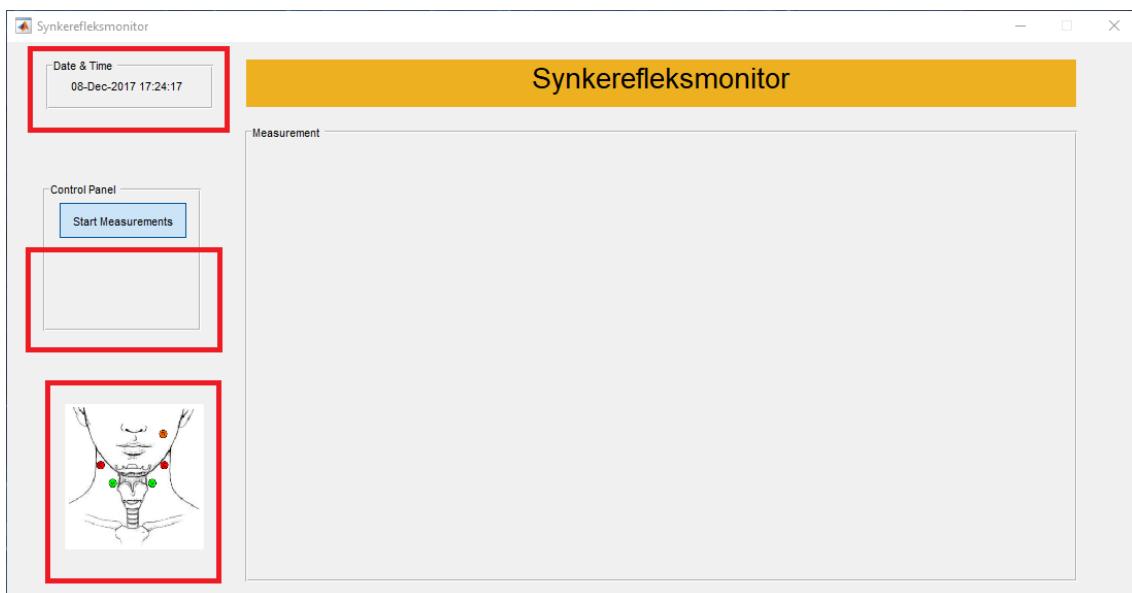
Figur 3.13: Oprettelse af dato og tid.

3.2. Software

3.2.1.5 Start_GUI



Figur 3.14: Save_Measurements knappen er gemt fra start.



Figur 3.15: Resultat om koden i "Start_GUI"

3.2.1.6 Generate_SineWave



Figur 3.16: Resultat om koden i "Generate_SineWave"

3.2. Software

Variables - s	
s x s.Channels x	
1x1 Session	
Property	Value
UserData	[]
Vendor	1x1 <i>diligent</i>
Channels	1x1 <i>FunctionGeneratorStandardSineCh</i>
Connections	0x0 Connection
IsRunning	0

Figur 3.17: Oprettelsen til Analog Discovery

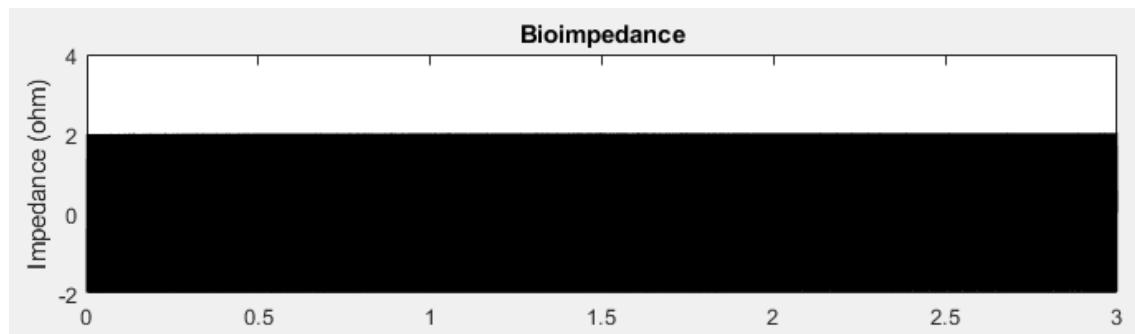
Variables - s.Channels	
s x s.Channels x	
s.Channels	
Property	Value
Phase	0
Range	1x1 <i>Range</i>
TerminalConfig	'SingleEnded'
Gain	2
Offset	0
Frequency	20000
WaveformType	1x1 <i>WaveformType</i>
FrequencyLimit	1x1 <i>ParameterLimit</i>
Name	"
ID	'1'
Device	1x1 <i>DeviceInfo</i>
MeasurementType	'Sine'

Figur 3.18: Funktionsgenerator tilføjet.

3.2. Software

Variables - handles	
	s x s.Channels x handles x
1x1 struct with 11 fields	
Field ▲	Value
figure1	1x1 Figure
uipanel7	1x1 Panel
uipanel6	1x1 Panel
text2	1x1 UIControl
uipanel1	1x1 Panel
axes3	1x1 Axes
txtDate	1x1 UIControl
axes1	1x1 Axes
Btn_Save_Measurement	1x1 UIControl
Btn_Start_Measurement	1x1 UIControl
GS	1x1 Session

Figur 3.19: Variablen "handles.GS" er oprettet i handles til brug i funktionen "Read_Measurements".



Figur 3.20: GUI resultat om koden i "Generate_SineWave"

3.2.1.7 Read_Measurements

Variables - s	
	s x aiBL x aiEMG x s.Channels x
1x1 Session	
Property ▲	Value
NumberOfScans	1000000
DurationInSeconds	2
Rate	500000
<input checked="" type="checkbox"/> IsContinuous	0
NotifyWhenDataArrives	50000

Figur 3.21

3.2. Software

s.Channels			
	1	2	3
1	1x1 Function...	1x1 AnalogInputVoltageChannel	1x1 AnalogInputVoltageChannel
2			
3			

Figur 3.22

Workspace	
Name	Value
aiBI	1x1 AnalogInputVolt...
aiEMG	1x1 AnalogInputVolt...
h	1x1 Figure
handles	1x1 struct
s	1x1 Session

Figur 3.23

 data	1000000x2 double
 timestamps	1000000x1 double

Figur 3.24

3.2. Software

Variables - data

data

1000000x2 double

	1	2
1	0.6462	0.0066
2	0.1543	-5.9705e-04
3	-0.3412	-0.0077
4	-0.8190	-0.0149
5	-1.2442	-0.0256
6	-1.5885	-0.0292
7	-1.8380	-0.0364
8	-1.9644	-0.0399
9	-1.9715	-0.0328
10	-1.8520	-0.0292
11	-1.6166	-0.0256
12	-1.2723	-0.0221
13	-0.8577	-0.0113
14	-0.3868	-0.0042
15	0.1121	0.0101
16	0.6075	0.0173
17	1.0608	0.0244
18	1.4543	0.0280
19	1.7459	0.0387
20	1.9392	0.0352
21	2.0024	0.0352
22	1.9427	0.0352
23	1.7670	0.0280

3.2. Software

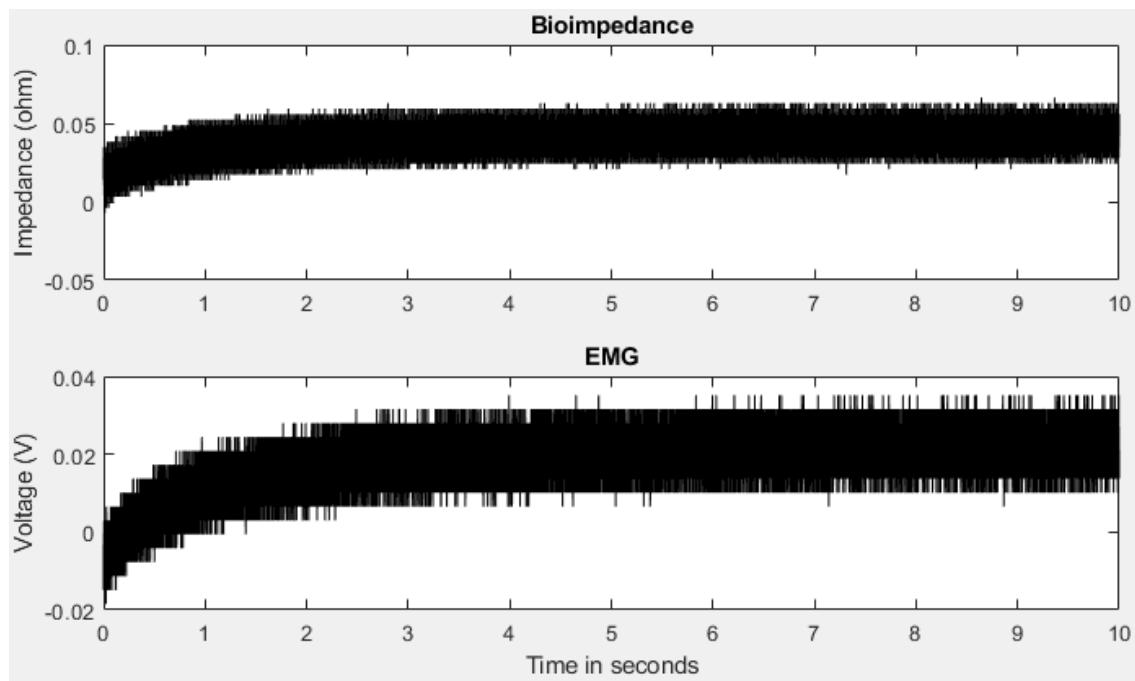
handles	
1x1 struct with 13 fields	
Field	Value
figure1	1x1 Figure
uipanel7	1x1 Panel
uipanel6	1x1 Panel
text2	1x1 UIControl
uipanel1	1x1 Panel
axes3	1x1 Axes
txtDate	1x1 UIControl
axes1	1x1 Axes
Btn_Save_Measurem...	1x1 UIControl
Btn_Start_Measurem...	1x1 UIControl
GS	1x1 Session
data	1000000x2 double
timestamps	1000000x1 double

Figur 3.26

Workspace	
Name	Value
aiBI	1x1 AnalogInputVolt...
aiEMG	1x1 AnalogInputVolt...
data	1000000x2 double
h	1x1 Figure
handles	1x1 struct
timestamps	1000000x1 double

Figur 3.27: Session "s" er stoppet og fjernet fra workspace.

3.2. Software



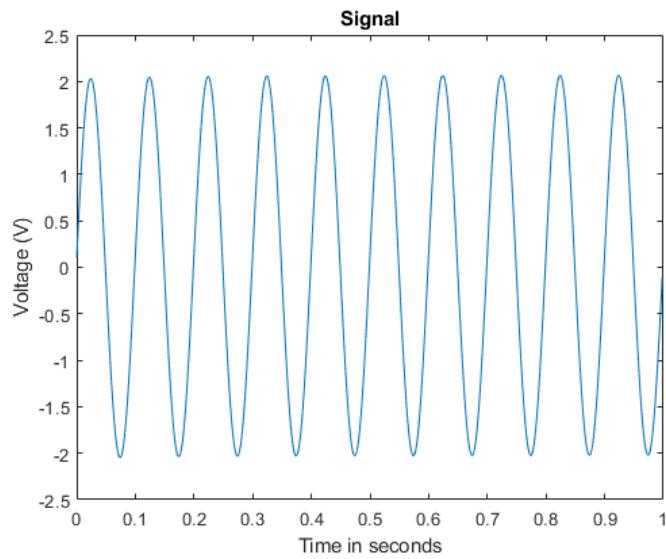
Figur 3.28: Resultat om koden i "Read_Measurements"

3.2.1.8 Process_Measurements

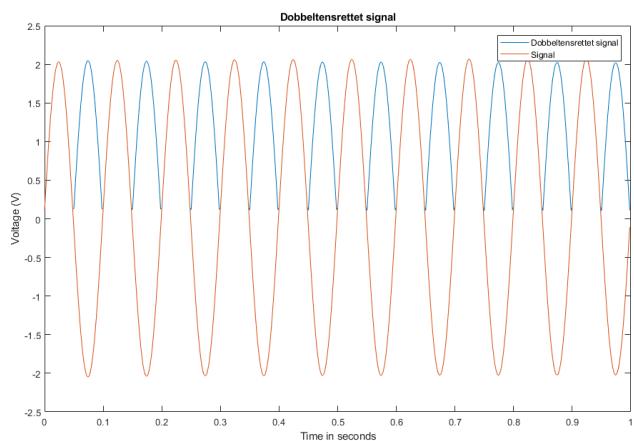


Figur 3.29: Resultat om koden i "Generate_SineWave"

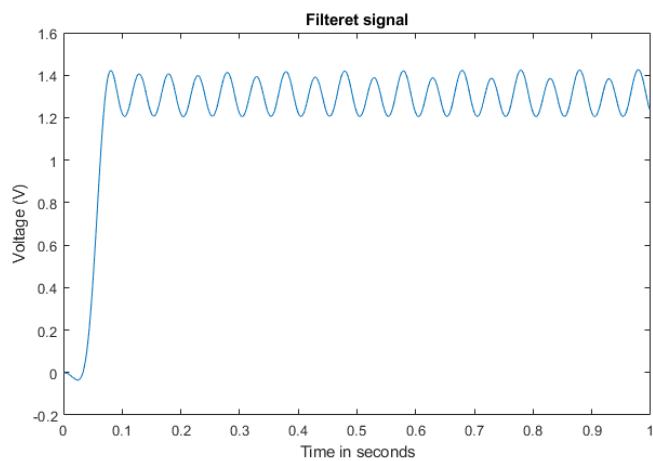
3.2. Software



Figur 3.30: Resultat om koden i "Read_Measurements"



Figur 3.31: Resultat om koden i "Read_Measurements"



Figur 3.32: Resultat om koden i "Read_Measurements"

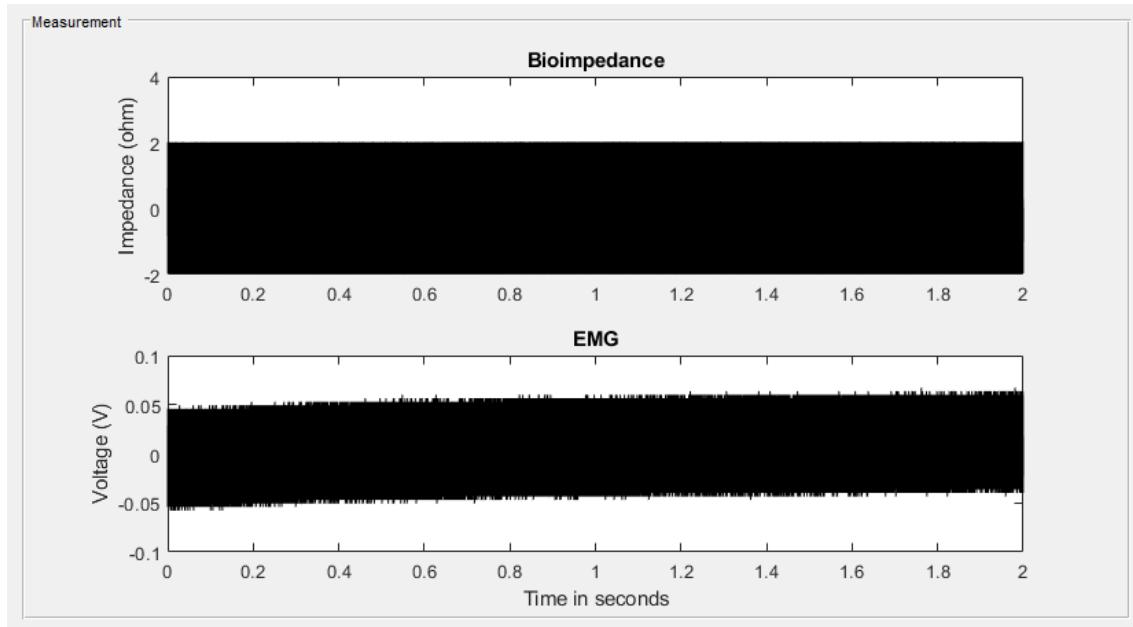
3.2. Software

Field	Value
figure1	1x1 Figure
uipanel7	1x1 Panel
uipanel6	1x1 Panel
text2	1x1 UIControl
uipanel1	1x1 Panel
axes3	1x1 Axes
txtDate	1x1 UIControl
axes1	1x1 Axes
Btn_Save_Measurem...	1x1 UIControl
Btn_Start_Measurem...	1x1 UIControl
GS	1x1 Session
data	1000000x2 double
timestamps	1000000x1 double
Blsignal	1000000x1 double

Figur 3.33

3.2. Software

3.2.1.9 Show _ Measurements



Figur 3.34

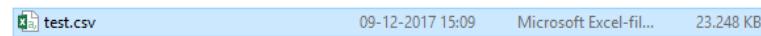
3.2.1.10 Save _ Measurements

Workspace	
Name	Value
FileName	'test.csv'
filepath	'C:\Users\mbana\Desktop\test.csv'
handles	1x1 struct
PathName	'C:\Users\mbana\Desktop\'

Figur 3.35



Figur 3.36



Figur 3.37

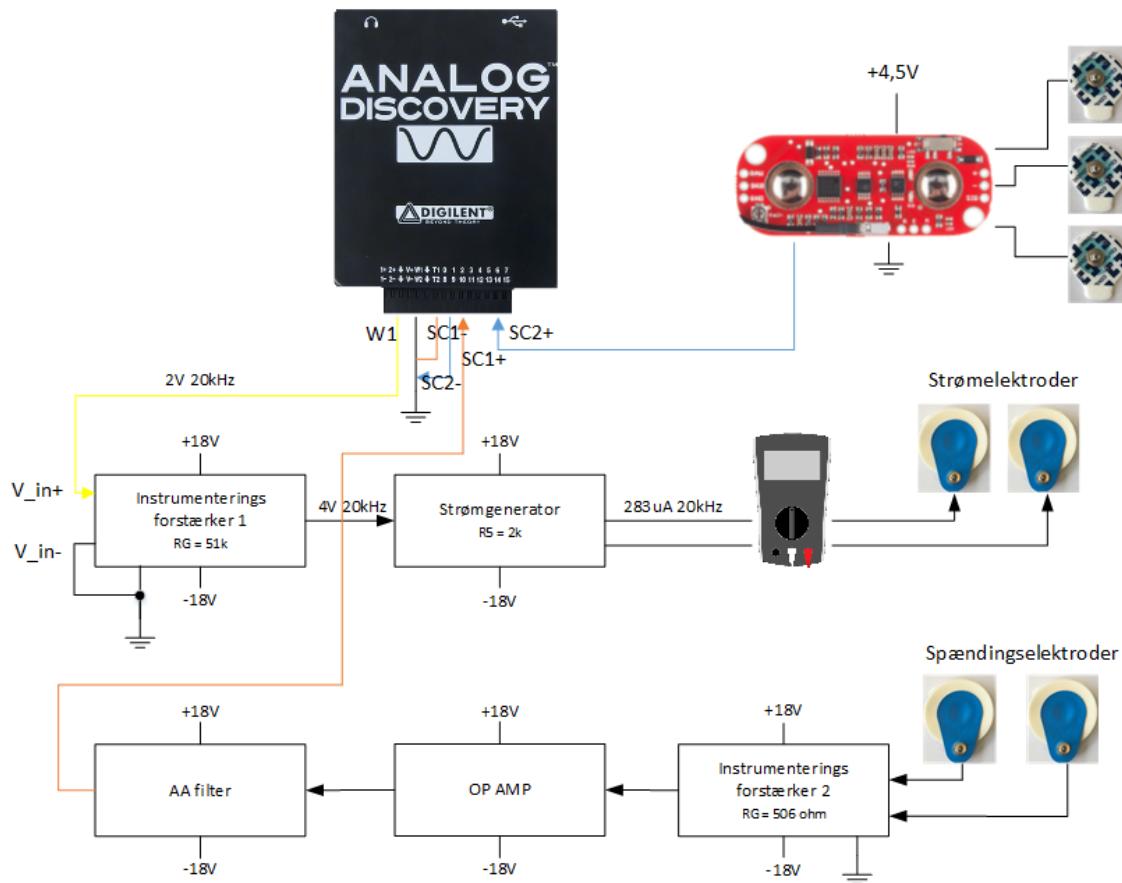


Figur 3.38

Integrationstest 4

4.1 Indledning

Efter modultesten sammensættes de enkelte hardware komponenter gradvist til et færdigt system. I det følgende afsnit dokumenteres ved brug af billeder af opstillingen, multimeter og Analog Discovery. På figur 4.1 er det samlet integreret system inkl. EMG-måler.

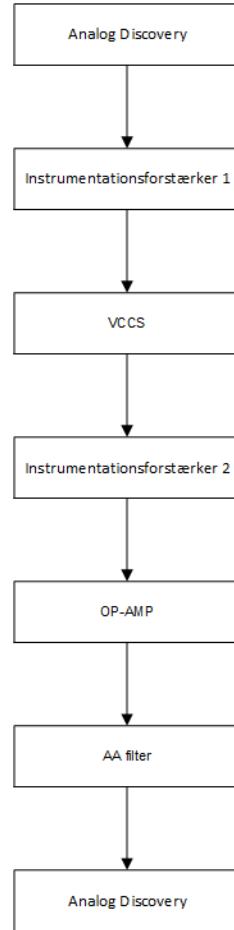


Figur 4.1

4.1.1 Hardware Top-down test

Ved hjælp af Top-down test styres denne integration af det samlet system som kan ses på figur 4.2. Undervejs er input og output blevet noteret for hver komponent. Forløbet kan ses i tabel 4.1.

4.1. Indledning

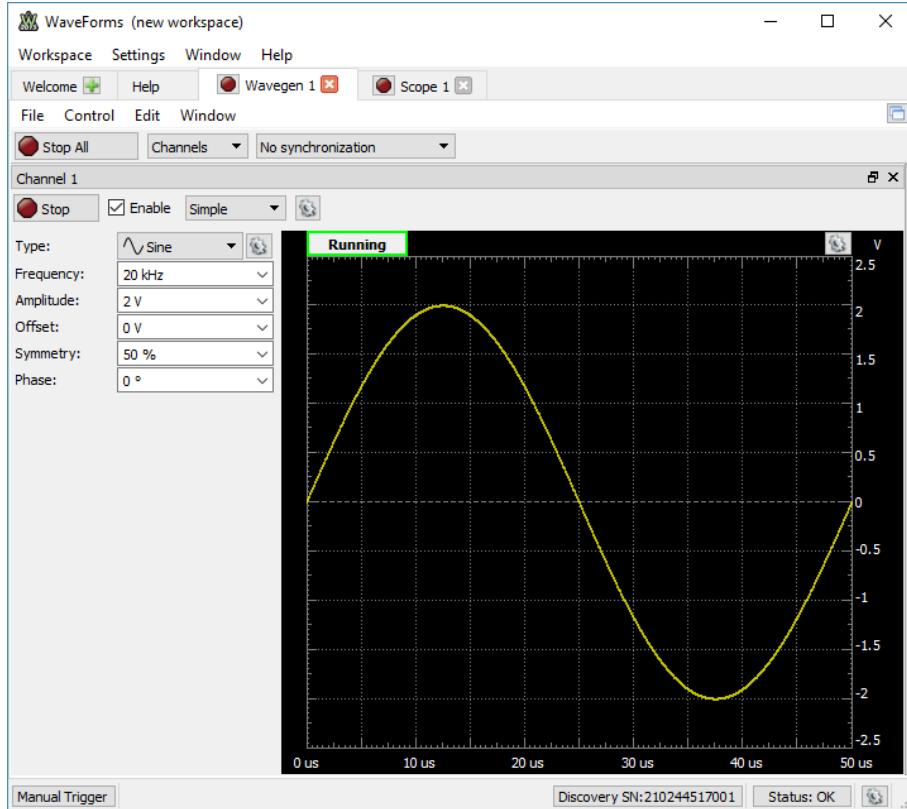


Figur 4.2: Top-down test af integrationstesten.

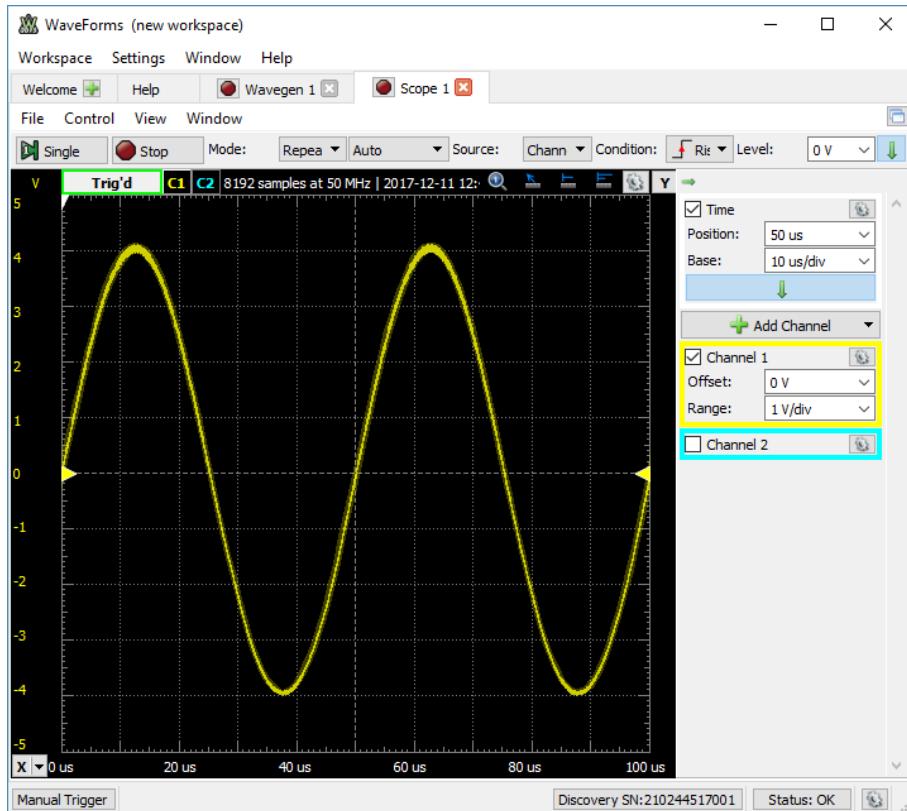
Navn	Input	Output	Kommentar
Analog Discovery		2 V / 20 kHz	Se figur 4.3
Instrumentationsforstærker 1	2 V / 20 kHz	4 V / 20 kHz	Se figur 4.4
VCCS	4 V / 20 kHz	283 uA / 20 kHz	Se figur 4.5
Instrumentationsforstærker 2	Biosignal	1,67 V	Se figur 4.6. Strøm faldet til 104 uA, se figur 4.7
Op-AMP	1,67 V	14,3 V	Se figur 4.8
AA filter	14,3 V	13,4 V	Se figur 4.9 og for spektrum analyse se figur 4.10.

Tabel 4.1: Oversigt over input og output for hver komponent, med henvisning enten til figur eller billede.

4.1. Indledning



Figur 4.3: Output på Analog Discovery.

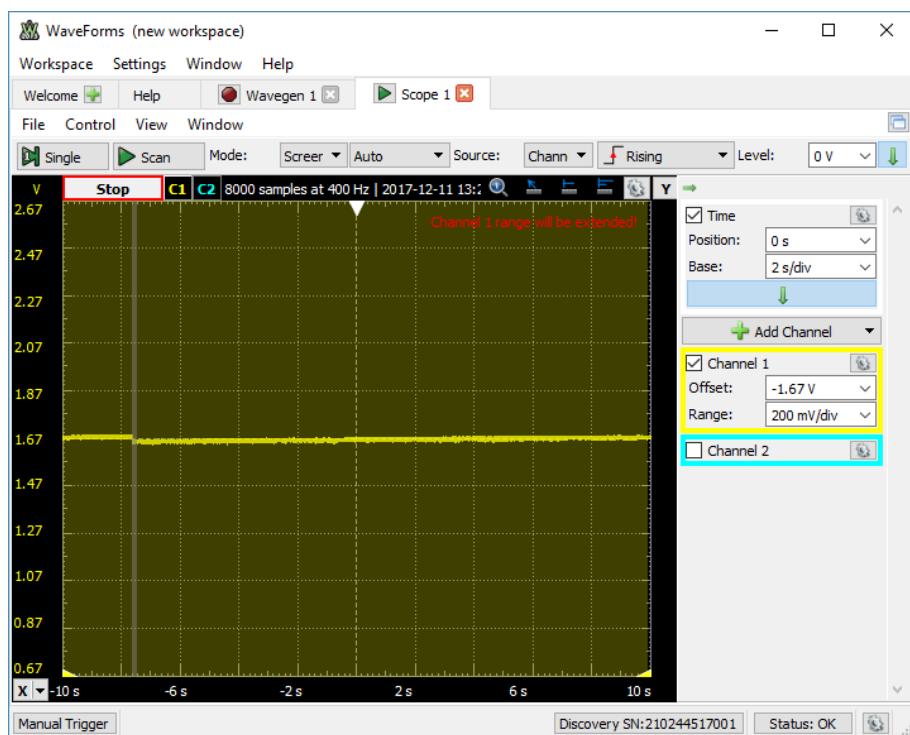


Figur 4.4: Output på Instrumentationsforstærker 1.

4.1. Indledning



Figur 4.5: Output på VCCS

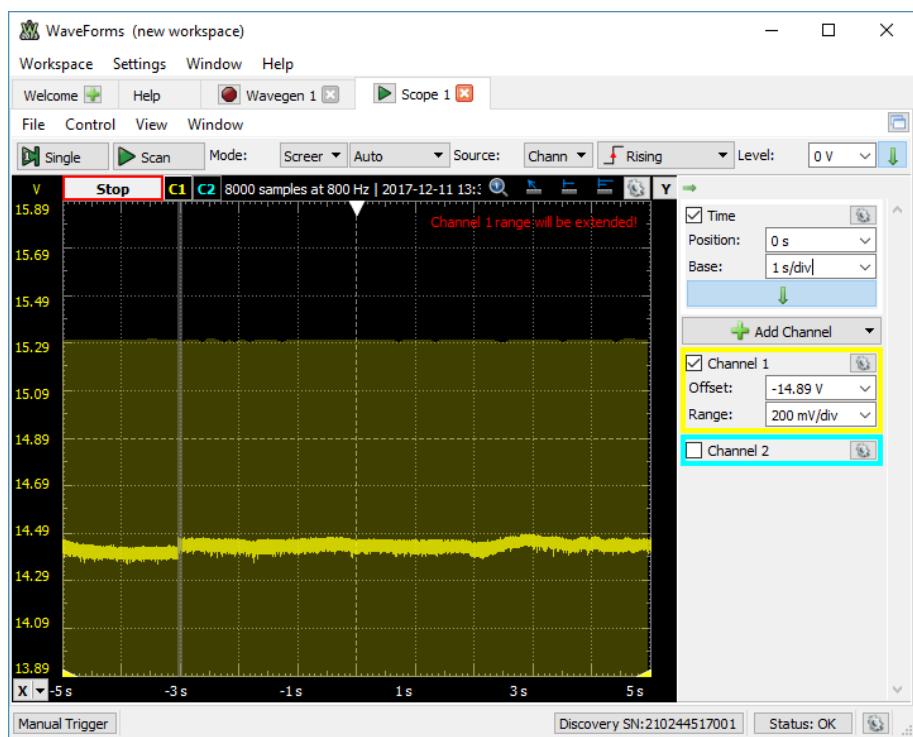


Figur 4.6: Output på Instrumentationsforstærker 2

4.1. Indledning

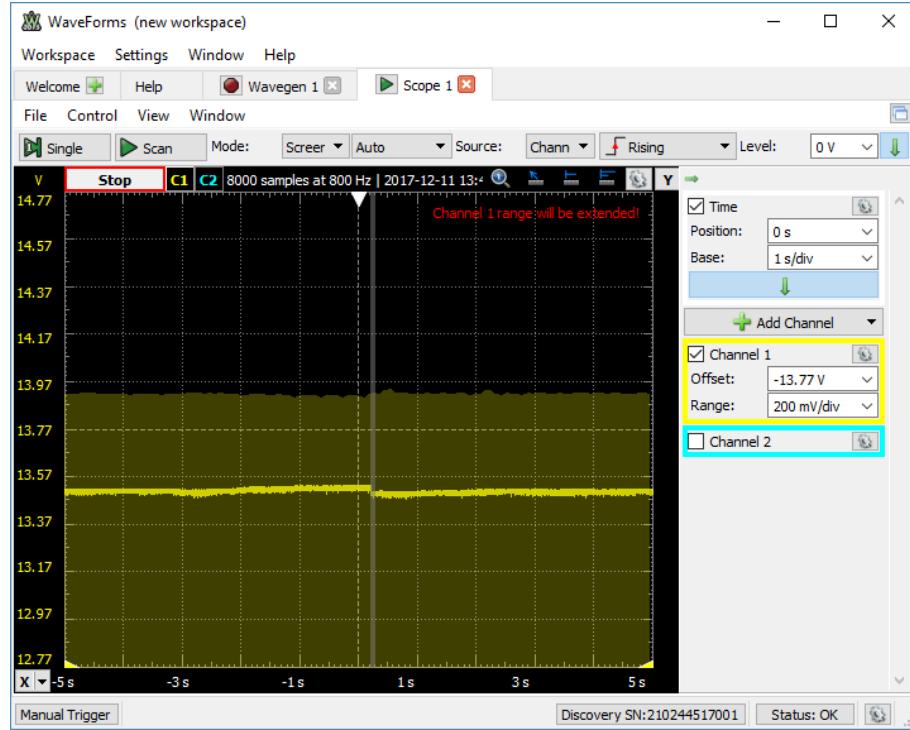


Figur 4.7: Strømmen var faldet til 104 μ A fra VCCS, da elektroderne påsættes.

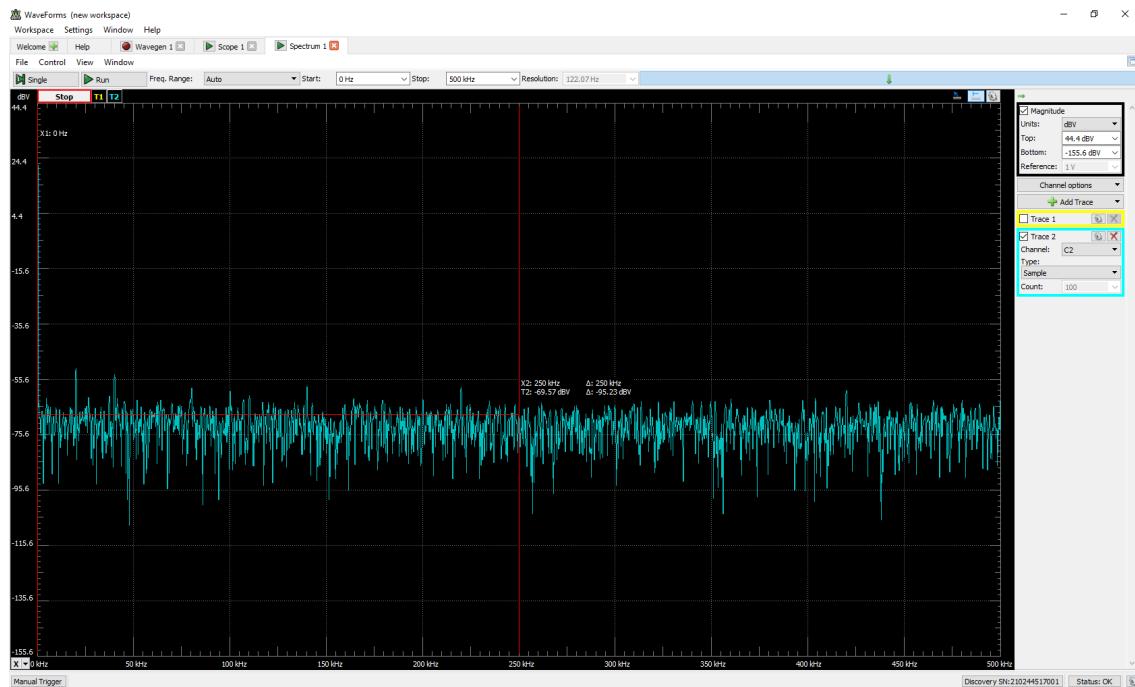


Figur 4.8: Output på OP-AMP

4.1. Indledning

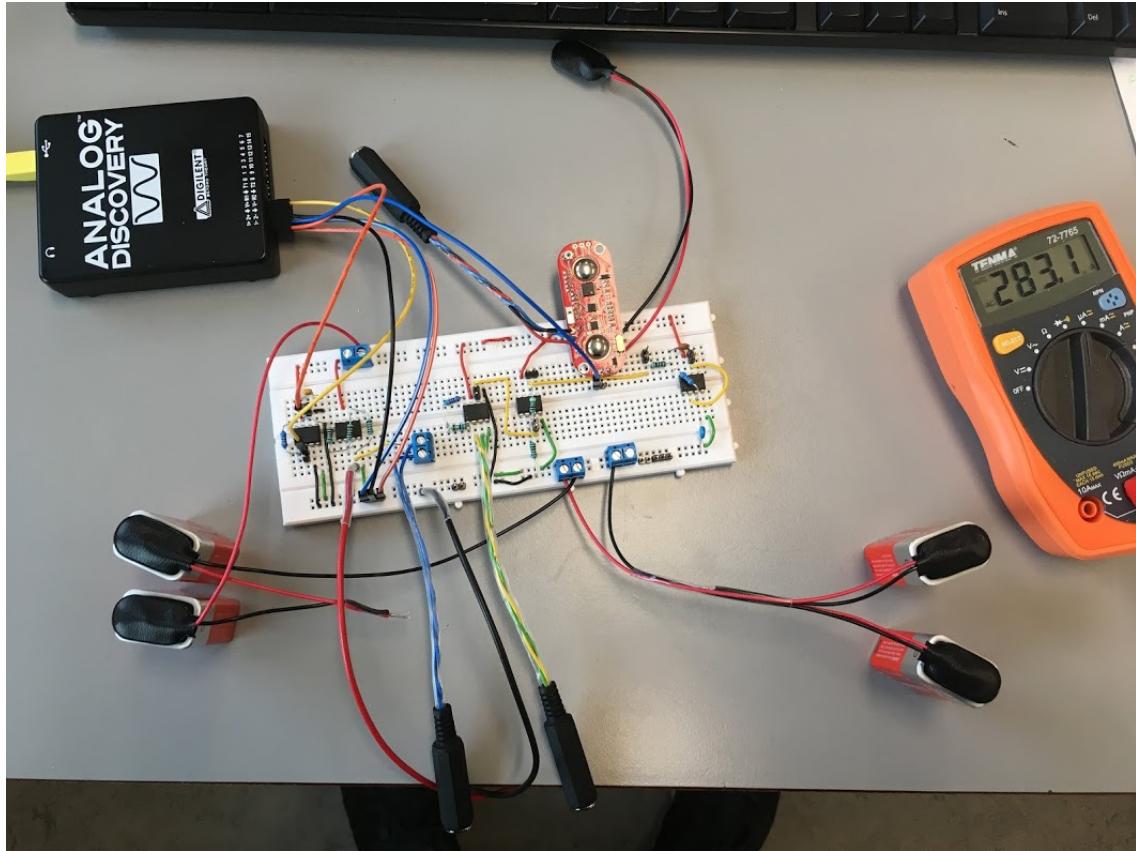


Figur 4.9: Output på AA filter



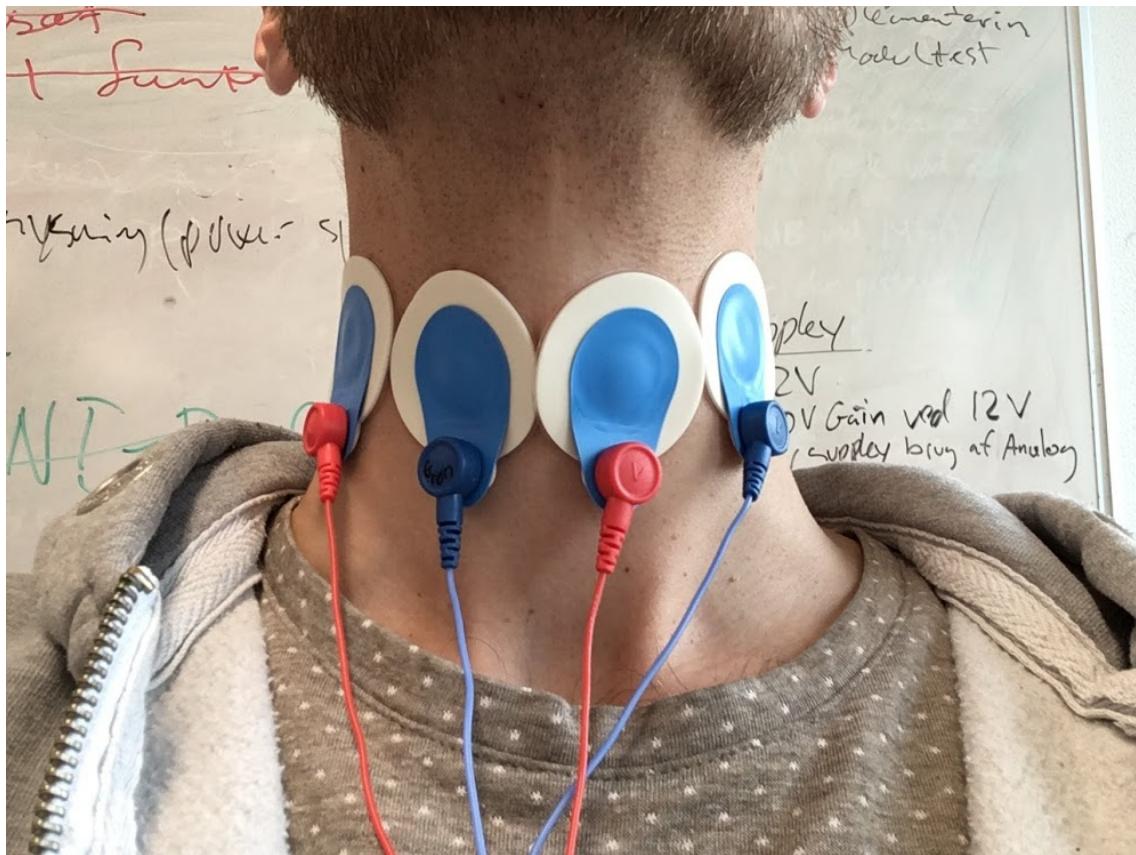
Figur 4.10: Frekvens spektrum af det samlet system.

4.1. Indledning



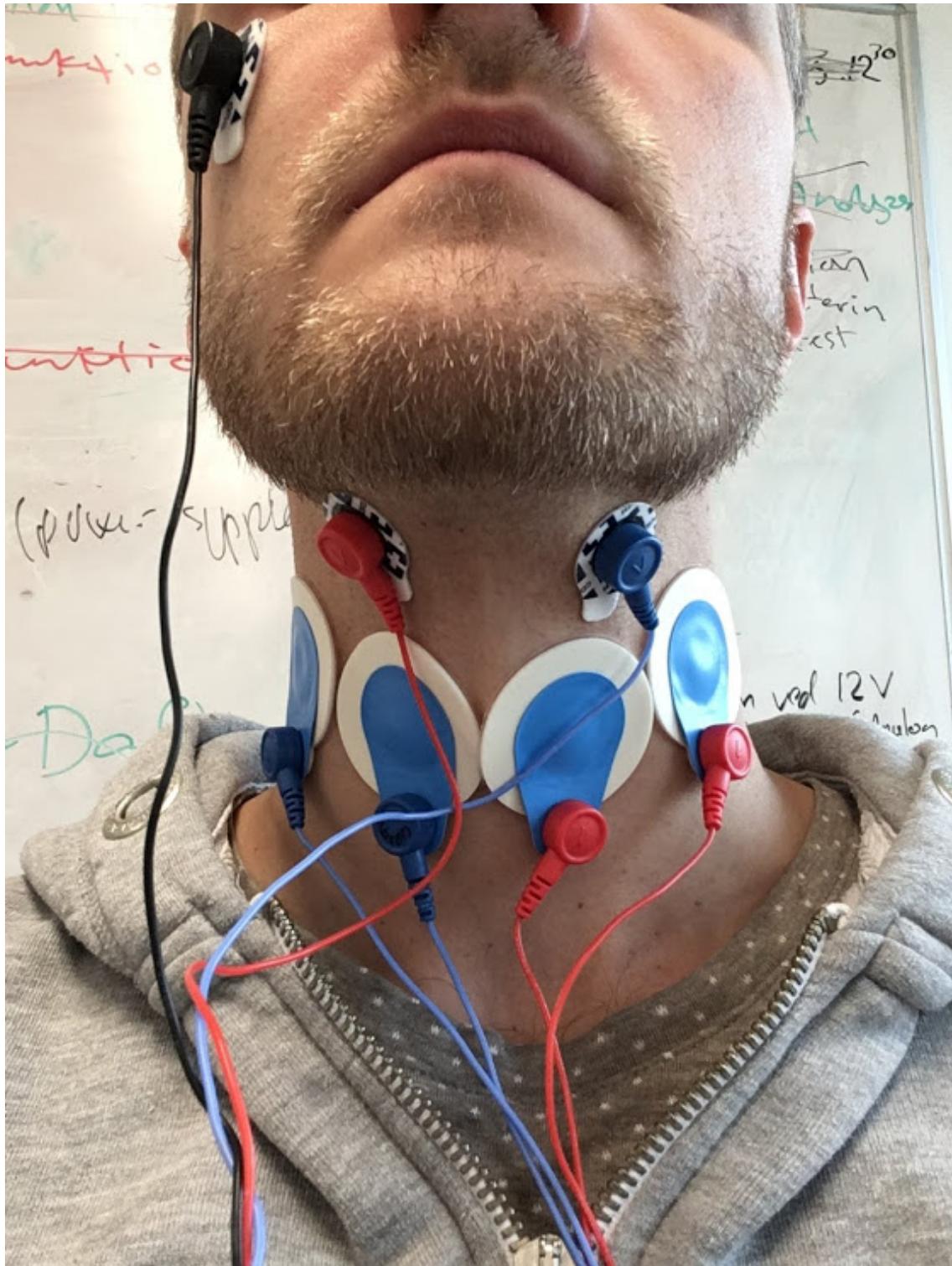
Figur 4.11: Billede af alle komponenter integreret.

4.1. Indledning



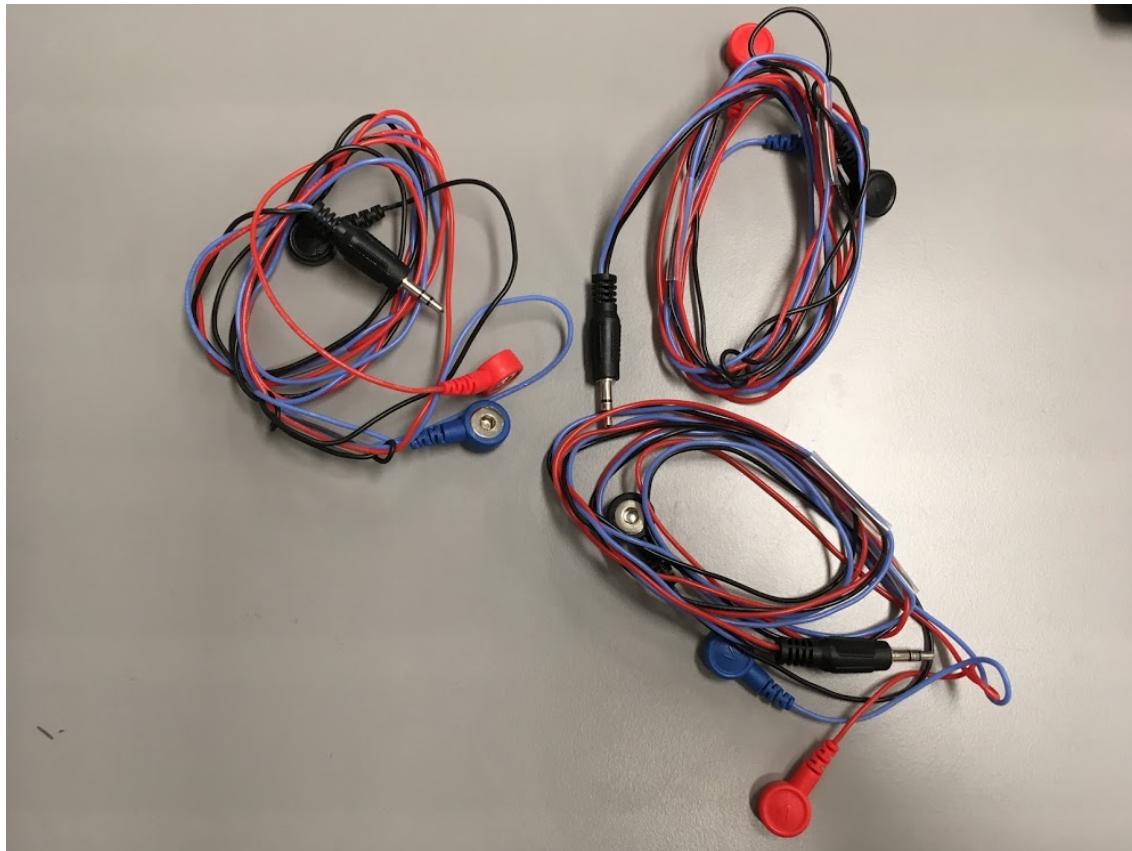
Figur 4.12: Placering af elektroderne uden EMG.

4.1. Indledning



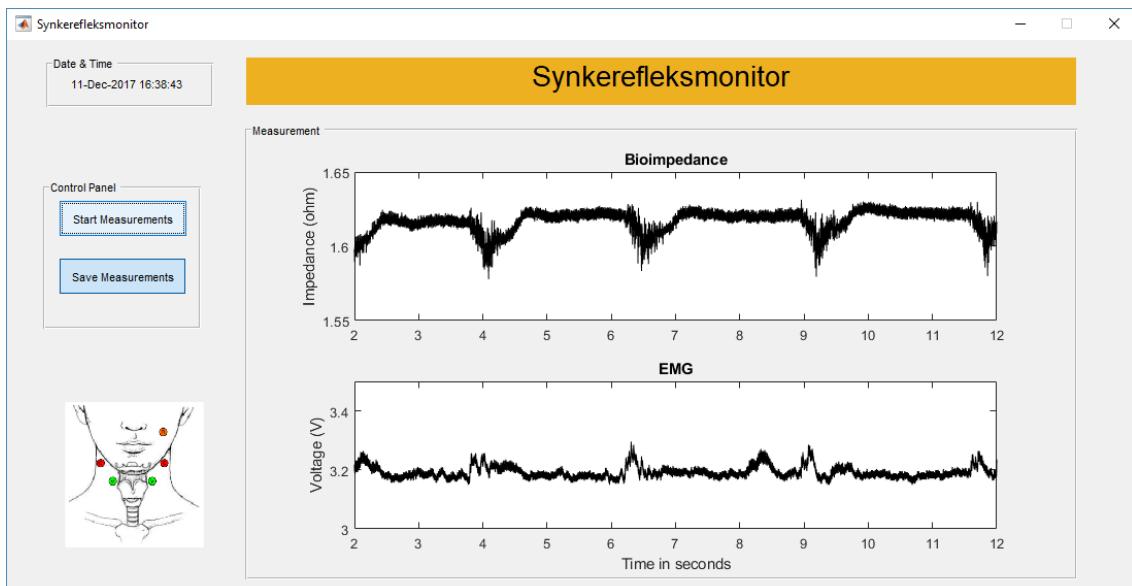
Figur 4.13: Placering af elektroderne med EMG.

4.1. Indledning



Figur 4.14: De brugte elektrodeledninger.

Resultatet af integrationstesten, kan også vises i GUI som viser BI- og EMG-måling simultant på figur 4.15.



Figur 4.15: Simultant måling af BI og EMG signal.

Accepttest 5

5.1 Indledning

Accepttesten skal vise om produktet lever op til de standarder, der er blevet sat op for, at den aktivt kan indgå til klinisk brug. Accepttesten er en opfølgning af kravspecifikation, som har til formål at sikre, at alle kravene er overholdt. Der vil blive testet på hovedscenarier. Det er målsætningen, at disse test sikrer produktets kvalitet, idet produktet vil blive afprøvet før det tages i brug. Derfor er det accepttestens ansvarsfunktion, at godkende de opsatte delmål for produktet, hvad angår både funktionelle, samt ikke-funktionelle krav.

Når der i feltet Godkendt er et flueben, betyder det at testen er godkendt. Hvis der er et flueben i parenteser, betyder det at den er delvis godkendt. Hvis der er et kryds betyder det, at den ikke er godkendt.

5.2 Accepttest af Usecases

5.2.1 Use Case 1

Det forventes for Use Case 1, at sundhedspersonalet har fået tilsluttet BI-måleren, EMG-måleren og påsat elektroder på måleobjektet.

Test af Use Case 1	Start Measurements
Scenarie	Hovedscenarie
Prækondition	Synkerefleksmonitor er monteret korrekt. BI-måleren og EMG-måleren er ledige og operationelle. Elektroderne påsat måleobjektet og GUI-vinduet er åbent

Handling	Forventet observasjon/resultat	Faktisk observasjon/resultat	Godkendt
<i>Hovedscenarie</i>			

5.2. Accepttest af Usecases

1. Sundhedspersonalet trykker på knappen "Start Measurements"	Systemet foretager en måling, hvorefter målingerne vises simultant i graf.	✓
---	--	---

Tabel 5.2: Accepttest af Use Case 1

5.2.2 Use Case 2

Test af Use Case 2	Save Measurements
Scenarie	Hovedscenarie
Prækondition	Use Case 1 er kørt succesfuldt.

Handling	Forventet observasjon/resultat	Faktisk observasjon/resultat	Godkendt
<i>Hovedscenarie</i>			
1. Sundhedspersonalet trykker på knappen "Save Measurements"	Målingerne er gemt i CSV-fil		✓

Tabel 5.4: Accepttest af Use Case 2

5.2.3 Use Case 3

Test af Use Case 3	Load Measurements
Scenarie	Hovedscenarie
Prækondition	Use Case 1 og 2 er kørt succesfuldt

Handling	Forventet observasjon/resultat	Faktisk observasjon/resultat	Godkendt
<i>Hovedscenarie</i>			

5.2. Accepttest af Usecases

1.	Sundhedspersonalet trykker på knappen "Load Measurements"	Systemet henter målinger og viser i graf	Når Forsker indtaster Forsøgsnavn, gør systemet Start Måling-knap tilgængelig	✓
----	---	--	---	---

Tabel 5.6: Accepttest af Use Case 3

5.3. Accepttest af ikke-funktionelle krav

5.3 Accepttest af ikke-funktionelle krav

Krav nr.	Krav	Test	Forventet resultat	Resultat	Godkendt
1.	Sundhedspersonalet skal kunne anvende synkerefleksmonitoren efter 10 minutters instruktion.	Introducér synkerefleksmonitoren	Efter 10 minutter kan Sundhedspersonalet anvende synkerefleksmonitoren		✓
2.	Sundhedspersonalet skal kunne efter endt introduktion til synkerefleksmonitoren foretage en måling uden fejl	Kør Use Case 1	En måling uden fejl		✓
3.	Sundhedspersonalet skal kunne efter en periode, på en uge væk fra synkerefleksmonitoren, foretage en måling uden fejl	Vent en uge og kør Use Case 1	En måling uden fejl		✓

5.3. Accepttest af ikke-funktionelle krav

4.	Sundheds-personalet får mulighed for, at give karakter til GUI-designet på en skala fra 1-5, hvor 5 er yderst tilfredsstillende	Sundheds-personalet giver mundtlig karakter	Forventer 4	✓
5.	Sundheds-personalet skal kunne læse graferne fra GUI'en på 2 meters afstand.	Placér Sundheds-personalet 2 meter væk	Sundheds-personalet fint læse graferne	✓
6.	Det skal maksimalt tage 5 timer at gendanne synkerefleks-monitoren (MTTR - Mean Time To Restore).	Afmontér synkerefleks-monitoren og gendan synkerefleks-monitoren indenfor 5 timer	Systemet er funktionelt indenfor 5 timer	✓
7.	Synkerefleks-monitoren skal have en oppetid uden nedbrud på minimum 1 dag (24 timer) (MTBF - Mean Time Between Failure)	Tilslut Synkerefleks-monitoren og kør Use case 1,2 og 3 hver time i 24 timer	Ingen nedbrud efter 1 dag	✓

5.3. Accepttest af ikke-funktionelle krav

8.	Synkerefleks-monitoren skal have en oppetid/køretid på: <i>Availability =</i> $\frac{MTBF}{MTBF+MTTR} \cdot 100 = \frac{24}{24+5} \cdot 100 = 82,76\%$		✓	
9.	Synkerefleks-monitorens hardware skal kunne tændes indenfor 3 minutter	Tænd Synkerefleks-monituren og tag tid	Synkerefleks-monitorens hardware er tændt på under 3 minutter	✓
10.	Synkerefleks-monitorens GUI skal kunne vises indenfor 3 minutter	Start Synkerefleks-moniturens program og tag tid	Synkerefleks-monitorens GUI er fremme på under 3 minutter	✓
11.	GUI'ens responstid skal maksimum være 30 sekunder	Tryk rundt i GUI og registrer responstid	Responstiden er tilfredsstilende	✓
12.	Synkerefleks-monitorens hardware skal kunne fungere under temperatur mellem -10°C til 40°C	Synkerefleks-moniturens hardware placeres i -10°C og 40°C	Synkerefleks-monitoren fungerer	✓

5.4. Problemrapport

13.	Sundheds-personalet skal kunne udskifte batterierne til hardwaren inden for 2 minutter	Sundheds-personalet fjerner batterierne og i sætter nye mens der tages tid	Batterierne er skiftet under 2 minutter	✓
14.	Sundheds-personalet skal kunne udskifte elektroderne inden for 4 minutter	Sundheds-personalet fjerner elektroderne og påsætter nye mens der tages tid	Elektroderne er skiftet under 4 minutter	✓
15.	Softwaren skal opbygges med lav samhørighed			✓

Tabel 5.7: Accepttest af Ikke-funktionelle krav

5.4 Problemrapport

Her noteres misvisninger i forhold til forventet resultat, både fra usecases og ikke-funktionelle krav.

[2]

Litteratur

- [1] Christopher J Chester. Electrical-Impedance Biofeedback Instrument for Swallowing Rehabilitation. 2014.
- [2] Kevin R. Aroom, Matthew T. Harting, Charles S. Cox, Ravi S. Radharkrishnan, Carter Smith, and Brijesh S. Gill. Bioimpedance Analysis: A Guide to Simple Design and Implementation. *Journal of Surgical Research*, 153(1):23–30, 2009.