

# E4DSA

## Case projekt 4 – pulsoximetri

Team 8

9. maj 2018

### Indhold

<b>1</b>	<b>Introduktion</b>	<b>2</b>
<b>2</b>	<b>Pulsoximetri</b>	<b>2</b>
<b>3</b>	<b>Data</b>	<b>3</b>
<b>4</b>	<b>Analyse</b>	<b>3</b>
4.1	1. midlingsfilter . . . . .	4
4.2	Afgrænsning af pulsvariation . . . . .	4
4.2.1	Udregning af iltprocent . . . . .	6
4.3	2. midlingsfilter . . . . .	6
4.3.1	Udregning af puls . . . . .	7
<b>5</b>	<b>Konklusion</b>	<b>8</b>
<b>A</b>	<b>Matlab-kode</b>	<b>9</b>

# 1 Introduktion

Denne rapport omhandler anvendelsen af digital signalanalyse på data indhentet fra måling af rødt og infrarødt lys efter det har gennemlyst en fingerspids.

Der anvendes i opgaven midlingsfiltre for at rense- og analysere data.

## 2 Pulsoximetri

Et puls-oximeter (se figur 1) består overordnet set af en IR-transmitter og modtager som sidder på hver side af en fingerspids. Røde blodlegemer med ilt i vores blod optager infrarødt lys. Derfor kan man måle iltprocenten i blodet ved at måle hvor meget IR lys der kommer gennem fingerspidsen. Ilten i blodet falder og stiger for hvert puls slag. Derfor kan man også måle pulsfrekvensen ved at registrere frekvensen af fald og stigninger i iltprocenten.



Figur 1: Moderne pulsoximeter

Typisk anvendes en infrarød pære der udsender lys ved 940 nm og en rød pære der udsender lys ved 660 nm. Ratioen mellem disse kan anvendes til beregningen i ændringen i absorption af infrarødt lys:

$$R = \frac{\frac{V_{max}R - V_{min}R}{V_{min}R}}{\frac{V_{max}IR - V_{min}IR}{V_{min}IR}} \quad (1)$$

hvor  $V_{max/min}R/IR$  angiver spændingsmaksimum og -minimum for rød og infrarød pære.

Iltprocenten udregnes som

$$SpO_2 = (10,0002 \cdot R^3) - (52,887 \cdot R^2) + (26,871 \cdot R) + 98,283 \quad (2)$$

Forsøget har bygget på en rød og infrarød pære som har gennemlyst en finger og der er målt på rød og infrarød phototransistorer på den modsatte side af fingeren. Som tjek af vores resultater er der samtidig, på samme forsøgsperson men på en anden finger, målt iltprocent og pulsfrekvens på samme tidspunkt som dataopsamlingen. Iltprocenten blev målt til 98% og pulsfrekvensen blev målt til 77.

### 3 Data

Data består af en .csv-fil<sup>2</sup>. Data er udgør spændingssignalet fra en infrarødphototransistor samt fra en rødphototransistor som har siddet på den ene side af en finger, samtidig med at en rød og en infrarød pære har gennemlyst fra den anden side af fingeren.

Data er optaget ved 800 Hz og der er opsamlet 8192 samples, svarende til ca. 10 sekunder.

Data præpareres:

```
1  sig1=csvread('data/Simon.csv');
2
3  time_S=sig1(:, 1);
4  RED_S=sig1(:, 2);
5  IR_S= sig1(:, 3);
6
7  % udregning af variable
8  Ts = time_S(2)-time_S(1);
9  fs = 1/Ts; %800 hertz
10 N = 8192;
11 Tdur = N/fs;
12 n = (1:N);
```

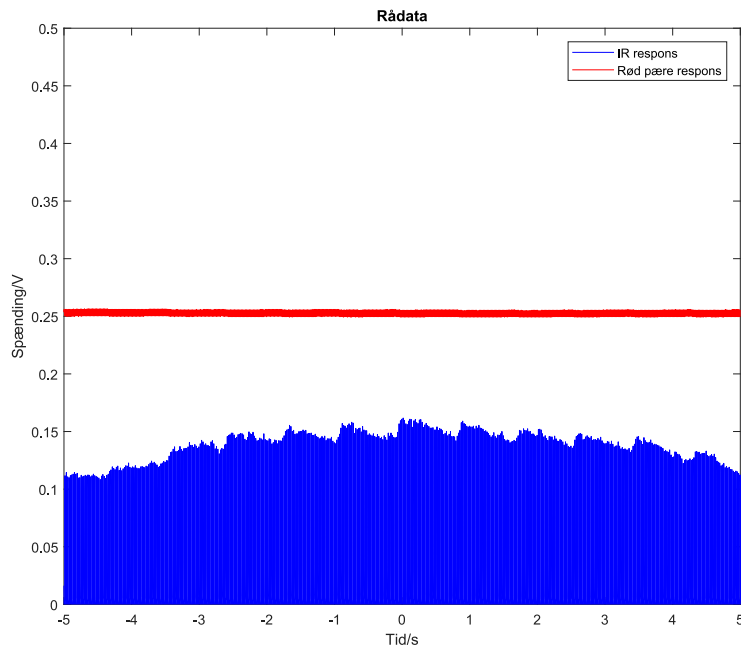
### 4 Analyse

I figur 2 ses plot af rådata. Linjen med meget svag spændingsvariation er data fra den røde phototransistor.

---

<sup>1</sup>[https://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/f2012/prd47/PulseOximeter/Pulse\\_ox.html](https://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/f2012/prd47/PulseOximeter/Pulse_ox.html)

<sup>2</sup><https://github.com/henriktrue/>



Figur 2: Rådata

De store variationer er data fra den infrarøde phototransistor. Det ses at der forekommer cykliske variationer i amplituden af udsvingene fra den infrarøde phototransistor. Disse antages at forekomme pga. variationen i iltmætning i blodet som konsekvens af pulsen. Det ses også at data fremstår meget støjfyldt. For at beregne iltmætning i blodet og pulsfrekvensen anvendes et midlingsfilter som oprensning af data.

#### 4.1 1. midlingsfilter

```

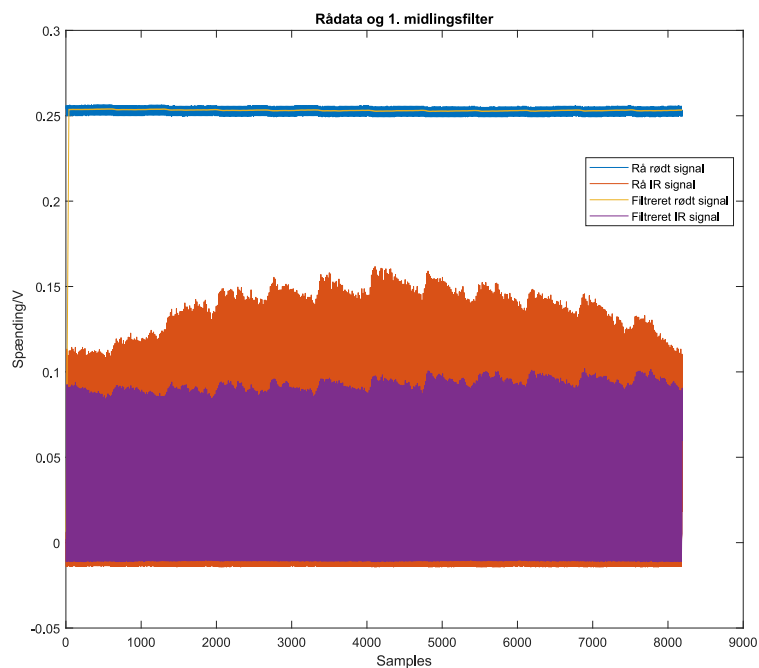
1  % rødt signal
2  M_S_R = 40; % filterkoefficienter
3  hMA_R_S = 1/M_S_R*ones(1,M_S_R); % MA-filter , filterkoefficienter
4  yMA_R_S = filter(hMA_R_S,1,RED_S); % filtrerer inputsignal
5
6  % IR signal
7  M_S_IR = 3; % filterkoefficienter
8  hMA_IR_S = 1/M_S_IR*ones(1,M_S_IR); % MA-filter , filterkoefficienter
9  yMA_IR_S = filter(hMA_IR_S,1,IR_S); % filtrerer inputsignal

```

I figur 3 ses rådata med data efter filtrering. Både rødt og infrarødt data fremstår mere ensartet.

#### 4.2 Afgrænsning af pulsvariation

For at afgrænse pulseringen analyseres data i sektioner af 16 samples hvor den højeste værdi registreres. Denne værdi erstatter så samtlige 16 værdier i sektionen. Koden er følgende:



Figur 3: Rådata

```

1  c = zeros(1,N);
2
3  for i = 1:512
4  c((i*16-15):(16*i)) = ones(1,16) * max(yMA_IR_S(16*i-15:16*i));
5  end

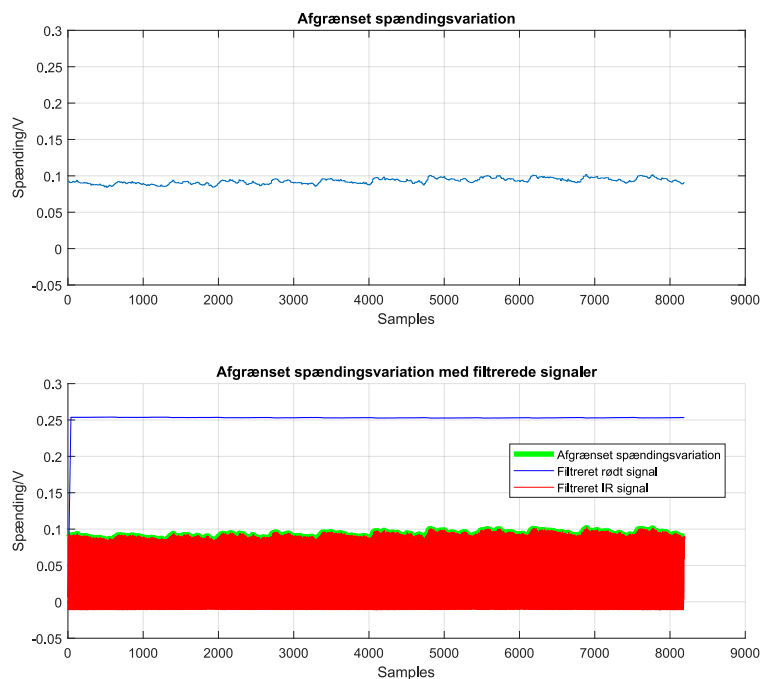
```

Følgende værdier aflæses:

Spænding	Matlab-kode	Værdi
$V_{min}R$	<code>min(yMA_R_S(41:length(yMA_R_S)))</code>	0,2524 V
$V_{max}R$	<code>max(yMA_R_S(41:length(yMA_R_S)))</code>	0,2541 V
$V_{min}IR$	<code>min(c)</code>	0,0843 V
$V_{max}IR$	<code>max(c)</code>	0,1019 V

Tabel 1: Spændingsminima og -maxima

I følgende subplots plottes først spændingsvariationen efter afgrænsning og under dette er afgrænsningen plottet oveni den midlingsfiltrede signal.



Figur 4: Afgrænsning af spændingsvariation

#### 4.2.1 Udregning af iltprocent

Med værdierne i tabel 1 kan ratioen ved hjælp af ligning 1 udregnes til:

$$R = \frac{\frac{V_{max}R - V_{min}R}{V_{min}R}}{\frac{V_{max}IR - V_{min}IR}{V_{min}IR}} \Rightarrow \frac{\frac{0,2541V - 0,2524V}{0,2524V}}{\frac{0,1019V - 0,0843V}{0,0843V}} = 0,0323 \quad (3)$$

Iltprocenten udregnes som

$$SpO_2 = (10,0002 \cdot R^3) - (52,887 \cdot R^2) + (26,871 \cdot R) + 98,283 = 99,1\% \quad (4)$$

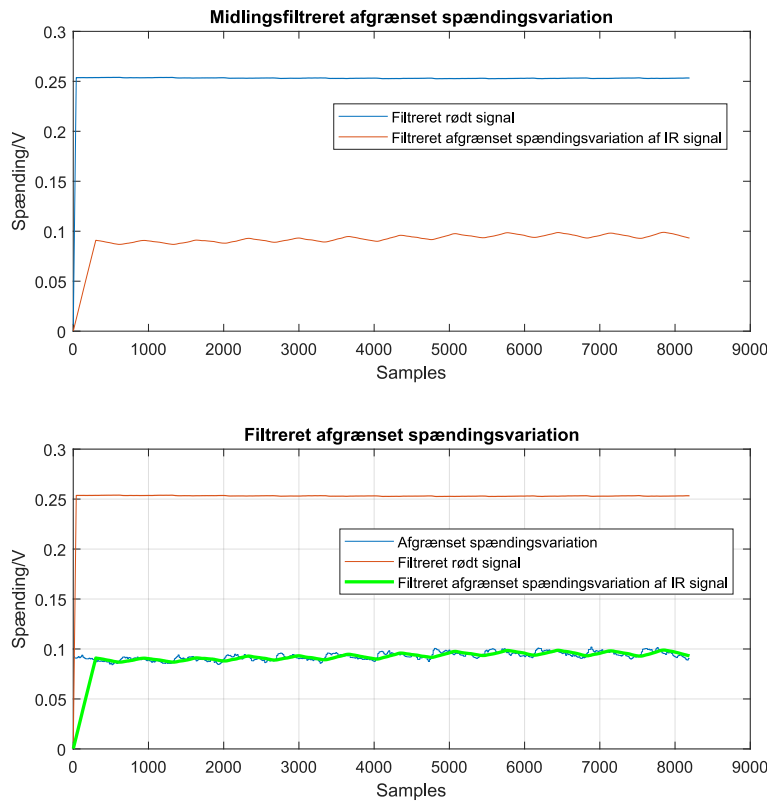
Iltprocenten findes altså at være ca. 99%.

### 4.3 2. midlingsfilter

For at udregne pulsfrekvensen påføres den afgrænsede spændingsvariation et midlingsfilter. Dette bevirker at pulsudsvingene bliver mere udjævnet:

```
1 % IR lampesignal
2 M_S_IR_filtreret = 300; % filterkoefficienter
3 hMA_IR_S_filtreret = 1/M_S_IR_filtreret*ones(1,M_S_IR_filtreret); % MA-filter , filterkoefficienter
4 yMA_IR_S_filtreret = filter(hMA_IR_S_filtreret,1,c); % filtrerer inputsignal for Simons del
```

Kurven er plottet herunder for sig selv og sammen med den afgrænsede spændingsvariation:



Figur 5: Midlingsfilter påført det afgrænsede spændingsvariationssignal

#### 4.3.1 Udregning af puls

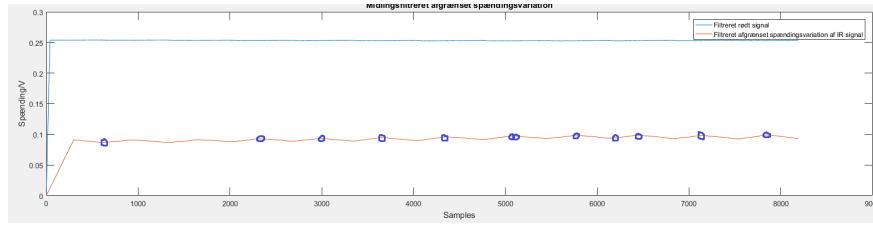
For at udregne pulsen udregnes først antallet af højderygge i det midlede signal samt samplennummeret for hver højderyg. Hver højderyg repræsenterer et pulsslæg. Nedenfor er koden vist og `puls` angiver det samlede antal af pulsslæg og `p_x` angiver samplennummeret ved en højderyg:

```

1  puls = 0;
2  p_x = [];
3  for i = (M_S_IR_filtreret/8+1):(N/8)
4      if (yMA_IR_S_filtreret(i*8) >= yMA_IR_S_filtreret((i*8)-1)) && (yMA_IR_S_filtreret(i*8) >=
        yMA_IR_S_filtreret((i*8)+1))
5          puls = puls + 1;
6          p_x = ([p_x, i*8]);
7      end
8  end

```

Det ses at antallet af højderygge er 12. Dog ses det ved undersøgelse af vektoren  $p_x$  består af følgende faktorer:  $p_x = [620, 2332, 3004, 3660, 4348, 5068, 5084, 5772, 6140, 6444, 7448, 7852]$ . Det ses desværre at højderygge ikke falder i regelmæssige intervaller:



Figur 6: Højderygge markeret

Den første højderyg er hvorfra filteret er aktivt. Tiden,  $\Delta t$ , mellem den anden højderyg (den første højderyg som følge af et pulsslag),  $p_x(2)$ , og sidste højderyg,  $p_x(12)$  er:

$$\Delta t = \frac{p_x(12) - p_x(1)}{800Hz} = 9,04s \quad (5)$$

Pulsfrekvensen,  $f_{puls}$ , kan udregnes som

$$f_{puls} = \frac{60}{\Delta t} \cdot 12 = 79,4 \text{ slag pr. minut} \quad (6)$$

Dette svarer til følgende matlab-kode:

```
1    pulsfr = (60 / ((p_x(length(p_x)) - p_x(1)) / 800)) * puls;
```

Pulsfrekvensen findes altså at være 79.

## 5 Konklusion

Rapporten har vist hvordan digitalsignal analyse med bl.a. midlingsfiltrering kan udregne pulsfrekvens og iltprocent fra målinger af gennemlysning af rødt og infrarødt lys i en finger. Dette er blevet sammenholdt med målinger med et konventionelt pulsoximeter. Der er vist en pæn overensstemmelse og samtidig nytten af dataoprensning med midlingsfiltre.



## A Matlab-kode

```
1 %% Målinger DSE opgave
2 % Henrik Truelssen, Viggo Lysdahl, Simon Mylius 03/05-2018
3
4 %% Generelt setup:
5 clear; close all; clc; format compact
6
7 %% Indlæsning af data til step respons
8 sig1=csvread('data/data.csv');
9
10 time_S=sig1(:, 1);
11 RED_S=sig1(:, 2);
12 IR_S= sig1(:, 3);
13 % udregning af variable
14 Ts = time_S(2)-time_S(1);
15 fs = 1/Ts; %800 hertz
16 N = 8192;
17 Tdur = N/fs;
18 n = (1:N);
19
20
21 %% 1. midlingsfilter
22 % rødt signal
23 M_S_R = 40; % filterkoefficienter
24 hMA_R_S = 1/M_S_R*ones(1,M_S_R); % MA-filter, filterkoefficienter
25 yMA_R_S = filter(hMA_R_S,1,RED_S); % filtrerer inputsignal
26
27 % IR signal
28 M_S_IR = 3; % filterkoefficienter
29 hMA_IR_S = 1/M_S_IR*ones(1,M_S_IR); % MA-filter, filterkoefficienter
30 yMA_IR_S = filter(hMA_IR_S,1,IR_S); % filtrerer inputsignal
31
32 c = zeros(1,N);
33
34 for i = 1:512
35     c((i*16-15):(16*i)) = ones(1,16) * max(yMA_IR_S(16*i-15:16*i));
36 end
37
38 % IR signal
39 M_S_IR_filtreret = 300; % filterkoefficienter
40 hMA_IR_S_filtreret = 1/M_S_IR_filtreret*ones(1,M_S_IR_filtreret); % MA-filter, filterkoefficienter
41 yMA_IR_S_filtreret = filter(hMA_IR_S_filtreret,1,c); % filtrerer inputsignal
42
43 puls = 0;
44 p_x = [];
45 for i = (M_S_IR_filtreret/8+1):(N/8)
46     if (yMA_IR_S_filtreret(i*8) >= yMA_IR_S_filtreret((i*8)-1)) && (yMA_IR_S_filtreret(i*8) >=
47         yMA_IR_S_filtreret((i*8)+1))
48         puls = puls + 1;
49         p_x = ([p_x,i*8]);
50     end
51 end
52 pulsfr=(60/((p_x(length(p_x))-p_x(2))/800))*puls;
53
54
55
56
57 %% plotning af signaler
58
59 figure
60 plot(time_S, IR_S, 'b'), grid, hold on
61 plot(time_S, RED_S, 'r'), grid;
62 xlabel('Tid/s'), ylabel('Spænding/V'), title('Rådata');
63 xlim([-5 5]), ylim([0 0.5]);
64 legend('IR respons', 'Rød pære respons');
65
```

```

66 figure
67 plot(n,RED_S), grid, hold on
68 plot(n,IR_S), grid, hold on
69 plot(n,yMA_R_S), grid, hold on
70 plot(n,yMA_IR_S), grid, hold on
71 xlabel('Samples'), ylabel('Spænding/V'), title('Rådata og 1. midlingsfilter');
72 legend('Rå rødt signal','Rå IR signal', ['Filtreret rødt ' ...
73         'signal'],'Filtreret IR signal');
74
75 figure
76 subplot(2,1,1)
77 plot(n,c), grid, hold on
78 ylim([-0.05 0.3]);
79 xlabel('Samples'), ylabel('Spænding/V'), title('Afgrænset spændingsvariation');
80
81 subplot(2,1,2)
82 hold on
83 plot(n,c, 'g','LineWidth', 4), grid, hold on
84 plot(n,yMA_R_S,'b'), grid, hold on
85 plot(n,yMA_IR_S,'r'), grid, hold on
86 xlabel('Samples'), ylabel('Spænding/V'), title(['Afgrænset ' ...
87         'spændingsvariation med filtrerede signaler']);
88 legend('Afgrænset spændingsvariation','Filtreret rødt signal','Filtreret IR signal');
89
90
91
92 figure
93 subplot(2,1,1)
94 plot(n,yMA_R_S), grid, hold on
95 plot(n,yMA_IR_S_filtreret),grid,hold on;
96 xlabel('Samples'), ylabel('Spænding/V'), title(['Midlingsfiltreret afgrænset ' ...
97         'spændingsvariation']);
98 legend('Filtreret rødt signal','Filtreret afgrænset spændingsvariation af IR signal');
99
100 subplot(2,1,2)
101 plot(n,c), grid, hold on
102 plot(n,yMA_R_S), grid, hold on
103 plot(n,yMA_IR_S_filtreret, 'g','LineWidth', 2),grid,hold on;
104 xlabel('Samples'), ylabel('Spænding/V'), title(['Filtreret afgrænset ' ...
105         'spændingsvariation']);
106 legend('Afgrænset spændingsvariation',['Filtreret rødt signal'],['Filtreret ' ...
107         'afgrænset spændingsvariation af IR signal']);

```