Oblig 2

Krister Borge
Hamza Muftic
Barthas Venkcus

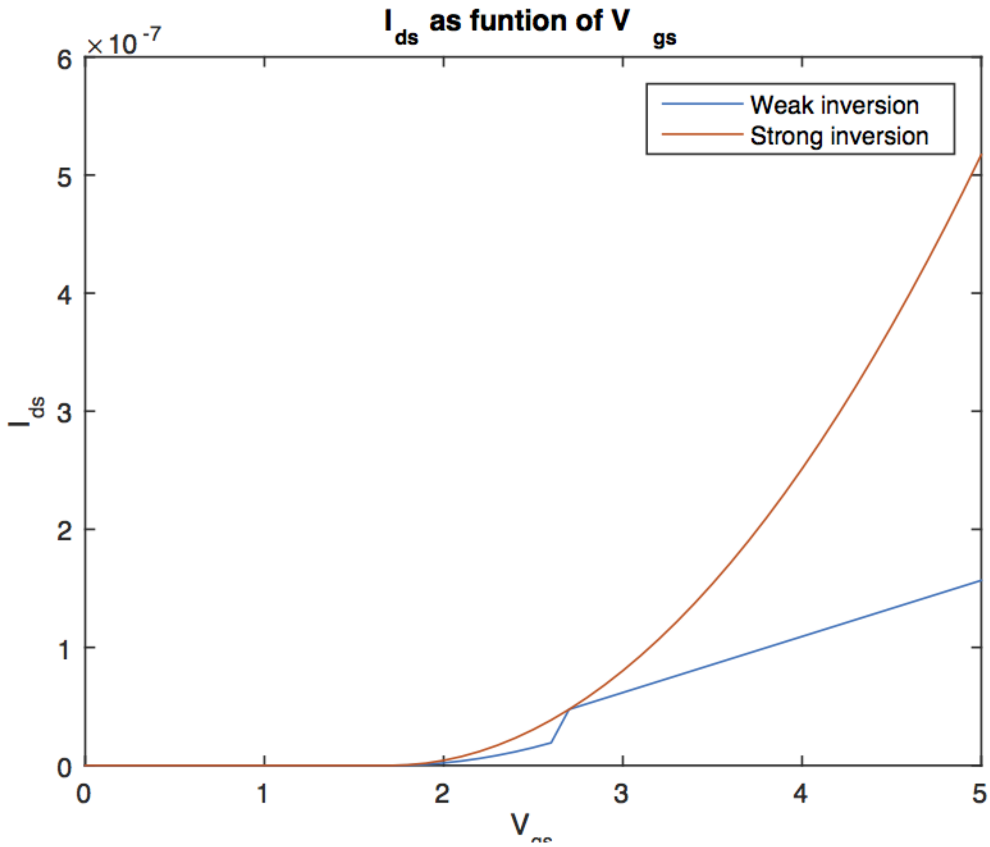
7. oktober 2015

Placement of images is hard in latex, therefor the images is on the last page.

0.1 Task1

Both a $Vds > vt$ for strong inversion and a $Vds < vt$ for weak inversion.

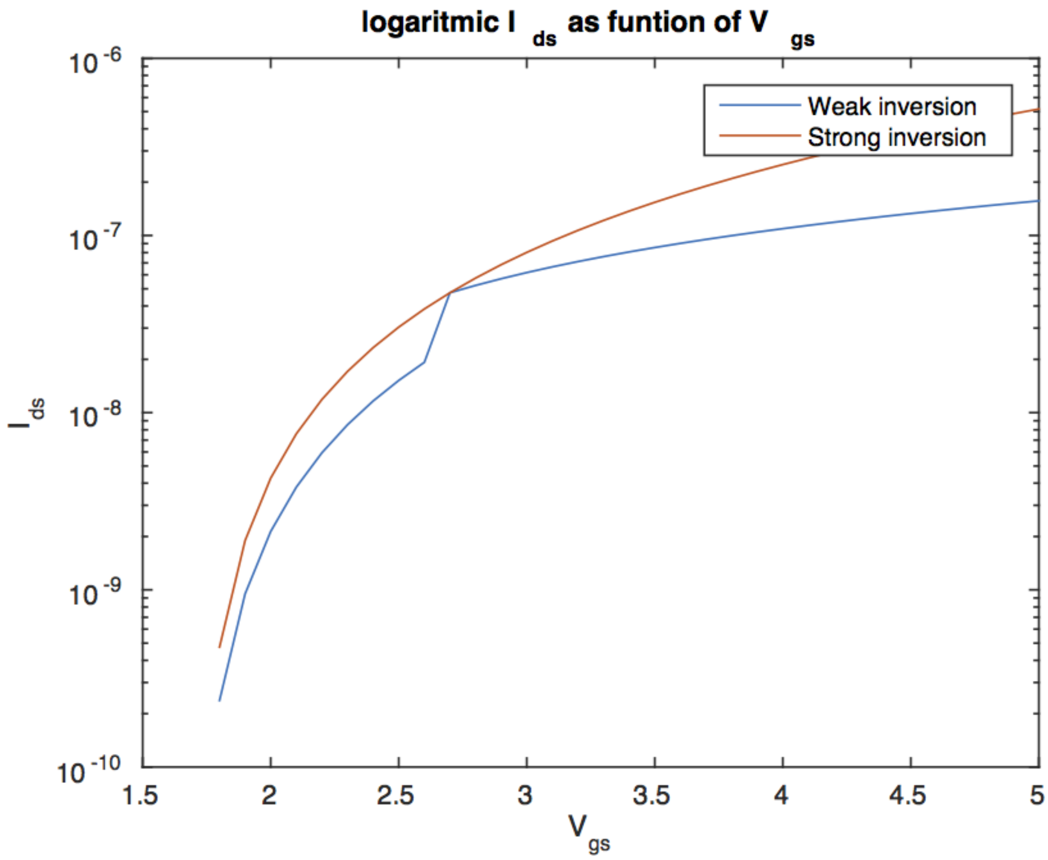
Figur 1: Task 1



logaritmic scale

```
Vgs=linspace(0, 5, 51)';
sIds=zeros(length(Vgs),1);
wIds=zeros(length(Vgs),1);
for n= 1:length(Vgs);
    sIds(n)=nmosmodel(Vgs(n),1.7,Vgs(n)-1.7); %strong inversion
    wIds(n)=nmosmodel(Vgs(n),1.7,1); %weak inversion
end
figure();


plot (Vgs,wIds)
hold on
plot (Vgs,sIds)
legend('Weak inversion', 'Strong inversion')
title('I_{ds} as funtion of V_{gs}')
xlabel('V_{gs}')
ylabel('I_{ds}')
xlabel('V_{gs}')
ylabel('I_{ds}')
```

Figur 2: Task 1 logaritmic




```
hold off
figure()

semilogy(Vgs,wIds)
hold on
semilogy(Vgs,sIds)
legend('Weak inversion', 'Strong inversion')
title('logaritmic I_{ds} as funtion of V_{gs}')
xlabel('V_{gs}')
ylabel('I_{ds}')

xlabel('V_{gs}')
ylabel('I_{ds}')
```

nMos model:

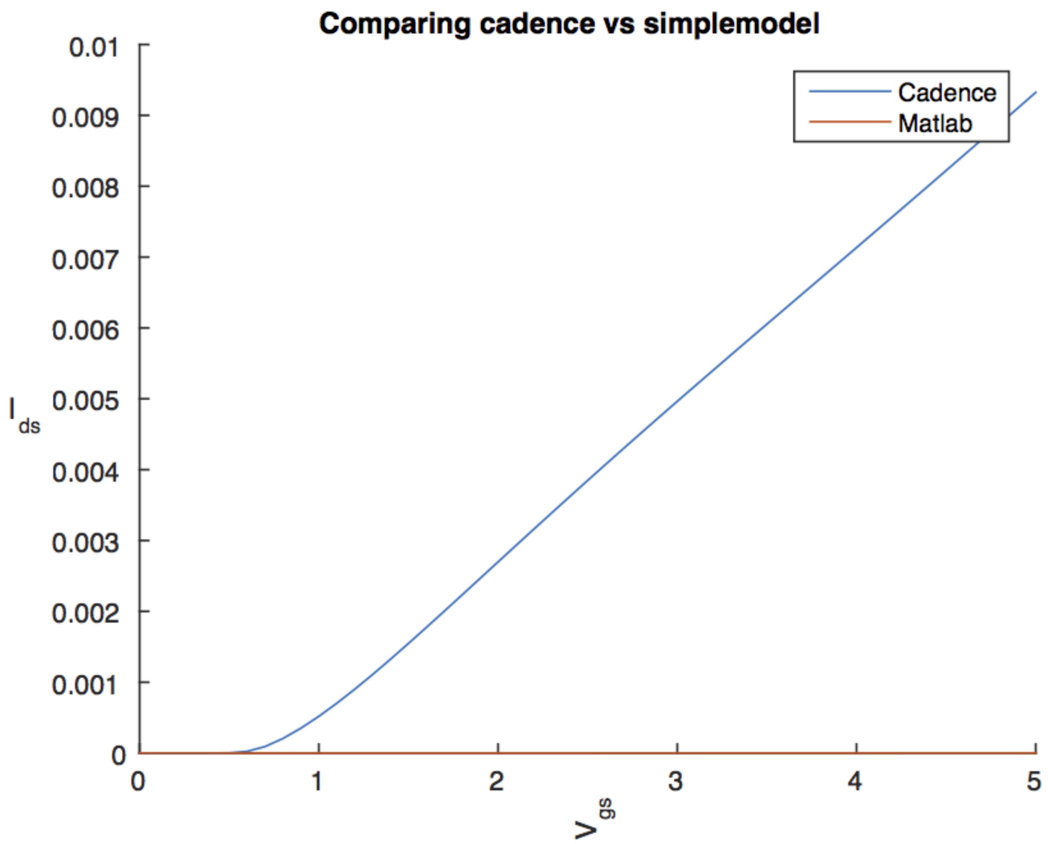
```
function [ Ids ] = nmosmodel(Vgs,vt,Vds)
%nmos-model plotting vds as function of Vgs
%   W/L=10um/0.4um
%   uCoxW/Lm=beta =190*10/0.4
%   Vtn=0.57
%   Cox=4.5
%   0.35um prosess
%   V_T=kT/q=26mV at 300 degree Kelvin
%   n=1.5 for weak inversion
%   n=1.7 for strong ionversion
```

```
%   no length modulation lambda

W=10;
L=0.4;
bolt=1.38e-23;
beta=190*W/L;
Veff=Vgs-vt;

if Vgs<vt
    Ids=0;
%triode region
else if (Vgs>vt && Vds<=Veff);
    vt=vt;
    Ids=(beta*(Veff)*Vds-(Vds^2/2))*10e-12;
%active region
else % (Vgs>vt && Vds>=Veff)
    Ids=(0.5*beta*Veff^2)*10e-12;
end
end
```

0.2 Task 2

Comparing cadence with our simple model

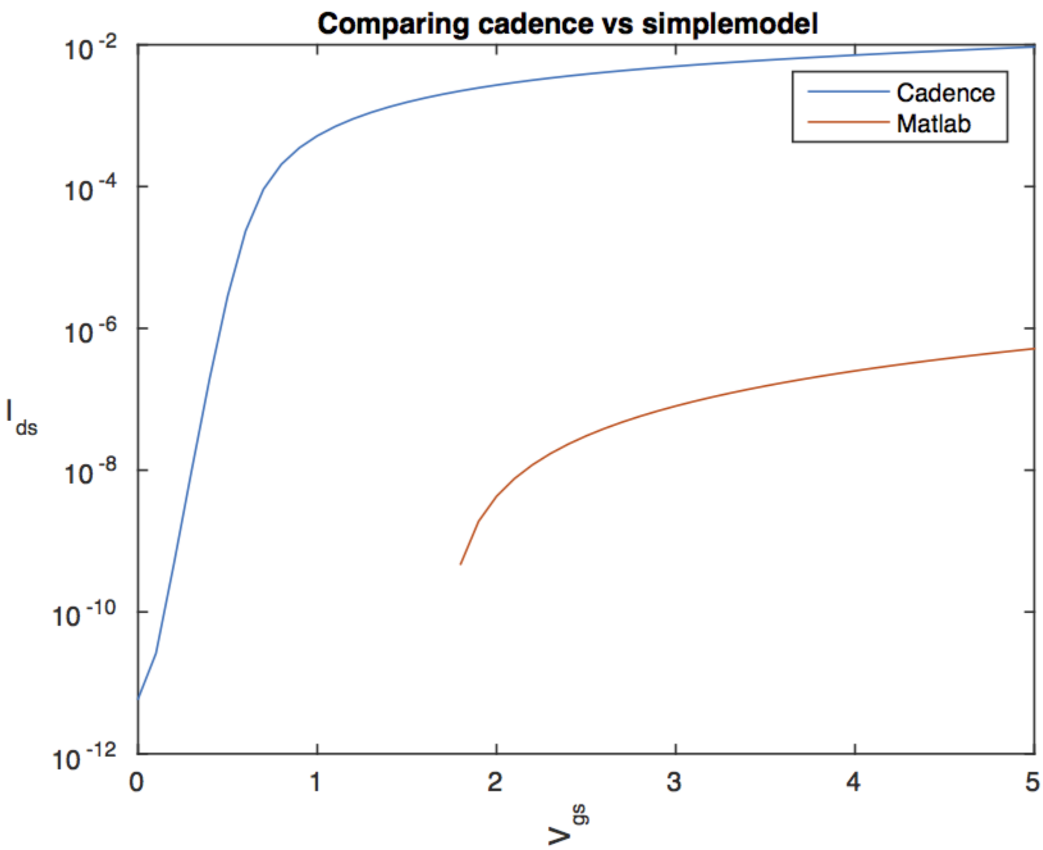
Figur 3: Task 2



The script with changed W anf L parameters:

```
function [ Ids ] = nmosmodel(Vgs,vt,Vds)
```

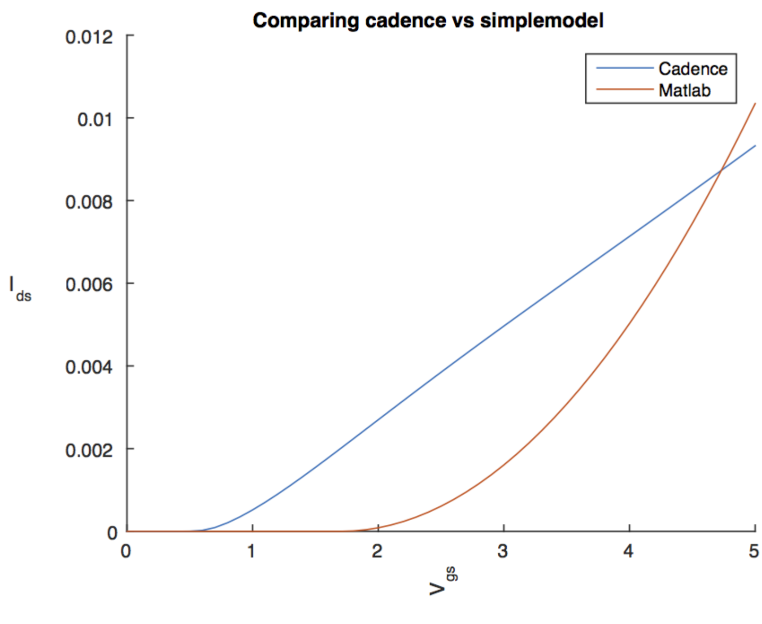
Figur 4: task 2 log



```
%nmos-model plotting vds as function of Vgs
%    W/L=10um/0.4um
%    uCoxW/Lm=beta =190*10/0.4
%    Vtn=0.57
%    Cox=4.5

%    0.35um prosess
%    V_T=kT/q=26mV at 300 degree Kelvin
%    n=1.5 for weak inversion
%    n=1.7 for strong ionversion
%    no length modulation lambda


W=100;
L=0.0002;
bolt=1.38e-23;
beta=190*W/L;
Veff=Vgs-vt;

if Vgs<vt
    Ids=0;
%triode region
else if (Vgs>vt && Vds<=Veff);
    vt=vt;
    Ids=(beta*(Veff)*Vds-(Vds^2/2))*10e-12;
%active region
else % (Vgs>vt && Vds>=Veff)
    Ids=(0.5*beta*Veff^2)*10e-12;
end
end
```

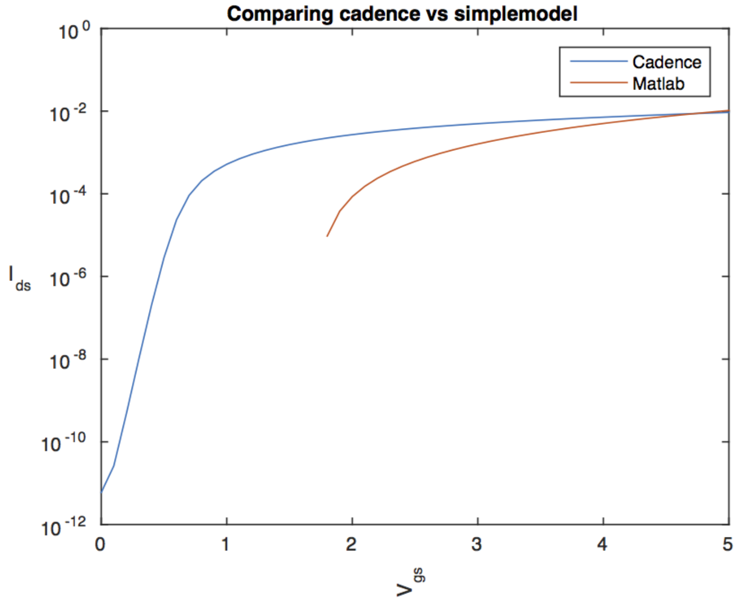
Matched plots: bla bla

Figur 5: task 2 modified W/L



this works nicely

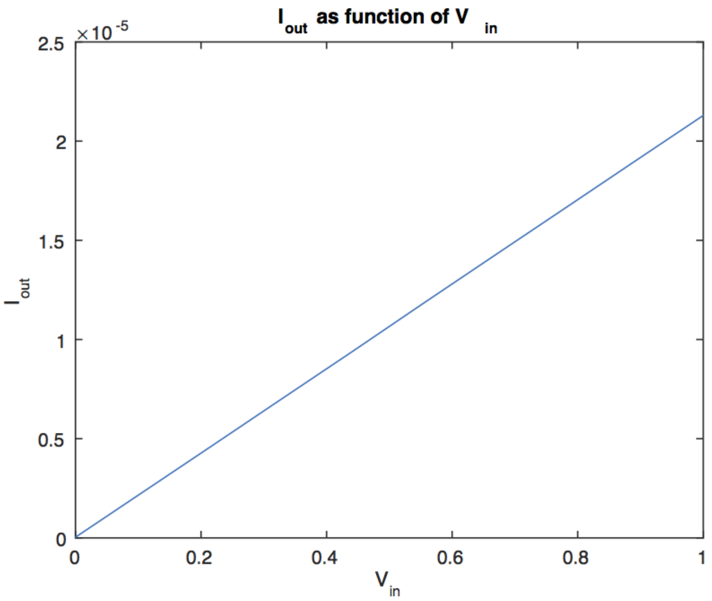
Figur 6: task 2 log modified W/L



0.3 Task 3

47Ω resistor This shows the linear operation of a resistor

Figur 7: task 3



0.4 task 4

```
function [ iout ] = gpib_function4( vinmin,vinmax, steps )

HPE3631_Init;
HPE3631_SetILimit(1,0.01);
HPE3631_SetILimit(2,0.01);

HPE3631_Operate;
```

```
HPE3631_SetVolt(1,vinmax)  %setting drain voltage

K617_Init;
iout=zeros(1,steps)';
K617_SetRange(0);
K617_SetMode('A');

i=linspace(vinmin,vinmax,steps);
for k=1:length(i)
   HPE3631_SetVolt(2,i(k));
   pause(0.5);
   iout(k) = K617_ReadQuick();
end

end
```

and a logaritmic y-axis:
Seting up GPIB for measuring MC1400 transistor package:

0.5 Task 5

Measuring setup for nMOS and pMOS
See figures

0.6 Task 6

The same prosedure for pMOS as in 4
see figures

0.7 Task 7

By just observing the diferences of the plots for pMOS and nMOS, it is easy to see that I_{ds} is a function of V_{gs}. In the case of pMOS I_{ds} is low on a high V_{gs}., For nMOS I_{ds} is low on a low V_{gs}. μ mobility and β is lower for a pMOS than a nMOS. This means that a pMOS is a generally weaker amplifier than nMOS.

0.8 Task 8

see figures

0.9 Task 9

see figures

0.10 Task 10

0.11 Task 11

The first try we didn't manage to calculate beta correctly. All betas we got were complex numbers. We tried ageain. this is our results:

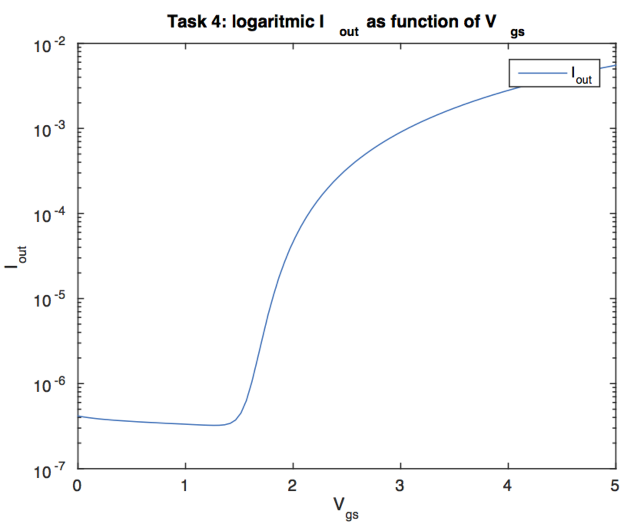
```
 Oppgave11betaVerdi =

    0.0145

function [ OppgavebetaVerdi] = lol(Iout)
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here

Vgsn = linspace(0, 5,100);


 semilogy(Vgsn, sqrt(Iout));
 syms x s;

 x = Vgsn(26);
 s = Iout(26);

Oppgave11betaVerdi = (sqrt(Iout(50))-sqrt(Iout(26)))/(Vgsn(50)-Vgsn(26))

 xlabel('Vgs');
 ylabel('Ids');



end
```

0.12 Task 12

0.13 Figures

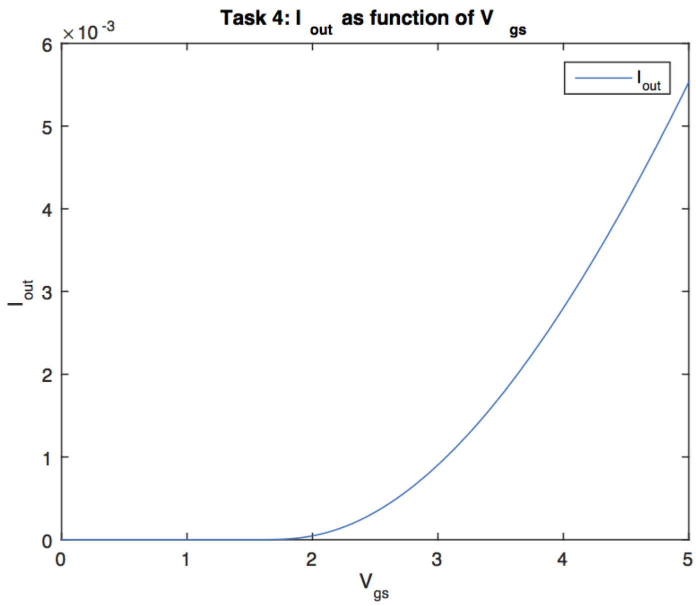
Figures

Figur 8: **task 4** log

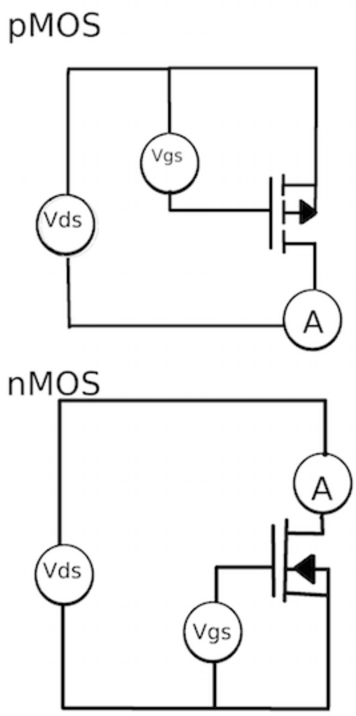


and logaritmic

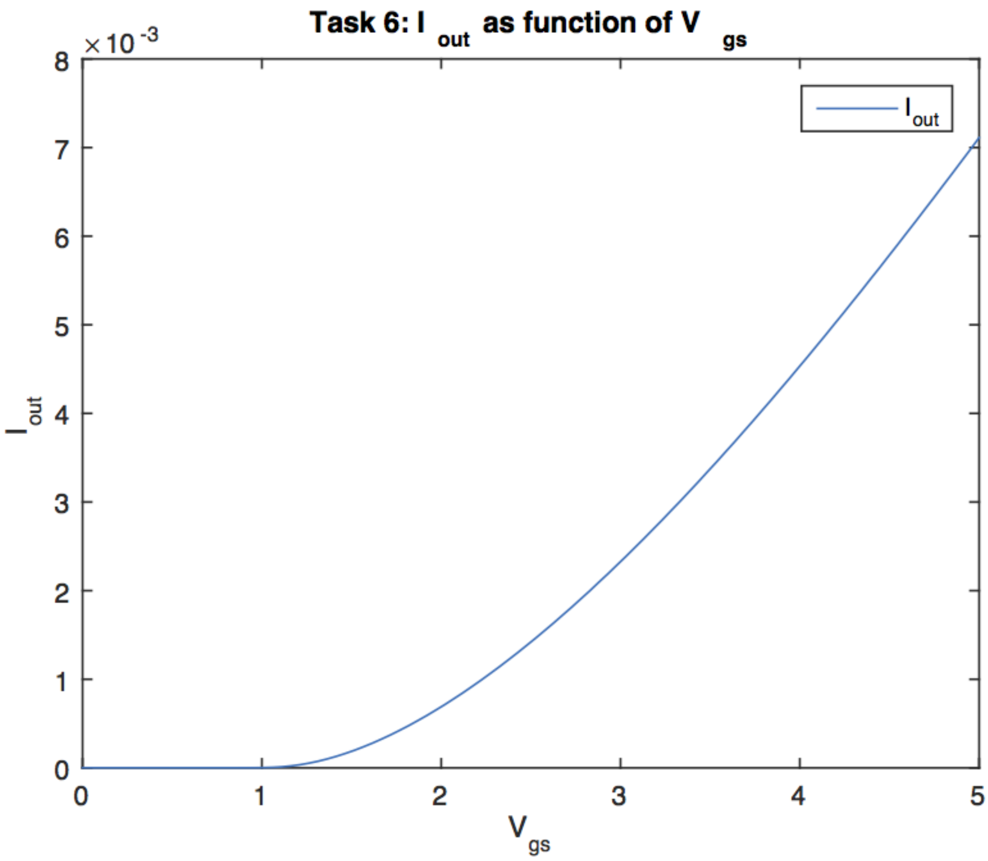
Figur 9: **task 4**



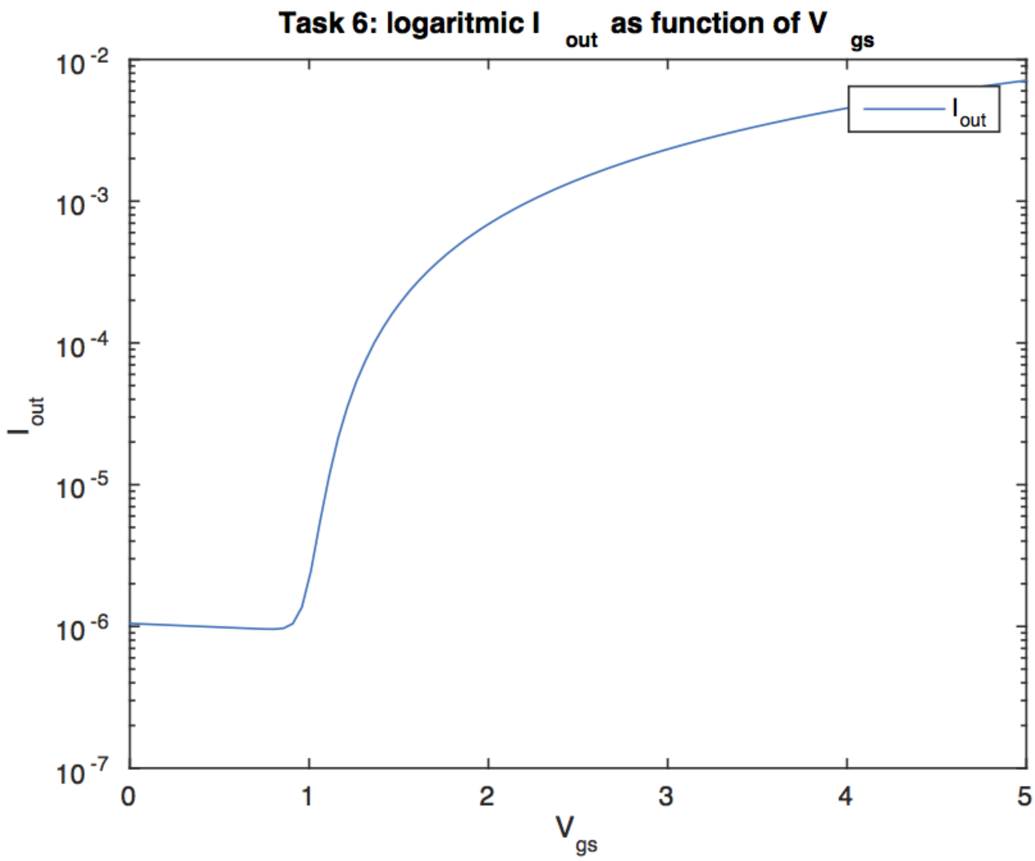
Figur 10: **task 5**



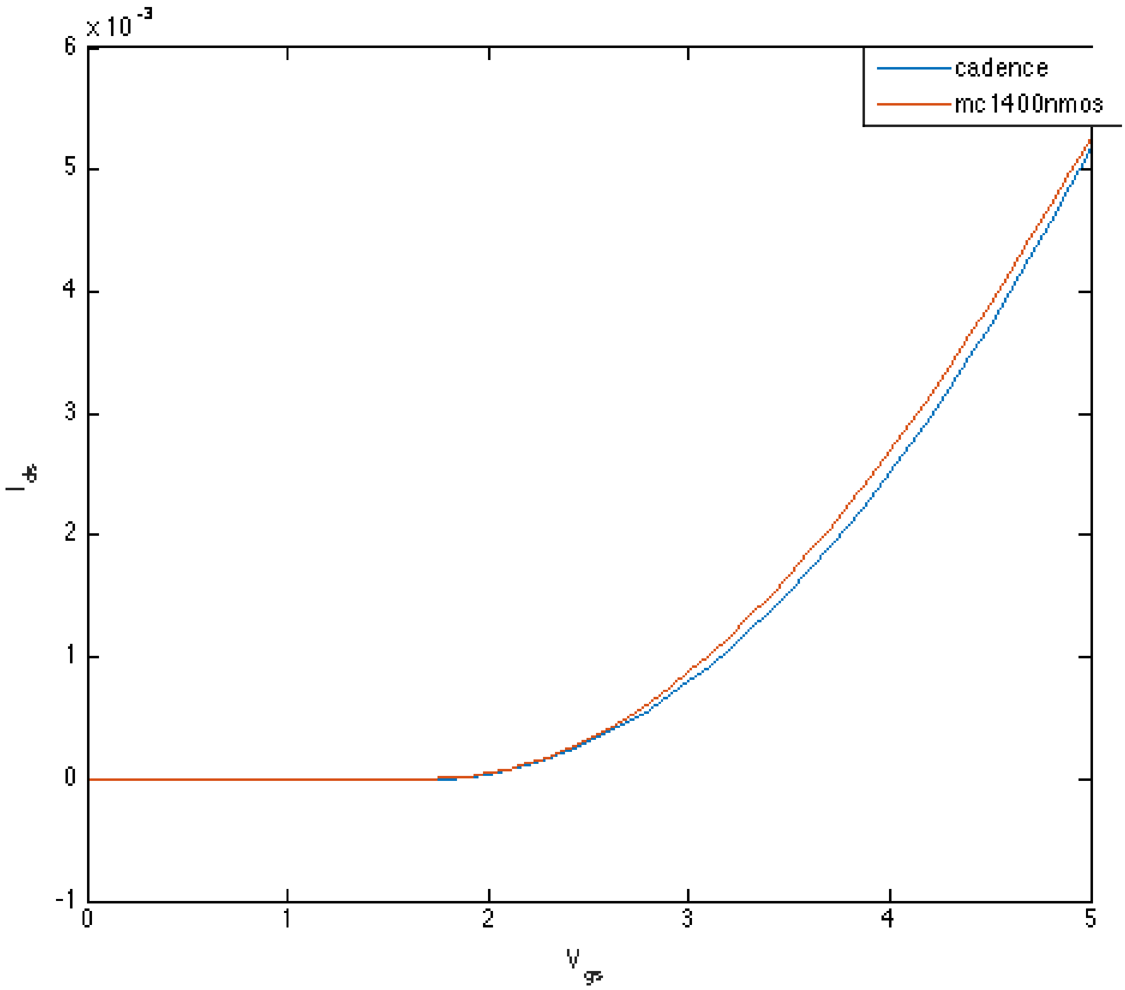
Figur 11: **task 6**



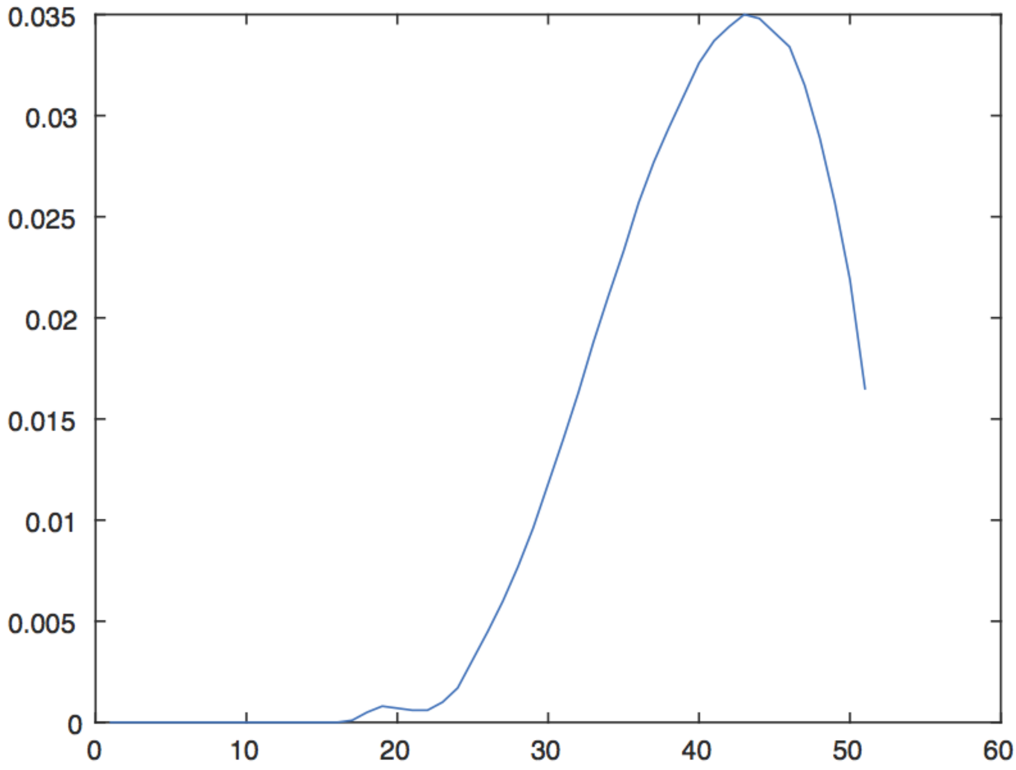
Figur 12: **task 6** Logaritmic



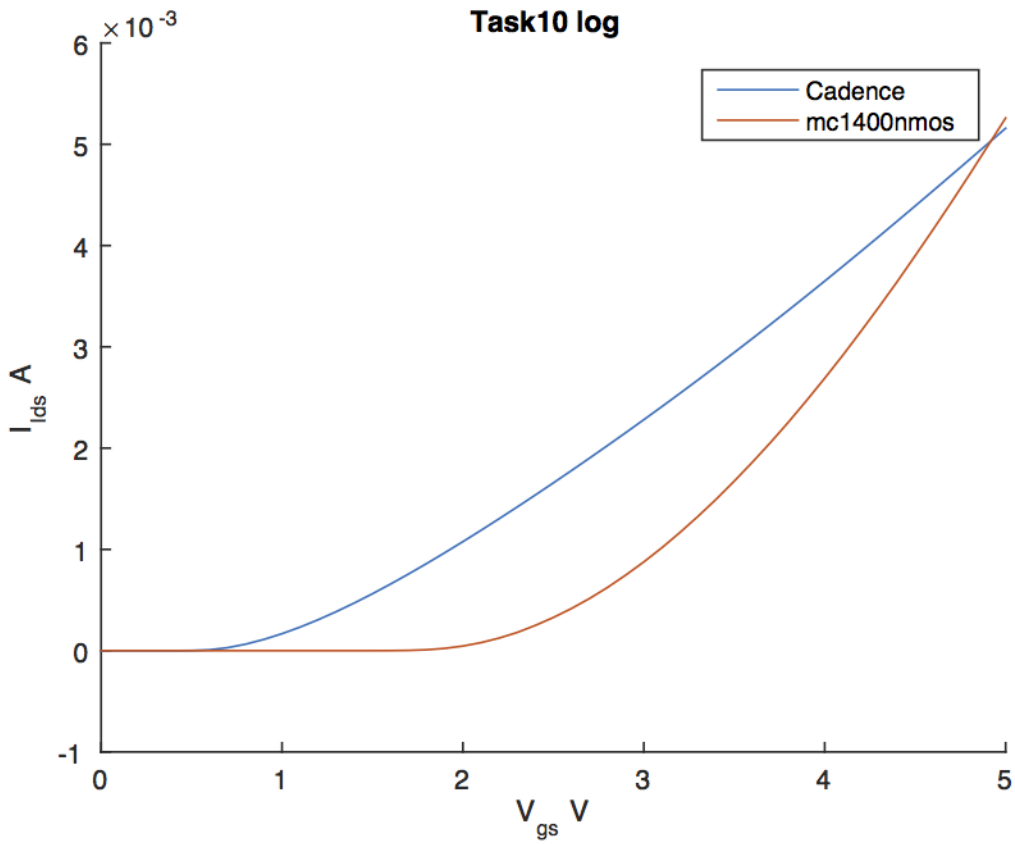
Figur 13: **task 8**



Figur 14: task 9



Figur 15: **task 10**



Figur 16: **task 11**: $\sqrt{I_{ds}}$ vs v_{gs}