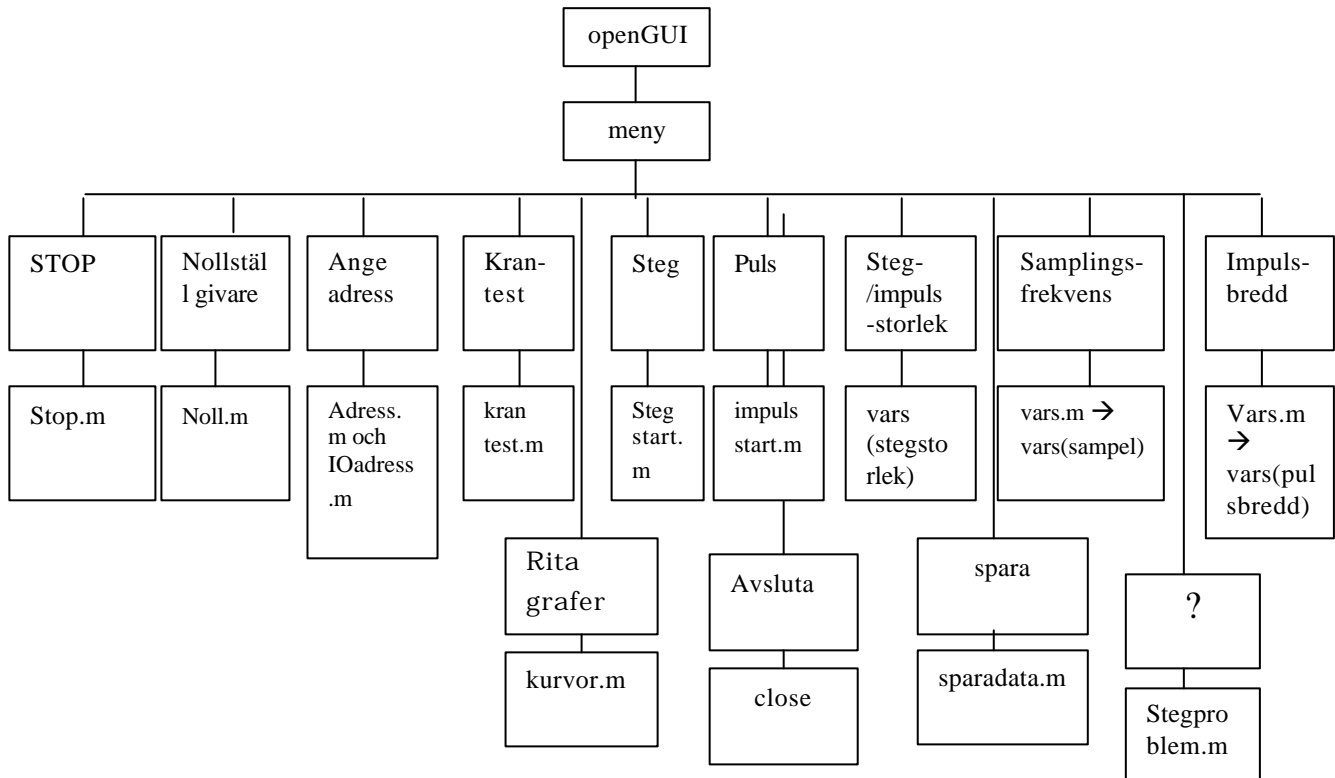


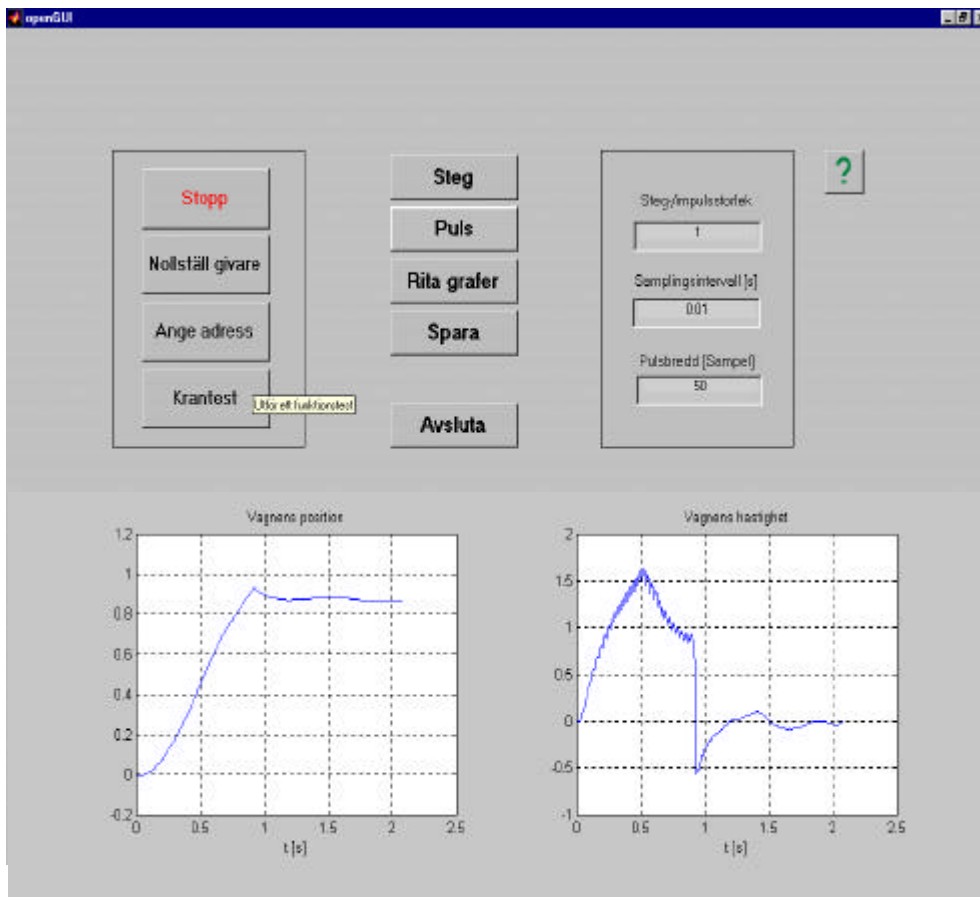
Bilaga 3, Steg- och pulstest för öppet system, openGUI

Då identifiering och validering av den matematiska modell som gavs i dokumentationen från FEEBACK är viktig skapades ett grafiskt användargränssnitt. Då det var ett av målen med examensarbetet att behärska hur ett GUI skapas gav detta ett utmärkt tillfälle bekanta sig med guide. Samt hur det går till att skapa ett GUI.



Figur 1. "Funktionsdiagram" för Steg- och pulstest för öppet system

I figur 1 redovisas hur de olika filerna är kopplade till de olika "pushbuttons" och "edit"-rutor som finns, se figur 2.



Figur 2. Det grafiska gränssnittet till openGUI

På följande sidor redovisas den källkod som är av intresse, det vill säga de filer som endast innehåller kod för menyer och liknande redovisas inte då de ej tillför något.

3.1 IOadress.m

Filen adress.m skapar ett figurfönster där användaren kan avläsa aktuell adress till I/O-kortet PCL-812. Då användaren anger en ny adress anropas filen IOadress.m som läser av inmatad adress och med hjälp av `pd_call` sätts den nya adressen.

```
function IOadress(action)
%Denna fil anropas av adress då användaren placerar
%markören i edit-rutan med 'Tag' 'NYAdress' från filen
%adress.m
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
%                               Micael Karlsson H97Ei
%-----

global BaseAddr; % = [], skapar alltså en tom global variabel
switch(action)
case 'NyAdress'
```

```

    Adress = get(gcbo,'String'); %Läser in strängen som skrivits i NYAddress
    BaseAddr = str2num(Adress);
end
pd_call('SetBaseAddress',BaseAddr); %Sätter I/O-adressen till BaseAddr

```

3.2 vars.m

filen vars.m anropas då något av värdena steg/pulsstorlek, Samplingtid och pulsbredd ändras. Värdena lagras i filen var.mat som används i flera av filerna.

```

function vars(action)
%I denna filen sparas variablerna u, SampTime PulsBredd
%som globala variabler vilket gör att man lätt kan känna
%av om de innehåller något eller ej.
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
%                               Micael Karlsson H97Ei
%-----

global u;
global SampTime;
global PulsBredd;

switch(action)
case 'stegstorlek'
    Str1 = get(gcbo,'String');
    u = str2num(Str1);
case 'sampel'
    Str2 = get(gcbo,'String');
    SampTime = str2num(Str2);
case 'pulsbredd'
    Str3 = get(gcbo,'String');
    PulsBredd = str2num(Str3);
end
save('var','u','SampTime','PulsBredd');

```

3.3 krantest.m

För att utföra ett stegtest för att ge användaren en visuellt bevis på att allt fungerar anropas filen krantest.m. Den fil ställer frågor då testet är klart som användaren skall besvara. Beroende på vilket svar som avges ställs antingen fler frågor eller så avbryts filen och experimenten kan påbörjas.

```

%Denna fil utför ett test för att undersöka om pendeln fungerar
%som den skall. Då testet utförts ställs frågor om hur det gått.
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
%                               Micael Karlsson H97Ei
%-----

pd_call('SetAlgNo',0);
pd_call('SetBaseAddress',544);
pd_call('ResetEncoder');

pd_call('SetAlgNo',5);

```

```

%P = [fric_comp Kp Ki Kd range umax isat]
P = [0.1, -1, 0, 0, 0.5, 0.4, 0.5];
Ref = 0.2;
pd_call('SetDesPosition',Ref);
pd_call('SetP',P);

svar = questdlg('Flyttades vagnen åt höger','Krantest','Ja','Nej','Vet
Ej','kbhit');
switch(svar) %svar
case 'Ja'
    msgbox(' Utmärkt !','openGUI');
case 'Nej'
    svar_nej = questdlg('Är knappen "POWER" i läge ON och lyser den runda
gröna knappen?','openGUI','Ja','Nej','Vet Ej','kbhit');
    switch(svar_nej) %svar_nej
    case 'Ja'
        Addr = pd_call('GetBaseAddress');
        if Addr ~= 544
            errordlg('FEL I/O-adress! Ange rätt adress och försök
igen.','openGUI');
        else
            errordlg('Kalla på lärare för hjälp!!!!','Kranfel','on');
        end
    case 'Nej'
        warndlg('Slå på "POWER" och den runda startknappen och kör
igen','openGUI');
    case 'Vet Ej'
    case 'kbhit'
    end %svar_nej

case 'Vet Ej'
    warndlg('Var mer uppmärksam och gör om testet från början','openGUI');
case 'kbhit'
end %svar

```

3.4 stegstart.m

Filen stegstart.m initierar helt enkelt ett steg med de värden som användaren angivit. Då det är klart anropas filen graf.m som "ritar" vagnens läge och hastighet som funktion av tiden i var sitt diagram.

```

%Denna funktion initierar ett steg för det öppna systemet.
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
% Micael Karlsson H97Ei
%-----

load var;
%Om någon av variablerna saknar värde används nedanstående
"standardvärden":
if isempty(u) == 1
    u = 0.2;
elseif isempty(SampTime) == 1
    SampTime = 0.01;
end
pd_call('AlgNo',0);
pd_call('SetSampleTime',SampTime);

%Möjliggör att hela banan används
%-----
P = pd_call('GetP');

```

```

P(1) = 1;
pd_call('SetP',P);

%Modifierar signalgeneratoren till konstant värde:
%-----
PW = pd_call('GetPW');
PW(1) = 0;
PW(2) = u;
pd_call('SetPW',PW);

val = questdlg('Avbryt med mellanslag då steget klar. Starta steget ?',...
              'openGUI','Ja','Nej','Avbryt','VetEj');
switch(val)
case 'Ja'
    pd_call('ResetTime'); %Nollställer tiden.
    pd_call('StartAcq'); %Börja samla data.
    pd_call('SetAlgNo',1); %Öppet system.
    pause;
    historia = pd_call('GetHistory'); %Hämta data från RTK-buffert.
    pd_call('SetAlgNo',0); %avbryter pågående försök.
    save('histall','historia','u','SampTime'); %Spara data i temporär .mat-
fil.
    graf; %Plottar vagnens läge och
position i menyfönstret.
case 'Nej'
case 'Avbryt'
case 'VetEj'
end %val

```

3.5 impulsstart.m

filen impulsstart fungerar på samma sätt som stegstart.m men en puls ges som insignal med den pulsbredd(n antal sampel) som användaren angivit.

%Denna funktion initierar ett steg för det öppna systemet.
%Det finns även möjlighet att bestämma pulsbredden i menyfönstret.
%I .mat-filen var.mat lagras u, SampTime och PulsBredd som används i denna fil:

```

%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
% Micael Karlsson H97Ei
%-----

```

```

load var;
%Om någon av variablerna saknar värde används nedanstående
"standardvärden":
if isempty(u) == 1
    u = 0.2;
elseif isempty(SampTime) == 1
    SampTime = 0.01;
elseif isempty(PulsBredd) == 1
    pulsbredd = 20;
end

pd_call('AlgNo',0);
pd_call('SetSampleTime',SampTime);

%Möjliggör att hela banan används

```

```

%-----
P = pd_call('GetP');
P(1) = 1;
pd_call('SetP',P);

%Modifierar signalgeneratoren för att få en
%puls med bredden SampTime*Pulsbredd sek.:
%-----
PW = pd_call('GetPW');
PW(1) = 1;
PW(2:5) = [SampTime SampTime*PulsBredd SampTime*5000 SampTime*5000];
PW(6:9) = [ 0 u 0 0 ];
pd_call('SetPW',PW);

val = questdlg('Avbryt med mellanslag då impulsen klar. Starta impulsen ?'
, 'openGUI', 'Ja', 'Nej', 'Avbryt', 'VetEj');
switch(val)
    case 'Ja'
        pd_call('ResetTime'); %Nollställ tidtagning.
        pd_call('StartAcq'); %Starta insamling av data.
        pd_call('SetAlgNo',1); %Öppet system.
        pause;
        historia = pd_call('GetHistory'); %Samla in mätdata.
        pd_call('SetAlgNo',0); %Stoppa pågående försök.
        save('histall', 'historia', 'u', 'SampTime'); %Spara "temp"-filen.
        graf; %Plottar vagnens läge och
        %hastighet i menyfönstret
    case 'Nej'
    case 'Avbryt'
    case 'VetEj'

end

```

3.6 graf.m

Filen graf.m "ritar" kurvorna för vagnens läge och hastighet i nedre halvan av menyfönstret då steg- eller pulstestet är klart. Denna fil anropas från filerna stegstart.m och impulsstart.m.

Det är först nu som en spara-funktion aktiveras i menyfönstret. Detta för att vara säker på att det finns data att spara.

Användaren ges också möjlighet att ändra färg på de båda graferna individuellt. Genom att högerklicka på grafen fås en meny upp där några olika färgalternativ finns.

```

%Denna fil anropas då användaren klickar på rutan
%"Rita grafer".
%Först laddas datan som sparats i en temporär fil
%där de senaste försöket sparas. Detta för att man
%inte skall behöva vänta på ett "misslyckat" försök
%
%

```

```

%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
% Micael Karlsson H97Ei
%-----

```

```

load histall; %läser in aktuell mätdata
if isempty(SampTime) == 1
    SampTime = 1;
elseif isempty(historia) == 1

```

```

    warndlg('Mätdata saknas!!','openGUI');
elseif isempty(u) == 1
end

%"Plockar ut" de variabler som är av intresse i denna fil
%-----
t = historia(1,:); %Tid i [s]
x = historia(4,:); %Position i [m]
v = historia(5,:); %Hastighet i [m/s]

%Initiera meny i figurfönstret.
%-----
cmenu = uicontextmenu;
cmenu2 = uicontextmenu;

%Ritar ut vagnens hastighet och position
%-----
subplot(2,2,3);hline1 = plot(t,x,'UIContextMenu',cmenu);
    grid;title('Vagnens position');xlabel('t [s]');
subplot(2,2,4);hline2 = plot(t,v,'UIContextMenu',cmenu2);
    grid;title('Vagnens hastighet');xlabel('t [s]');

%Sätter färg på grafen för vagnens
%position till den valda färgen.
%-----
cb1 = ['set(hline1,'Color','g')'];
cb2 = ['set(hline1,'Color','b')'];
cb3 = ['set(hline1,'Color','r')'];
cb4 = ['set(hline1,'Color','k')'];
cb5 = ['set(hline1,'Color','m')'];

%Sätter färg på grafen för vagnens
%hastighet till den valda färgen.
%-----
cb11 = ['set(hline2,'Color','g')'];
cb12 = ['set(hline2,'Color','b')'];
cb13 = ['set(hline2,'Color','r')'];
cb14 = ['set(hline2,'Color','k')'];
cb15 = ['set(hline2,'Color','m')'];

%Skapar menyalternativen för vagnens position.
%-----
item1 = uimenu(cmenu,'Label','grön', 'Callback',cb1);
item2 = uimenu(cmenu,'Label','blå', 'Callback',cb2);
item3 = uimenu(cmenu,'Label','röd', 'Callback',cb3);
item4 = uimenu(cmenu,'Label','svart', 'Callback',cb4);
item5 = uimenu(cmenu,'Label','magenta', 'Callback',cb5);

%Skapar menyalternativen för vagnens hastighet.
%-----
item11 = uimenu(cmenu2,'Label','grön', 'Callback',cb11);
item12 = uimenu(cmenu2,'Label','blå', 'Callback',cb12);
item13 = uimenu(cmenu2,'Label','röd', 'Callback',cb13);
item14 = uimenu(cmenu2,'Label','svart', 'Callback',cb14);
item15 = uimenu(cmenu2,'Label','magenta', 'Callback',cb15);

```

```
%Initierar en "spara"-möjlighet.
%-----
h1 = uicontrol('Units','points', ...
  'BackgroundColor',[0.752941176470588 0.752941176470588
0.752941176470588], ...
  'Callback','Sparadata', ...
  'FontSize',14, ...
  'FontWeight','bold', ...
  'HandleVisibility','callback', ...
  'ListboxTop',0, ...
  'Position',[299.25 343 99.75 30], ...
  'String','Spara', ...
  'Tag','Spara data', ...
  'TooltipString','Spara mätdata');
```

3.7 sparadata.m

Filen sparadata.m sparar den data dvs. mätdata, steg/impulsstorlek och samplingstiden. Om någon av variablerna skulle vara tom sparas inget och ett felmeddelande visas.

```
%Denna fil anropas då användaren klickar på knappen spara.
%en .mat-fil skapas med sökvägen: sv och namnet: fn.
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
%                               Micael Karlsson H97Ei
%-----

load histall %ladda in temporärt lagrad data.

if isempty(historia) == 0 & isempty(u) == 0 & isempty(SampTime) == 0
  [fn,sv] = uiputfile('a:\*.mat','Spara mätdata');
  namn = strcat(sv,fn);
  save(namn,'historia','u','SampTime');
elseif isempty(historia) == 1 | isempty(u) == 1 | isempty(SampTime) == 1
  warndlg('De variabler som skall sparas finns ej i arbetsarean','Fel vid
spara fil ...')
end
```

3.8 kurvor.m

M-filen kurvor ger användaren möjlighet att "plotta" de storheter som hon/han vill se i ett nytt figurfönster.

```
%Denna fil skapar en lista för olika plotalternativ
%som kan väljas ur en lista.
%
%-----
%Dessa filer ingår i examensarbetet: LQ-reglering av inverterad pendel
%
%                               Micael Karlsson H97Ei
%-----

h1 = uicontrol('Units','points', ...
  'BackgroundColor',[0.752941176470588 0.752941176470588
0.752941176470588], ...
  'Callback','Sparadata', ...
  'FontSize',14, ...
  'FontWeight','bold', ...
  'HandleVisibility','callback', ...
  'ListboxTop',0, ...
  'Position',[299.25 343 99.75 30], ...
```



```

'String','Spara', ...
'Tag','Spara data', ...
'TooltipString','Spara mätdata');

load('histall'); %Öppnar senaste mätserien.

str = {'Insignal';'Vagnens läge';'Vagnens hastighet';
      'Pendelns vinkel';'Pendelns vinkelhastighet'};
[urval,v] = listdlg('PromptString','Välj plottar:',...
                  'SelectionMode','multiple',...
                  'ListString',str);
str2 = {'stegstorlek []';'läge [m]';'Hastighet [m/s]';
      'Vinkel[rad]';'Vinkelhastighet [rad/s]'};

hist1 = zeros(size(historia));

%Lagrar mätdatan på "rätt plats"
hist1(1,:) = historia(6,:);
hist1(2,:) = historia(4,:);
hist1(3,:) = historia(5,:);
hist1(4,:) = historia(2,:);
hist1(5,:) = historia(3,:);

if v == 1
    %Skapar ett nytt figurfönster
    F=figure;
    set(F,'NumberTitle','off')
    set(F,'Position',[10 10 800 700])
    set(F,'Name','openGUI');
    ant = length(urval);
    if ant == 1
        plot(historia(1,:),hist1(urval(1),:));title(str(urval));
        xlabel('Tid [s]');ylabel(str2(1));grid
    elseif ant == 2

subplot(1,2,1);plot(historia(1,:),hist1(urval(1),:));title(str(urval(1)));
xlabel('Tid [s]');ylabel(str2(1));grid

subplot(1,2,2);plot(historia(1,:),hist1(urval(2),:));title(str(urval(2)));
xlabel('Tid [s]');ylabel(str2(2));grid
    elseif ant == 3

subplot(1,3,1);plot(historia(1,:),hist1(urval(1),:));title(str(urval(1)));
xlabel('Tid [s]');ylabel(str2(1));grid

subplot(1,3,2);plot(historia(1,:),hist1(urval(2),:));title(str(urval(2)));
xlabel('Tid [s]');ylabel(str2(2));grid

subplot(1,3,3);plot(historia(1,:),hist1(urval(3),:));title(str(urval(3)));
xlabel('Tid [s]');ylabel(str2(3));grid
    elseif ant == 4

subplot(2,2,1);plot(historia(1,:),hist1(urval(1),:));title(str(urval(1)));
xlabel('Tid [s]');ylabel(str2(1));grid

subplot(2,2,2);plot(historia(1,:),hist1(urval(2),:));title(str(urval(2)));
xlabel('Tid [s]');ylabel(str2(2));grid

subplot(2,2,3);plot(historia(1,:),hist1(urval(3),:));title(str(urval(3)));
xlabel('Tid [s]');ylabel(str2(3));grid

```

```
subplot(2,2,4);plot(historia(1,:),hist1(urval(4),:));title(str(urval(4)));
xlabel('Tid [s]');ylabel(str2(4));grid
elseif ant == 5

subplot(2,2,1);plot(historia(1,:),hist1(urval(1),:));title(str(urval(1)));
xlabel('Tid [s]');ylabel(str2(2));grid

subplot(2,2,2);plot(historia(1,:),hist1(urval(2),:));title(str(urval(2)));
xlabel('Tid [s]');ylabel(str2(2));grid

subplot(2,3,4);plot(historia(1,:),hist1(urval(3),:));title(str(urval(3)));
xlabel('Tid [s]');ylabel(str2(3));grid

subplot(2,3,5);plot(historia(1,:),hist1(urval(4),:));title(str(urval(4)));
xlabel('Tid [s]');ylabel(str2(4));grid

subplot(2,3,6);plot(historia(1,:),hist1(urval(5),:));title(str(urval(5)));
xlabel('Tid [s]');ylabel(str2(5));grid
end
hpop0 = uicontrol('Style','pushbutton',...
    'String','Stäng',...
    'Callback','close',...
    'FontWeight','bold',...
    'FontSize',10,...
    'Position',[340,15,70,25],...
    'TooltipString','Stäng fönstret');

end
```