ANEXOS

```
% José Carlos Sabana Padilla, Código: 19960806
% Este programa realiza la estimación de la velocidad angular wr, a través
% del método: MRAS based full order Luenberger Observer

clear; close all; clc; timexxx=cputime;

%xxxxxxxxxx Parámetros reales de motor de inducción xxxxxxxxxx
Ls=0.001269; Rs=0.04224;
Lr=0.001932; Rr=0.04117;
Lm=0.000911; polos=4/2; Jm=2.5;
alpha= 1-(Lm*Lm)/(Ls*Lr);
wr=188.5*3;


a11=-(Lm*Lm*Rr+Lr*Lr*Rs)/(alpha*Ls*Lr*Lr);
a12=0;
a13=(Lm*Rr)/(alpha*Ls*Lr*Lr);
a14=(Lm*wr*polos)/(alpha*Ls*Lr);
a21=0;
a22=-(Lm*Lm*Rr+Lr*Lr*Rs)/(alpha*Ls*Lr*Lr);
a23=-(Lm*wr*polos)/(alpha*Ls*Lr);
a24=(Lm*Rr)/(alpha*Ls*Lr*Lr);
a31=Lm*Rr/Lr;
a32=0;
a33=-Rr/Lr;
a34=-wr*polos;
a41=0;
a42=Lm*Rr/Lr;
a43=wr*polos;
a44=-Rr/Lr;

b11=1/(alpha*Ls);
b12=0;
b21=0;
b22=1/(alpha*Ls);
b31=0;
b32=0;
b41=0;
b42=0;

A = [ a11  a12  a13  a14
      a21  a22  a23  a24
      a31  a32  a33  a34
      a41  a42  a43  a44 ];

B = [ b11  b12
      b21  b22
      b31  b32
```

```
    b41  b42 ];

C=[ 1  0  0  0
   0  1  0  0 ];

%xxxxxx Voltaje de estator (alpha,beta) de Entrenamiento xxxxxx
dt = 0.0001;

ti1 = 0; tf1 =1.5 ; nt1=(tf1-ti1)/dt;
t=linspace(ti1,tf1,nt1); t=t';
f=linspace(0,30/4,nt1); f=f'; ft=f.*t;
amp1=linspace(0,365/4,nt1);amp1=amp1';
Tq=linspace(0,1,nt1); Tq=Tq'; tor_q1=sqrt(Tq).*sqrt(Tq); %torque cuadrático
va=amp1.*cos(2*pi*ft);
vb=amp1.*cos(2*pi*ft-2*pi/3);
vc=amp1.*cos(2*pi*ft+2*pi/3);
vs1_1=va;     % Se convierte de referencia abc/alpha,beta
vs2_1=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti2 = 1.5+dt; tf2 = 4.5; nt2=(tf2-ti2)/dt;
t=ti2:dt:tf2; t=t';
f=60/4;
amp1=365/4;
tor_q2=ones(nt2+1,1);%torque cuadrático
va=amp1*cos(2*pi*f*t);   % V/F constante
vb=amp1*cos(2*pi*f*t-2*pi/3);
vc=amp1*cos(2*pi*f*t+2*pi/3);
vs1_2=va;     % Se convierte de referencia abc/alpha,beta
vs2_2=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti3 = 4.5+dt; tf3 = 6; nt3=(tf3-ti3)/dt;
t=linspace(1,0,nt3);t=t.*t; t=t';
f=12;
amp1=linspace(365/4,0,nt3);amp1=amp1';
Tq=linspace(1,0,nt3); Tq=Tq'; tor_q3=sqrt(Tq).*sqrt(Tq); %torque cuadrático
va=amp1.*cos(2*pi*f*t);
vb=amp1.*cos(2*pi*f*t+2*pi/3);
vc=amp1.*cos(2*pi*f*t-2*pi/3);
vs1_3=va;     % Se convierte de referencia abc/alpha,beta
vs2_3=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti4 = 6+dt; tf4 = 9; nt4=(tf4-ti4)/dt;
t=ti4:dt:tf4; t=t';
amp1=0;
tor_q4=(0)*ones(nt4+2,1);%torque cuadrático
va=amp1*cos(2*pi*f*t);
vb=amp1*cos(2*pi*f*t-2*pi/3);
vc=amp1*cos(2*pi*f*t+2*pi/3);
vs1_4=va;     % Se convierte de referencia abc/alpha,beta
vs2_4=(1/sqrt(3))*va+(2/sqrt(3))*vb;
```

```
ti5 = 9+dt; tf5 = 10.5; nt5=(tf5-ti5)/dt;
t=linspace(0,1,nt5); t=t.*t; t=t';
f=20;
amp1=linspace(0,365/2,nt5);amp1=amp1';
Tq=linspace(0,1,nt5); Tq=Tq'; tor_q5=sqrt(Tq).*sqrt(Tq);%torque cuadrático
va=amp1.*cos(2*pi*f*t);
vb=amp1.*cos(2*pi*f*t-2*pi/3);
vc=amp1.*cos(2*pi*f*t+2*pi/3);
vs1_5=va;     % Se convierte de referencia abc/alpha,beta
vs2_5=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti6 = 10.5+dt; tf6 = 13.5; nt6=(tf6-ti6)/dt;
t=ti6:dt:tf6; t=t';
f=30;
tor_q6=ones(nt6+1,1); %torque cuadrático
va=365/2*cos(2*pi*f*t);   % V/F constante
vb=365/2*cos(2*pi*f*t-2*pi/3);
vc=365/2*cos(2*pi*f*t+2*pi/3);
vs1_6=va;     % Se convierte de referencia abc/alpha,beta
vs2_6=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti7 = 13.5+dt; tf7 = 15;
nt7=(tf7-ti7)/dt;
t=linspace(1,0,nt7); t=t.*t; t=t'; f=22;
amp1=linspace(365/2,0,nt7); amp1=amp1';
Tq=linspace(1,0,nt7); Tq=Tq'; tor_q7=sqrt(Tq).*sqrt(Tq);%torque cuadrático
va=amp1.*cos(2*pi*f*t+pi);
vb=amp1.*cos(2*pi*f*t+pi+2*pi/3);
vc=amp1.*cos(2*pi*f*t+pi-2*pi/3);
vs1_7=va;     % Se convierte de referencia abc/alpha,beta
vs2_7=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti8 = 15+dt; tf8 = 18; nt8=(tf8-ti8)/dt;
t=ti8:dt:tf8; t=t';
amp1=365*0;
tor_q8=(0)*ones(nt8+1,1);%torque cuadrático
va=amp1*cos(2*pi*f*t);
vb=amp1*cos(2*pi*f*t-2*pi/3);
vc=amp1*cos(2*pi*f*t+2*pi/3);
vs1_8=va;     % Se convierte de referencia abc/alpha,beta
vs2_8=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti9 = 18+dt; tf9 = 19.5; nt9=(tf9-ti9)/dt;
t=linspace(0,1,nt9); t=t.*t; t=t'; f=40;
amp1=linspace(0,365,nt9); amp1=amp1';
tor_q9=sqrt(amp1/365).*sqrt(amp1/365);%torque cuadrático
va=amp1.*cos(2*pi*f*t);
vb=amp1.*cos(2*pi*f*t-2*pi/3);
vc=amp1.*cos(2*pi*f*t+2*pi/3);
```

```
vs1_9=va;     % Se convierte de referencia abc/alpha,beta
vs2_9=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti10 = 19.5+dt; tf10 = 22.5; nt10=(tf10-ti10)/dt;%ti2 = 5+dt; tf2 = 8; nt2=(tf2-ti2)/dt;
t=ti10:dt:tf10; t=t';
f=60;
amp1=365;
tor_q10=ones(nt10+1,1);%torque cuadrático
va=amp1*cos(2*pi*f*t);   % V/F constante
vb=amp1*cos(2*pi*f*t-2*pi/3);
vc=amp1*cos(2*pi*f*t+2*pi/3);
vs1_10=va;     % Se convierte de referencia abc/alpha,beta
vs2_10=(1/sqrt(3))*va+(2/sqrt(3))*vb;

ti11 = 22.5+dt; tf11 =24; nt11=(tf11-ti11)/dt;
t=linspace(1,0,nt11); t=t.*t; t=t'; f=45;
amp1=linspace(365,0,nt11); amp1=amp1';
tor_q11=sqrt(amp1/365).*sqrt(amp1/365);%torque cuadrático
va=amp1.*cos(2*pi*f*t);
vb=amp1.*cos(2*pi*f*t+2*pi/3);
vc=amp1.*cos(2*pi*f*t-2*pi/3);
vs1_11=va;     % Se convierte de referencia abc/alpha,beta
vs2_11=(1/sqrt(3))*va+(2/sqrt(3))*vb;

vs1=[vs1_1
    vs1_2
    vs1_3
    vs1_4
    vs1_5
    vs1_6
    vs1_7
    vs1_8
    vs1_9
    vs1_10
    vs1_11];
vs2=[vs2_1
    vs2_2
    vs2_3
    vs2_4
    vs2_5
    vs2_6
    vs2_7
    vs2_8
    vs2_9
    vs2_10
    vs2_11];
vs=[vs1 vs2]';
nv = length(vs);

torque_q=[tor_q1
```

```
        tor_q2
        tor_q3
        tor_q4
        tor_q5
        tor_q6
        tor_q7
        tor_q8
        tor_q9
        tor_q10
        tor_q11];


%xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Ob=[ C
   C*A
   C*A^2
   C*A^3 ];

detOb=rank(Ob);

q1 = input('Peso q1 : ');
q2 = input('Peso q2 : ');
q3 = input('Peso q3 : ');
q4 = input('Peso q4 : ');
Q = diag([ q1 q2 q3 q4]);

%xxxxxx Kp y Ki son pesos obtenidos de func. Lyapunov xxxxxxx
Kp = input('Constante proporcional (Kp) : ');
Ki = input('Constante integrativa (Ki) : ');

x(1,1) = 0;     % Corriente alpha del estator inicial
x(2,1) = 0;     % Corriente beta del estator inicial
x(3,1) = 0.0;   % Flujo alpha del rotor inicial
x(4,1) = 0.0;   % Flujo beta del estator inicial
xh(1,1) = 0;
xh(2,1) = 0;
xh(3,1) = 0;
xh(4,1) = 0;
wrh=0;wx=0;er_rms_wh=0;Te=0;
integra_1=0;

k = 1;
for tt = dt:dt:(tf9-4*dt)

   y=C*(x+[randn(1,1)*0 randn(1,1)*0 0 0]');
   volas(k,1)=vs(1,k);
   volbs(k,1)=vs(2,k);
   ias(k,1) = x(1,1); %y(1,1);
   ibs(k,1) = x(2,1); %y(2,1);
   flowar(k,1) = x(3,1);
```

```
flowbr(k,1) = x(4,1);
xh1(k,1) = xh(1,1);
xh2(k,1) = xh(2,1);
xh3(k,1) = xh(3,1);
xh4(k,1) = xh(4,1);
error_ia(k,1)=x(1,1)-xh(1,1);
error_ib(k,1)=x(2,1)-xh(2,1);
error_fa(k,1)=x(3,1)-xh(3,1);
error_fb(k,1)=x(4,1)-xh(4,1);
error_wr(k,1)=wx-wrh;  er_rms_wh=er_rms_wh + (wx-wrh)^2;
velh_ang(k,1)= wrh;
vel_ang(k,1)= wx;
torque_ele(k,1)=Te;
torque_carg(k,1)=Tl;
time(k,1)=tt;

%xxxxxxxxxxxxxx Se actualiza matrices reales xxxxxxxxxxxxxx
a14=(Lm*wx*polos)/(alpha*Ls*Lr);
a23=-(Lm*wx*polos)/(alpha*Ls*Lr);
a34=-wx*polos;
a43=wx*polos;

%a11x=-(Lm*Lm*(1.5*Rr)+Lr*Lr*Rs)/(alpha*Ls*Lr*Lr);
%a22x=-(Lm*Lm*(1.5*Rr)+Lr*Lr*Rs)/(alpha*Ls*Lr*Lr);

%Ak=[1+a11x*dt  a12*dt   (a13*1.5)*dt    a14*dt
%    a21*dt    1+a22x*dt  a23*dt    (a24*1.5)*dt
%    (a31*1.5)*dt    a32*dt    1+(a33*1.5)*dt  a34*dt
%    a41*dt    (a42*1.5)*dt    a43*dt    1+(a44*1.5)*dt];

Ak=[1+a11*dt  a12*dt    a13*dt    a14*dt
   a21*dt    1+a22*dt  a23*dt    a24*dt
   a31*dt    a32*dt    1+a33*dt  a34*dt
   a41*dt    a42*dt    a43*dt    1+a44*dt ];


Bk=[b11*dt   b12*dt
    b21*dt  b22*dt
    b31*dt  b32*dt
    b41*dt  b42*dt];

%xxxxxxxxxxxx Se actualiza matrices observadas xxxxxxxxxxxx
a14=(Lm*wrh*polos)/(alpha*Ls*Lr);
a23=-(Lm*wrh*polos)/(alpha*Ls*Lr);
a34=-wrh*polos;
a43=wrh*polos;

Ah = [ a11  a12  a13  a14
      a21  a22  a23  a24
      a31  a32  a33  a34
```

```
        a41  a42  a43  a44 ];

  R=[1];
  P = are(Ah',B*inv(R)*B',Q);
  L=P*B;

  Ahk=[1+a11*dt a12*dt    a13*dt     a14*dt
     a21*dt    1+a22*dt  a23*dt     a24*dt
     a31*dt    a32*dt    1+a33*dt   a34*dt
     a41*dt    a42*dt    a43*dt     1+a44*dt];

  Bhk=Bk;

  l11=L(1,1); l12=L(1,2);
  l21=L(2,1); l22=L(2,2);
  l31=L(3,1); l32=L(3,2);
  l41=L(4,1); l42=L(4,2);

  Lk=[l11*dt  l12*dt
     l21*dt  l22*dt
     l31*dt  l32*dt
     l41*dt  l42*dt];

  %xxxxxxxxxxxxxx Calculo valores futuros xxxxxxxxxxxxxxxxxxxx
  x = Ak*x + Bk*[vs(1,k) vs(2,k)]' ;
  xh = Ahk*xh+Bhk*[vs(1,k) vs(2,k)]'+Lk*(y-C*xh);

  %Utilizando función de Lyapunov calculo velocidad observada
  e_ias=y(1,1)-xh(1,1);
  e_ibs=y(2,1)-xh(2,1);
  dif1=e_ias*xh(4,1)-e_ibs*xh(3,1);
  integra_1=integra_1+dif1*dt;
  wrh=Kp*(dif1)+Ki*integra_1;

  %xxxxxxxxxx Se calcula la velocidad real xxxxxxxxxxxxxxxxxxx
  Te=polos*(x(3,1)*x(2,1)-x(4,1)*x(1,1)); Tl=500*torque_q(k,1);
  wp=(Te-Tl)/Jm;
  wx =wx+wp*dt;

  k = k+1;

end

cputime-timexxx

vel_MRAS=velh_ang;
error_MRAS=error_wr;
vel_angMRAS=vel_ang;
sqrt(er_rms_wh)/k
```

```
save paper1 time vel_angMRAS vel_MRAS error_MRAS;

figure(1);
subplot(2,1,1)
plot(time,volas,'-r');
title('Voltaje alpha Estator '); xlabel('s'); ylabel('V');
subplot(2,1,2)
plot(time,volbs,'-r');
title('Voltaje beta Estator '); xlabel('s'); ylabel('V');
figure(2);
subplot(4,1,1)
plot(time,ias,'-r',time,xh1,'-b');
title('Corriente alpha Estator (real->rojo / observada->azul)'); xlabel('s');ylabel('A');
subplot(4,1,2)
plot(time,ibs,'-r',time,xh2,'-b');
title('Corriente beta Estator (real->rojo / observada->azul)');xlabel('s');ylabel('A');
subplot(4,1,3)
plot(time,flowar,'-r',time,xh3,'-b');
title('Flujo alpha del Rotor (real->rojo / observada->azul)'); xlabel('s');ylabel('Wb');
subplot(4,1,4)
plot(time,flowbr,'-r',time,xh4,'-b');
title('Flujo beta del Rotor (real->rojo / observada->azul)'); xlabel('s');ylabel('Wb');
figure(3);
subplot(4,1,1)
plot(time,error_ia,'-r');
title('Error de Corriente alpha'); xlabel('s');ylabel('A');
subplot(4,1,2)
plot(time,error_ib,'-r');
title('Error de Corriente beta'); xlabel('s');ylabel('A');
subplot(4,1,3)
plot(time,error_fa,'-r');
title('Error de Flujo alpha'); xlabel('s');ylabel('Wb');
subplot(4,1,4)
plot(time,error_fb,'-r');
title('Error de Flujo beta'); xlabel('s');ylabel('Wb');
figure(4);
subplot(2,1,1)
plot(time,vel_ang,'-r',time,velh_ang,'b');
ylim([-10 200]);%ylim([0 160]);
title('Velocidad angular (real->rojo / observada->azul)'); xlabel('s');ylabel('rad/s');
subplot(2,1,2)
plot(time,error_wr,'-r');
ylim([-20 20]);%ylim([-40 40]);
title('Error de Velocidad angular'); xlabel('s');ylabel('rad/s');
figure(5);
subplot(2,1,1)
plot(time,torque_ele,'-r');
ylim([-120 120]);
title('Torque Eléctrico'); xlabel('s');ylabel('Nm');
subplot(2,1,2)
```

```
plot(time,torque_carg,'-r');
ylim([-120 120]);
title('Torque de carga'); xlabel('s');ylabel('Nm');

%xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
%xxxxxxxxxxxxx   con dt= 0.0001   xxxxxxxxxxxxxxxxxxxxxxxxxxx
%Peso q1 : 1
%Peso q2 : 1
%Peso q3 : 0.0001
%Peso q4 : 0.0001
%Constante proporcional (Kp) : 0.01
%Constante integrativa (Ki) : 500
```