**POLITECNICO DI MILANO**

**Scuola di Ingegneria Industriale e dell'Informazione**

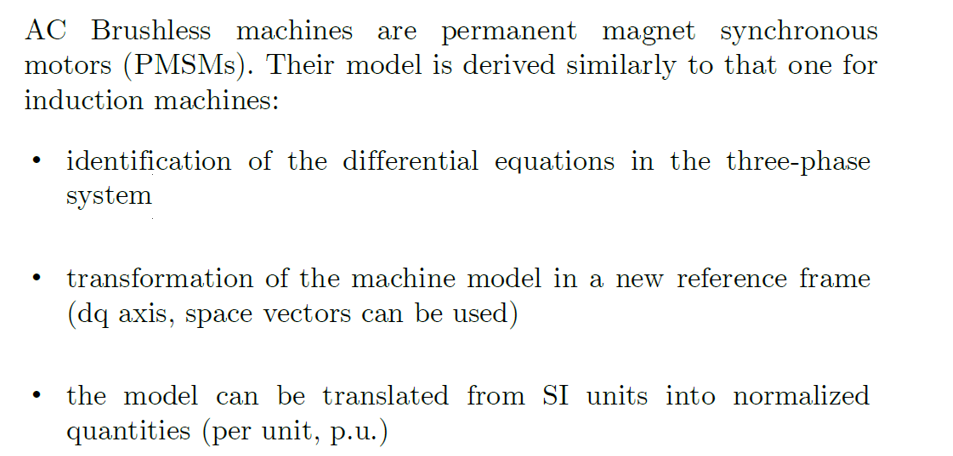
**Corso di Laurea in Ingegneria Elettrica**

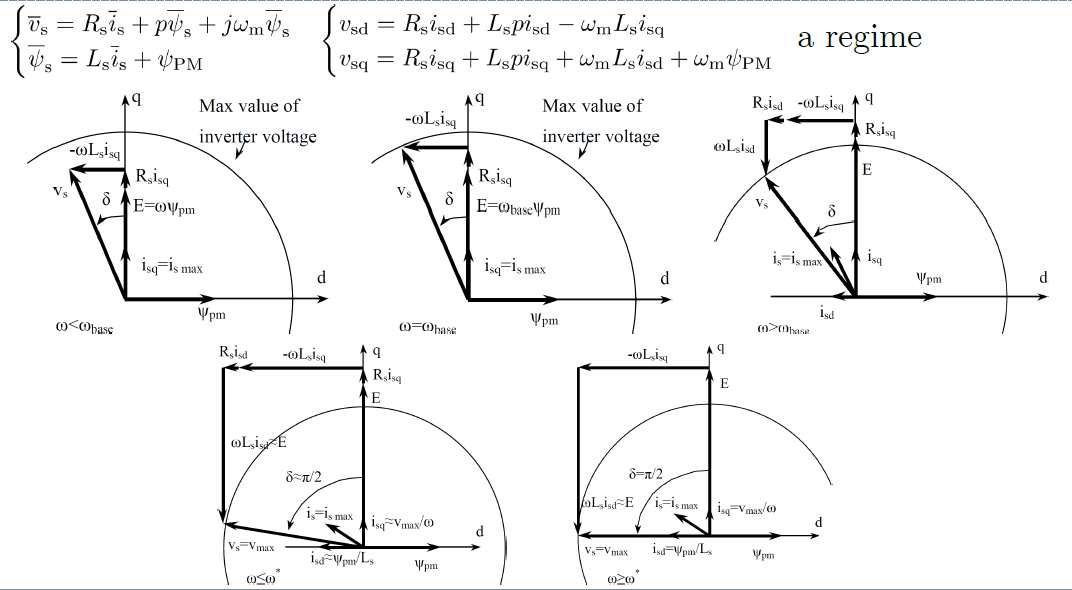
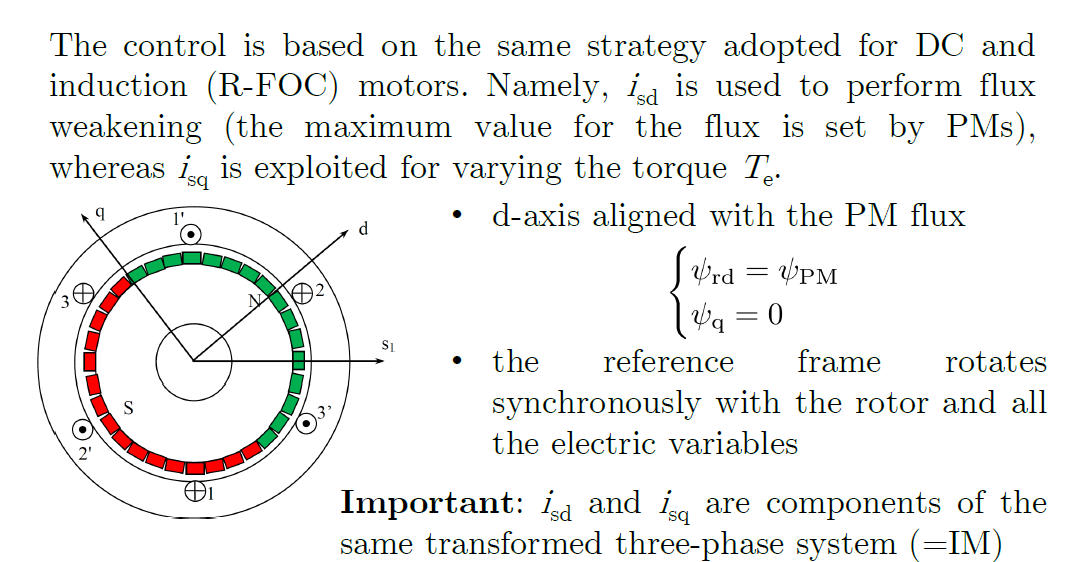
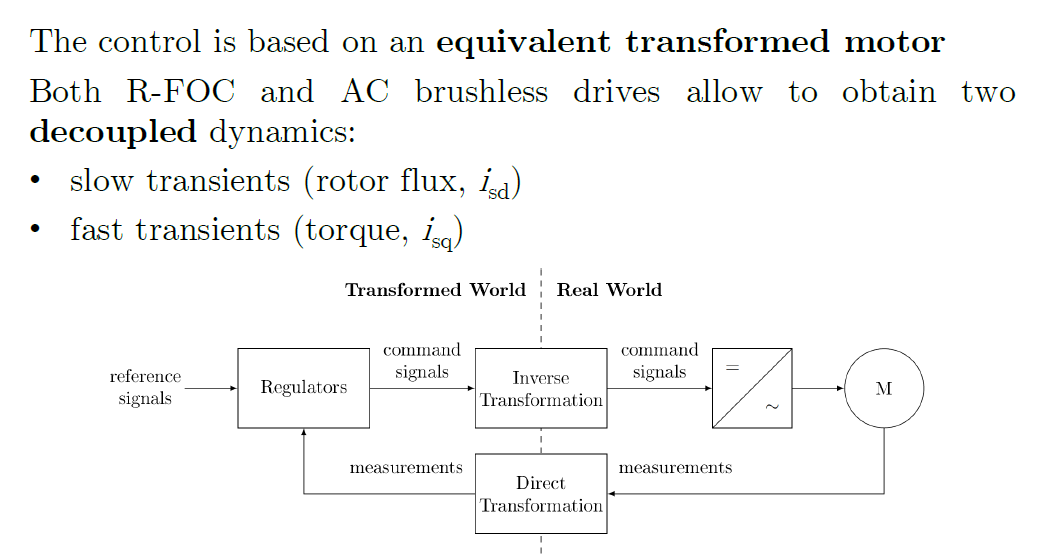


AC Brushless control for Alfa Romeo induction motor

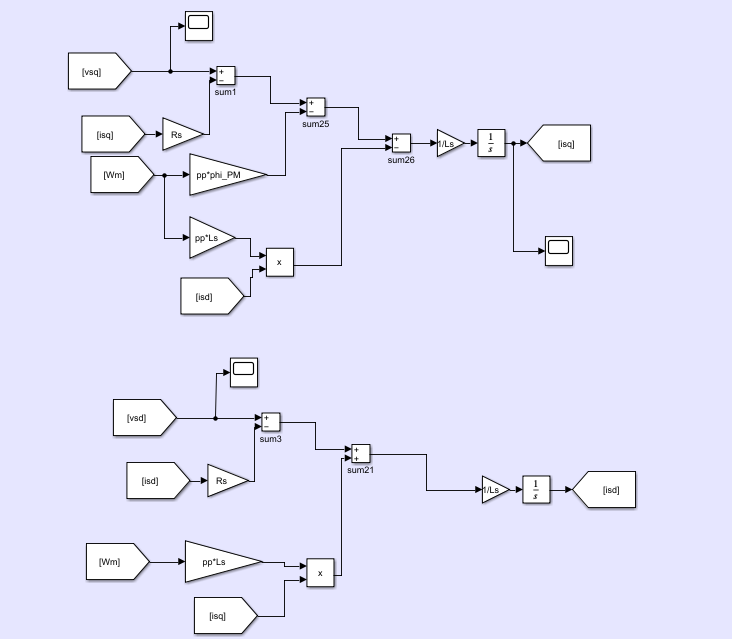
Alessandro Secchi

Matr. 944668

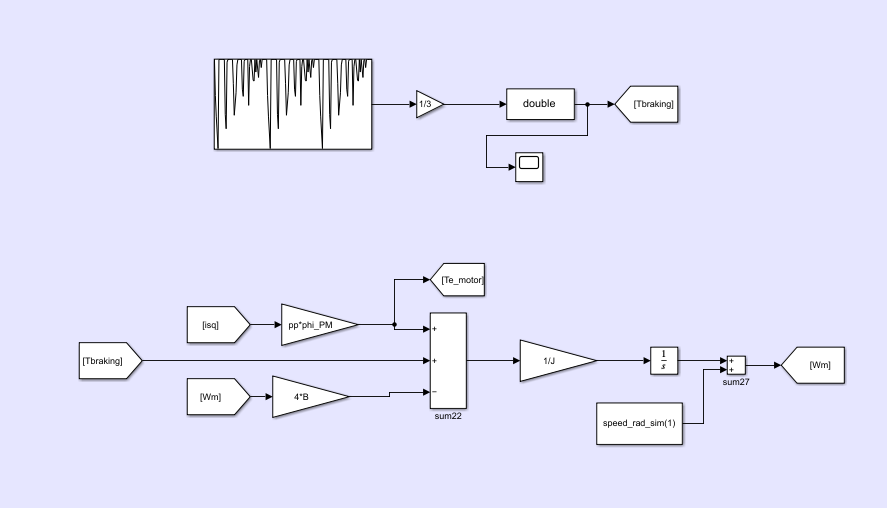




For the motor model, a dq reference frame has ben chosen:

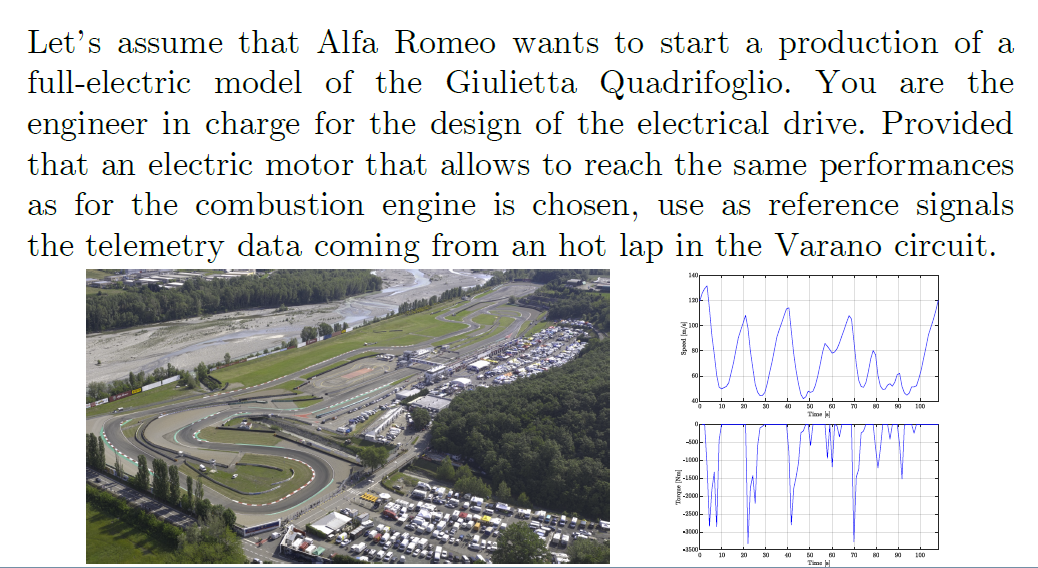


Moreover the mechanical equations are here represented by adding also a contribute of the braking torque. (the mechanical braking torque has been considered 1/3 of the overall braking torque thus electrical + mechiancal).



Exercise and data (please refer to the code at the end of the report):

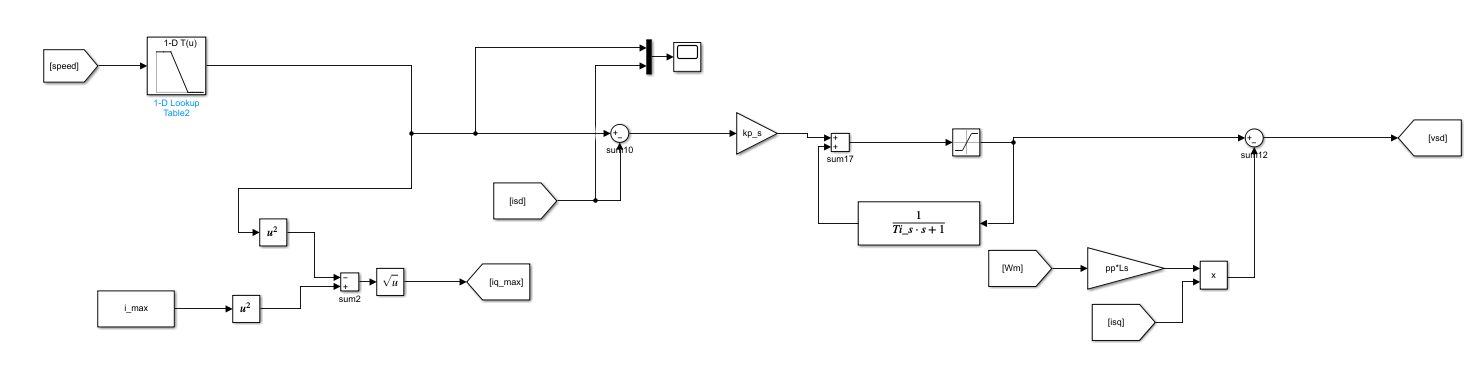




The lookup tables in the Simulink models have been built in the matlab script (refer to the code at the end of the report), by considering an arbitrary strategy for the flux weakening i.e. 70 km/h has been assumed as base speed and the slope of isd has been considered linear.

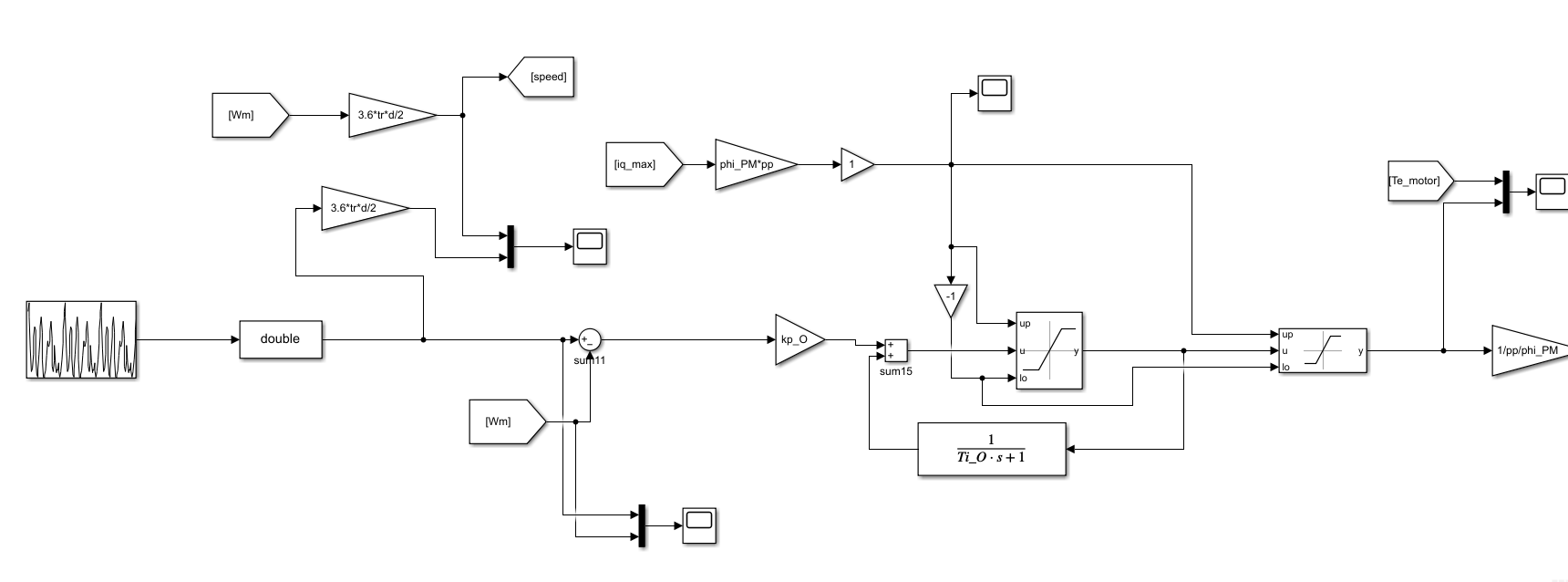
Simulink models:

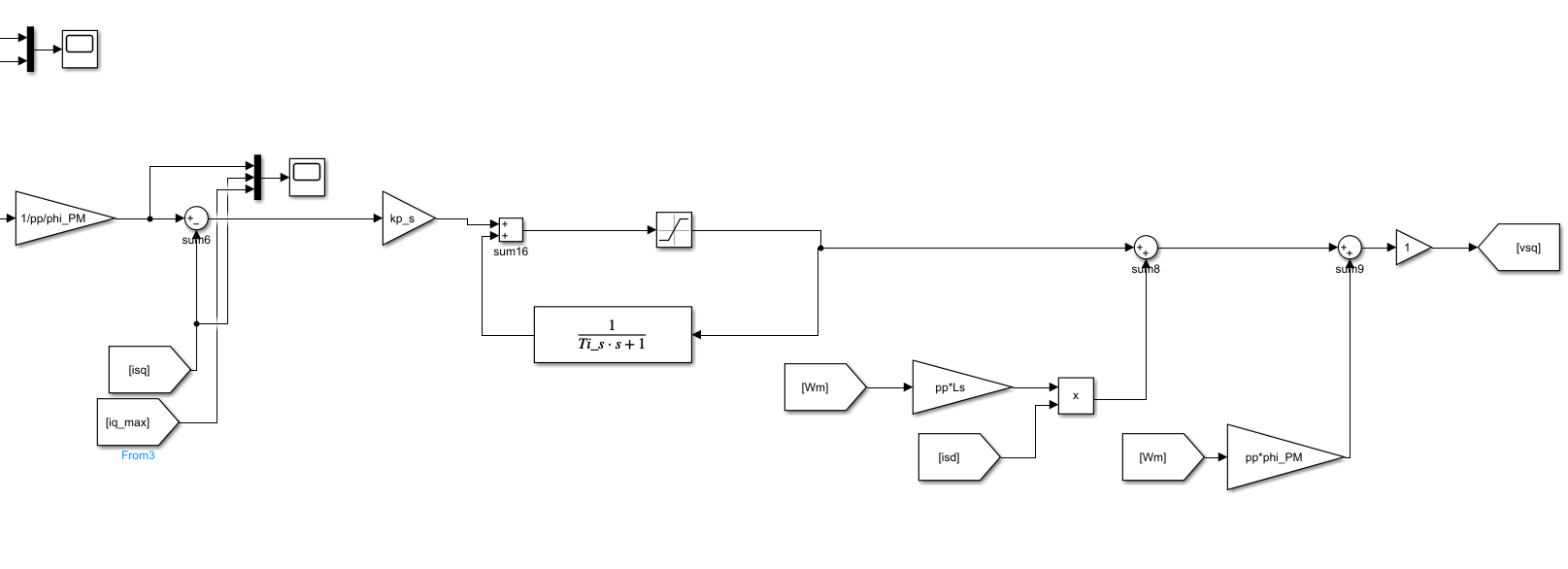
Isd current control and flux weakening region



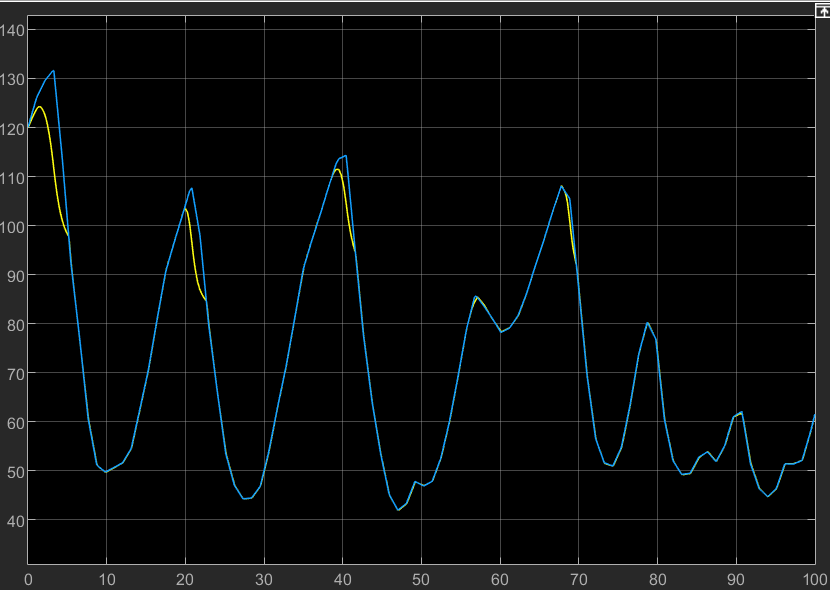
Note that “Pythagorean theorem” to fine the isq max has been used (considering the i\_max).

Torque and isq control:

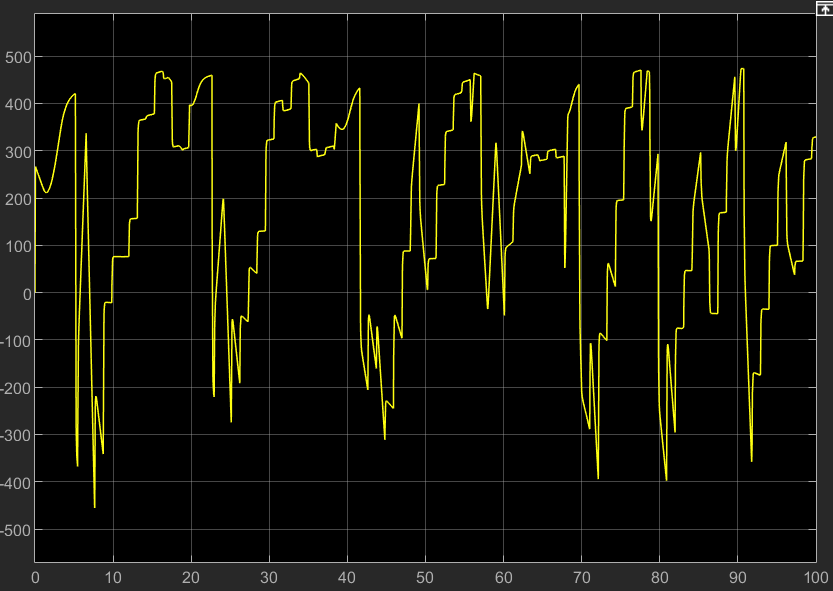




Speed profile in km/h:

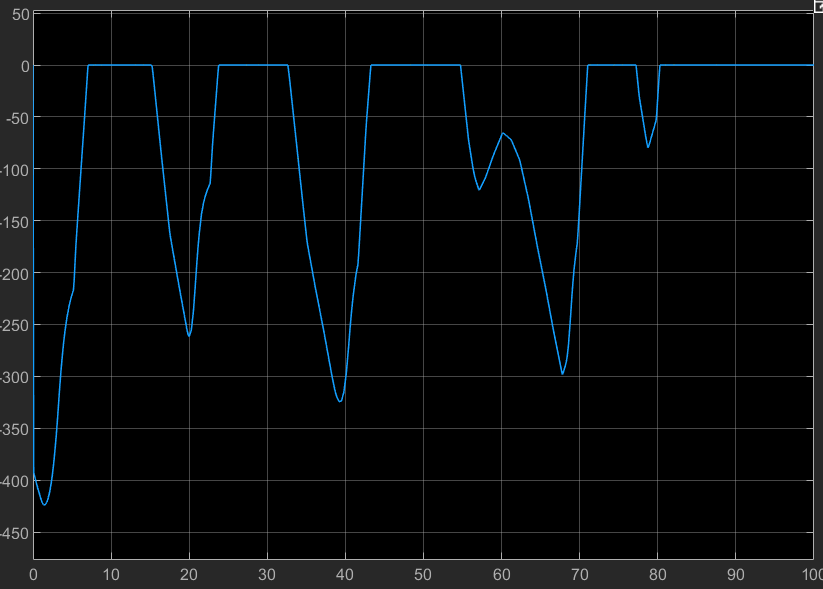


Isq measured and reference are well matched:

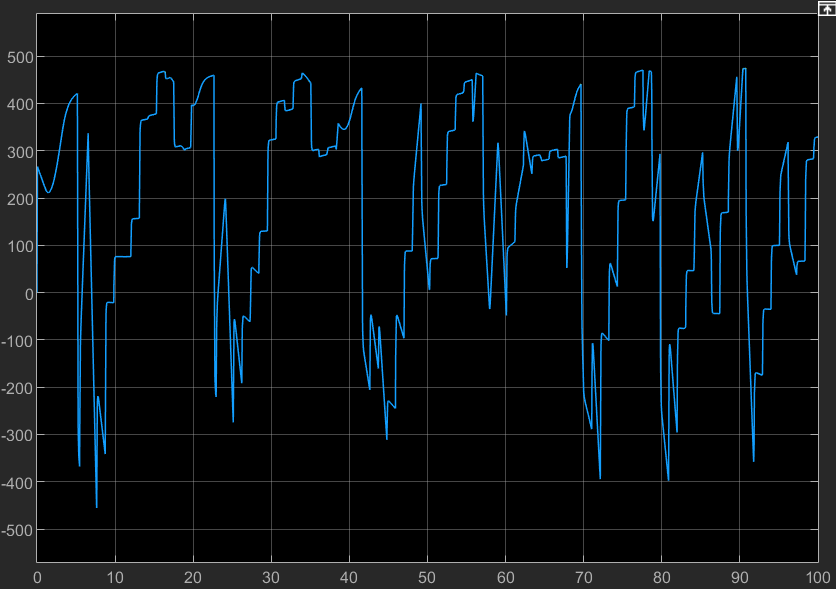


Note that the isq measured is directly taken from the motor scheme, but actually it should be taken from the phases currents and transformed in the respective space phasor.

In the following graph, the flux weakening over the base speed is well represented by isd and its reference:



Lastly, also the torque can follow the reference and when negative, it shown the “regenerative braking” due to the declaration in the speed profile.



clc

clear all

%% DC machine parameters

load('Traction\_e402B\_online');

load('ac\_brushless\_data');

%parameters of 1 motor = equivalent motor

V\_batt = 800; %voltage

pp=2; %pole pairs of each motor

cos\_fi=1; %power factor of each motor

Rs=0.1; %ohm

Ls=0.8\*10^-3; %H

d=60\*10^-2; %diameter wheels

tr=1/3; %trasmission ratio

mass\_tot=1580; %kg

v\_max=240\*1000/3600; %max speed m/s

omega\_max=v\_max/tr/d\*2; %rad/s

omega\_rated=400; %rad/s

v\_rated=omega\_rated\*tr\*d/2\*3.6; %km/h

eff=0.95; %efficiency

J=mass\_tot\*v\_max^2/omega\_max^2;

%J\_eq=mass\_tot\*(tr\*d)^2/4;

V\_base=70/3.6; %base speed assumed in m/s

Pn=180\*10^3; %tot mechanical power [W]

Pe\_tot=Pn/eff;

phi\_PM=0.5; %Wb

Tn = Pn/omega\_rated; % Nominal torque provided by the machine

i\_max=Tn/pp/eff/phi\_PM;

V\_motor=Pe\_tot/sqrt(3)/i\_max/cos\_fi;

tau\_s=Ls/Rs;

B=0.065;

tau\_O=J/B;

a\_max=4; %m/s

T\_max=mass\_tot\*a\_max\*d/2\*tr;

T\_friction=4\*B\*omega\_rated;

% B = Tfriction/rated\_speed\_motor; % friction coefficient < Or

% tau\_mec=J\_eq/B;

speed\_rad\_sim=speed\_kmh./((d/2)\*tr\*3.6); %rad/s

speed\_ref =[0, 35, 70,150,160,170,180,190,200,210,220];

id\_ref=[0,0,0,-phi\_PM/Ls,-phi\_PM/Ls,-phi\_PM/Ls,...

-phi\_PM/Ls,-phi\_PM/Ls,-phi\_PM/Ls,-phi\_PM/Ls,-phi\_PM/Ls];

%% PI controller design parameters

s=tf('s');

%Gi

tau\_s\_desired=tau\_s/1000;

wc\_s=2\*pi/tau\_s\_desired;

%GO

tau\_O\_desired=tau\_O/1000;

wc\_O=2\*pi/tau\_O\_desired;

%tf

Gi = 1/(Rs+Ls\*s);

GO = 1/(B+J\*s);

%% Zero Pole cancellation (90 phase margin)

% %PI parameters ia

% kp\_a=wc\_a\*La;

% ki\_a=wc\_a\*Ra;

% Regi=kp\_a+ki\_a/s

% Ti\_a=kp\_a/ki\_a;

% %tf open loop

% Li=Regi\*Gi;

% %tf close loop

% Fi=Li/(1+Li);

% % figure

% % bode(Li)

% % figure

% % bode(Fi)

% %PI parameters ie

% kp\_e=wc\_e\*Le;

% ki\_e=wc\_e\*Re;

% Rege=kp\_e+ki\_e/s

% Ti\_e=kp\_e/ki\_e;

% %tf open loop

% L\_e=Rege\*Ge;

% %tf close loop

% Fe=L\_e/(1+L\_e);

% % figure

% % bode(L\_e)

% % figure

% % bode(Fe)

% %PI parameters speed

% kp\_O=wc\_O\*J\_eq;

% ki\_O=wc\_O\*B;

% RegO=kp\_O+ki\_O/s

% Ti\_O=kp\_O/ki\_O;

% %tf open loop

% LO=RegO\*GO;

% %tf close loop

% FO=LO/(1+LO);

% % figure

% % bode(LO)

% % figure

% % bode(FO)

% %saturation

% SatUp\_Va = Van; % [A]

% SatLow\_Va = -Van;

% SatUp\_T = Tn\*10; % [V]

% SatLow\_T = -Tn\*10;

% SatUp\_Ve = Ven\*1.1;

% SatLow\_Ve = 0;

%% Pidtool

%otherwise use pidtool

phase\_m=90;

%pidtool(Gi)

opt=pidtuneOptions('PhaseMargin', phase\_m);

par\_regi=pidtune(Gi,'PI',wc\_s,opt);

ki\_s=par\_regi.Ki;

kp\_s=par\_regi.Kp;

Regi=kp\_s+ki\_s/s

Ti\_s=kp\_s/ki\_s;

%tf open loop

Li=Regi\*Gi;

%tf close loop

Fi=Li/(1+Li);

% figure

% bode(Li);

% figure

% bode(Fi);

% figure

% margin(Li);

%pidtool(GO)

par\_reg\_speed=pidtune(GO,'PI',wc\_O,opt);

ki\_O=par\_reg\_speed.Ki;

kp\_O=par\_reg\_speed.Kp;

RegO=kp\_O+ki\_O/s

Ti\_O=kp\_O/ki\_O;

%tf open loop

LO=RegO\*GO;

%tf close loop

FO=LO/(1+LO);

% figure

% bode(LO);

% figure

% bode(FO);

% figure

% margin(LO);