



Electrical Machines

Galina Demidova

Assoc. Prof.

demidova@itmo.ru

Dmitry Lukichev

Assoc. Prof.

lukichev@itmo.ru

Alexander Mamatov

Assist. Prof.

amamatov@itmo.ru

Permanent magnet synchronous motors

Laboratory task (10 points max)

Aleksandr Mamatov, PhD,

Assistant Professor, Faculty of Control System and Robotics

amamatov@itmo.ru

Laboratory task «Surface mounted permanent magnet synchronous motor modelling»

Initial data

| | |
|-------------------|---|
| $L_s = L_d = L_q$ | inductance of the stator phase [H] |
| R_s | resistance of the stator phase [Ohm] |
| ϕ_f | rotor flux [Wb] |
| z_p | number of pole pairs |
| J | inertia of the motor's shaft and mechanical load [kg * m ²] |
| n_{rated} | rated speed [rev/min] |
| $t_{startup}$ | time of the PMSM startup from zero to rated speed |

Laboratory task «Surface mounted permanent magnet synchronous motor modelling»

Task 1. Building the SPMSM model based on differential equations

1.1 Build the SPMSM model in d/q axes using following equations

$$v_{ds}^r = R_s i_{ds}^r + \frac{d\lambda_{ds}^r}{dt} - \omega \lambda_{qs}^r$$

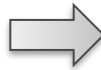
$$v_{qs}^r = R_s i_{qs}^r + \frac{d\lambda_{qs}^r}{dt} + \omega \lambda_{ds}^r$$

$$\lambda_{ds}^r = L_{ds} i_{ds}^r + \phi_f$$

$$\lambda_{qs}^r = L_{qs} i_{qs}^r$$

$$T_e = \frac{p}{2} \frac{3}{2} \left[\phi_f i_{qs}^r + (L_{ds} - L_{qs}) i_{ds}^r i_{qs}^r \right]$$

$$\boxed{\begin{aligned} \phi_f &= const \\ \frac{d\phi_f}{dt} &= 0 \end{aligned}}$$



$$\frac{di_{ds}^r}{dt} = \frac{v_{ds}^r}{L_{ds}} - \frac{R_s i_{ds}^r}{L_{ds}} + \frac{\omega L_{qs} i_{qs}^r}{L_{ds}}$$

$$\frac{di_{qs}^r}{dt} = \frac{v_{qs}^r}{L_{qs}} - \frac{R_s i_{qs}^r}{L_{qs}} - \frac{\omega L_{ds} i_{ds}^r}{L_{qs}} - \frac{\omega \phi_f}{L_{qs}}$$

$$T_e = \frac{p}{2} \frac{3}{2} \left[\phi_f i_{qs}^r + (L_{ds} - L_{qs}) i_{ds}^r i_{qs}^r \right]$$

Laboratory task «Surface mounted permanent magnet synchronous motor modelling»

Inputs of PMSM model

u_a, u_b, u_c abc voltages

ω_m, θ_m velocity and angle of the PMSM shaft

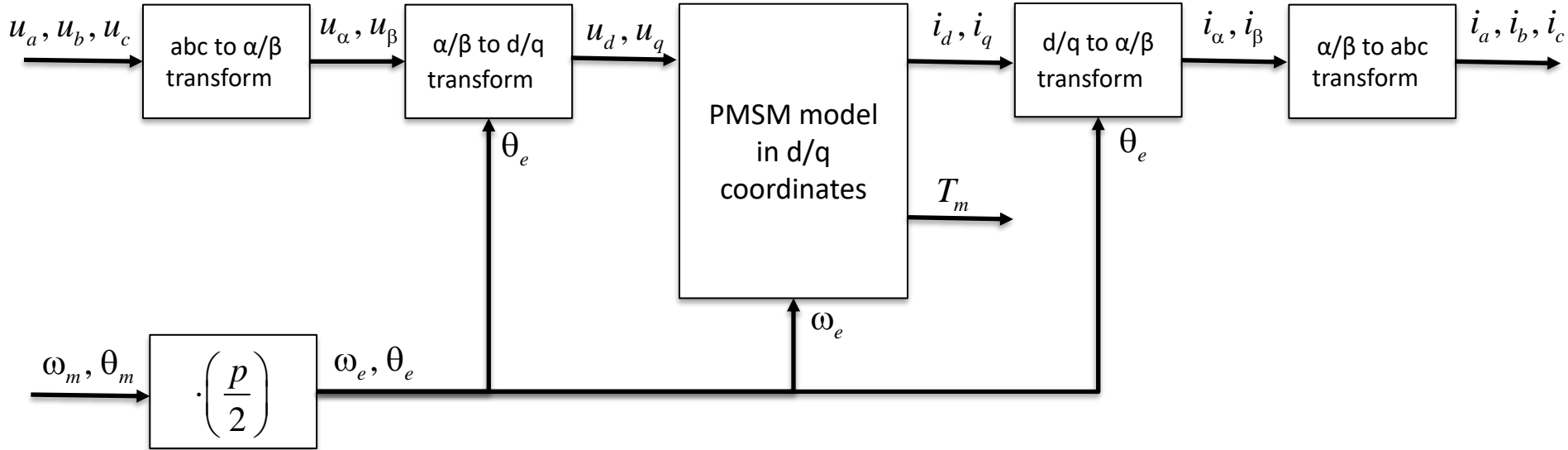
**Use magnitude invariance
reference frame transformation!**

Outputs of PMSM model

i_a, i_b, i_c abc currents

T_m torque of the PMSM

Laboratory task «Surface mounted permanent magnet synchronous motor modelling»



Laboratory task «Surface mounted permanent magnet synchronous motor modelling»

1.2 Build the mechanical part of the model based on follows equations:

$$J \frac{d\omega_m}{dt} = T_m - T_{load}$$

$$\frac{d\theta_m}{dt} = \omega_m$$

Task 2. Build the SPMSM model based on Simscape model of PMSM

MATLAB implementation of Task 1 and Task 2

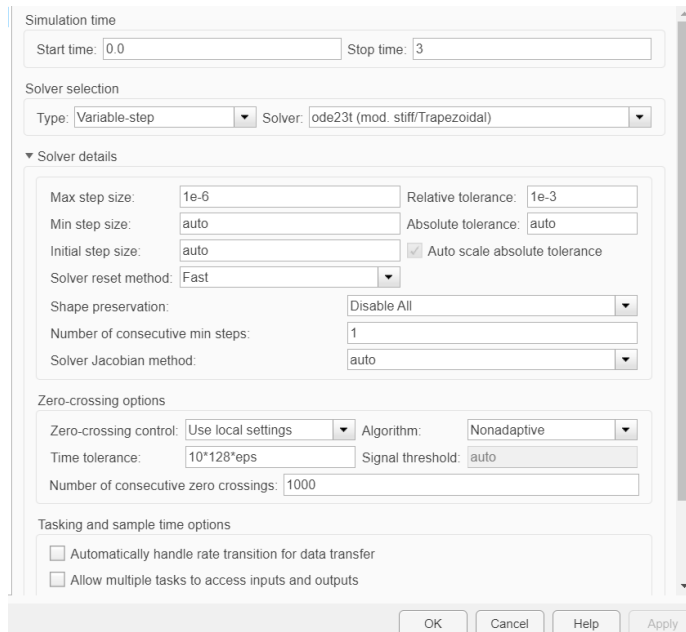
m-script with initial data:

```
1 Rs = 2.2; % resistance
2 Ls = 0.012; % inductance
3 zp = 15; % number of pole pairs
4 psi = 0.1815; % flux
5 J = 0.14; %inertia
6 n Rated = 35; % rated speed rev/min
7 t_startup = 10; % sec
```


MATLAB implementation of Task 1 and Task 2

Open Matlab/Simulink and create new model.

Tune the model configuration.



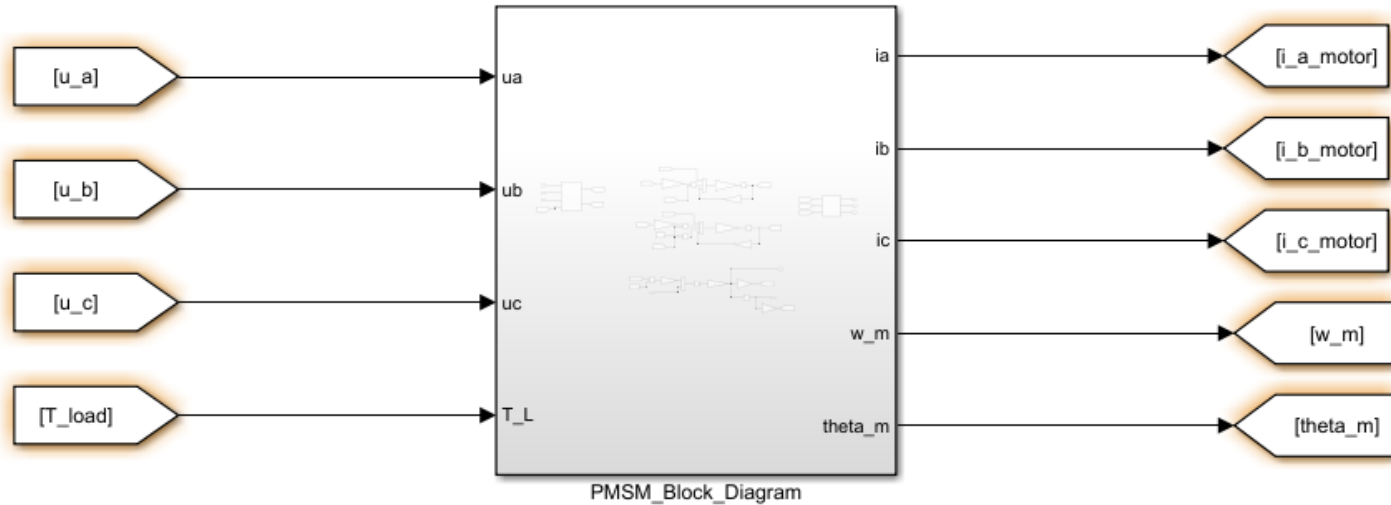
The image shows the MATLAB Solver Configuration dialog box, which is used to tune the model configuration. The dialog is organized into several sections:

- Simulation time:** Start time: 0.0, Stop time: 3.
- Solver selection:** Type: Variable-step, Solver: ode23t (mod. stiff/Trapezoidal).
- Solver details:**
 - Max step size: 1e-6, Relative tolerance: 1e-3.
 - Min step size: auto, Absolute tolerance: auto.
 - Initial step size: auto, ☒ Auto scale absolute tolerance.
 - Solver reset method: Fast.
 - Shape preservation: Disable All.
 - Number of consecutive min steps: 1.
 - Solver Jacobian method: auto.
- Zero-crossing options:**
 - Zero-crossing control: Use local settings, Algorithm: Nonadaptive.
 - Time tolerance: 10*128*eps, Signal threshold: auto.
 - Number of consecutive zero crossings: 1000.
- Tasking and sample time options:**
 - ☐ Automatically handle rate transition for data transfer.
 - ☐ Allow multiple tasks to access inputs and outputs.

At the bottom of the dialog are buttons for OK, Cancel, Help, and Apply.

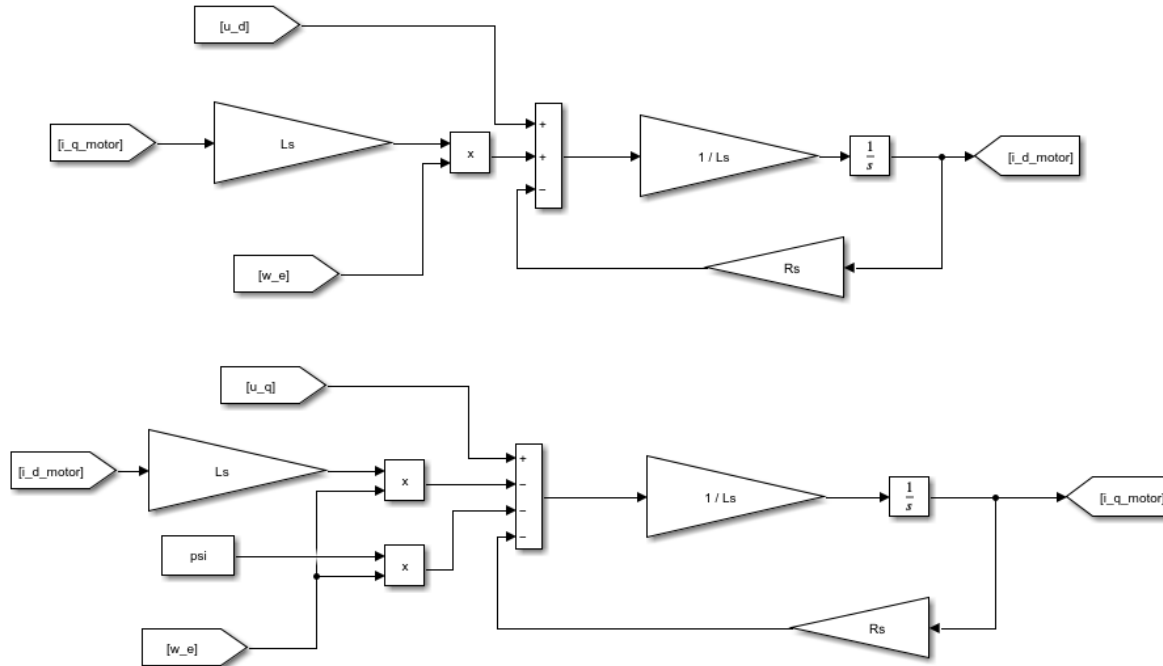
MATLAB implementation of Task 1 and Task 2

The model of SPMSM based on differential equations



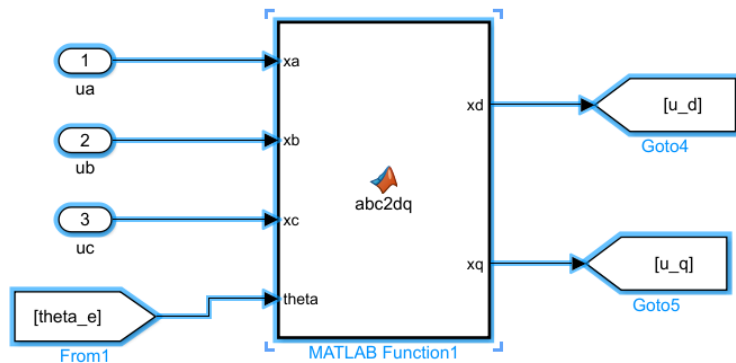
MATLAB implementation of Task 1 and Task 2

Block diagram of dq equations



MATLAB implementation of Task 1 and Task 2

Input Clark & Park transformation:



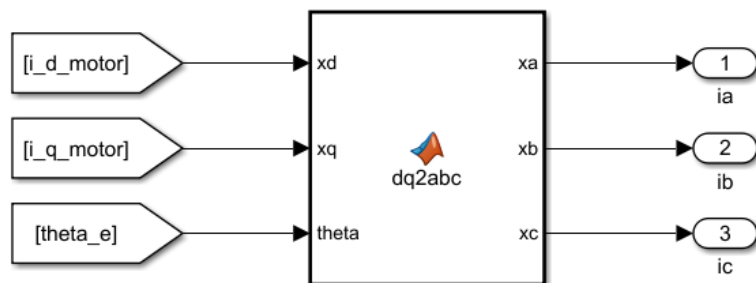
```

1 function [xd, xq] = abc2dq(xa, xb, xc, theta)
2     sin_t = sin(theta);
3     cos_t = cos(theta);
4     x2 = [2/3 -(1/3) -(1/3); 0 1/sqrt(3) -1/sqrt(3)] * [xa xb xc]';
5     x_alpha = x2(1);
6     x_beta = x2(2);
7     xd = x_alpha * cos_t + x_beta * sin_t;
8     xq = -x_alpha * sin_t + x_beta * cos_t;
9 end
10

```

MATLAB implementation of Task 1 and Task 2

Output inverse Park & Clark transformation:



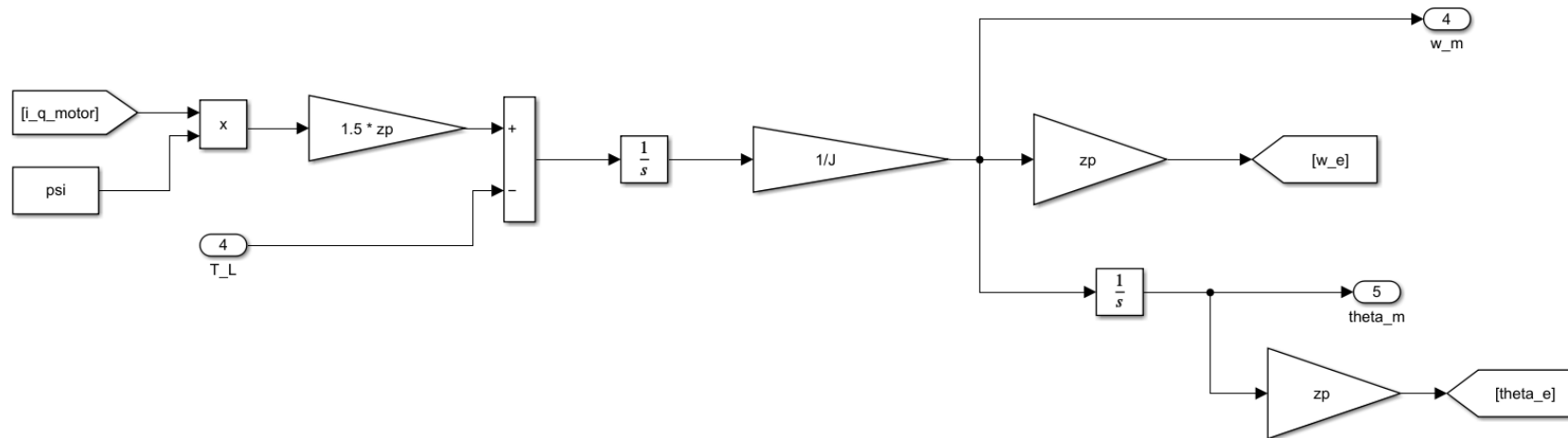
```

1  function [xa, xb, xc] = dq2abc(xd, xq, theta)
2      sin_t = sin(theta);
3      cos_t = cos(theta);
4      x_alpha = xd * cos_t - xq * sin_t;
5      x_beta = xd * sin_t + xq * cos_t;
6      xa = x_alpha;
7      xb = (-0.5) * x_alpha + sqrt(3) / 2 * x_beta;
8      xc = (-0.5) * x_alpha - sqrt(3) / 2 * x_beta;
9  end
10

```

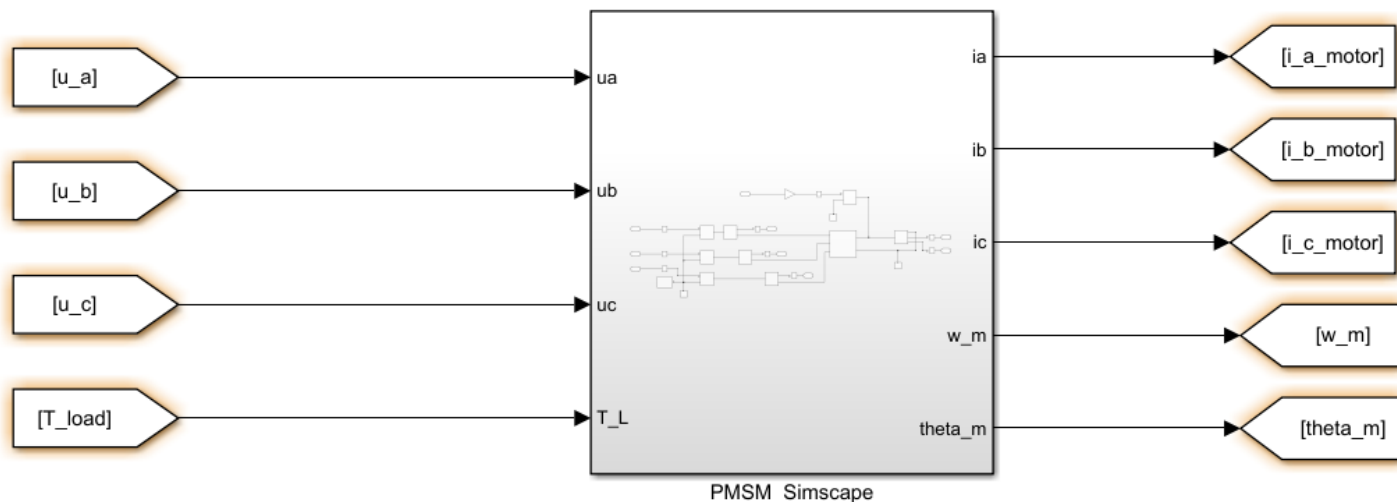
MATLAB implementation of Task 1 and Task 2

Mechanical part:



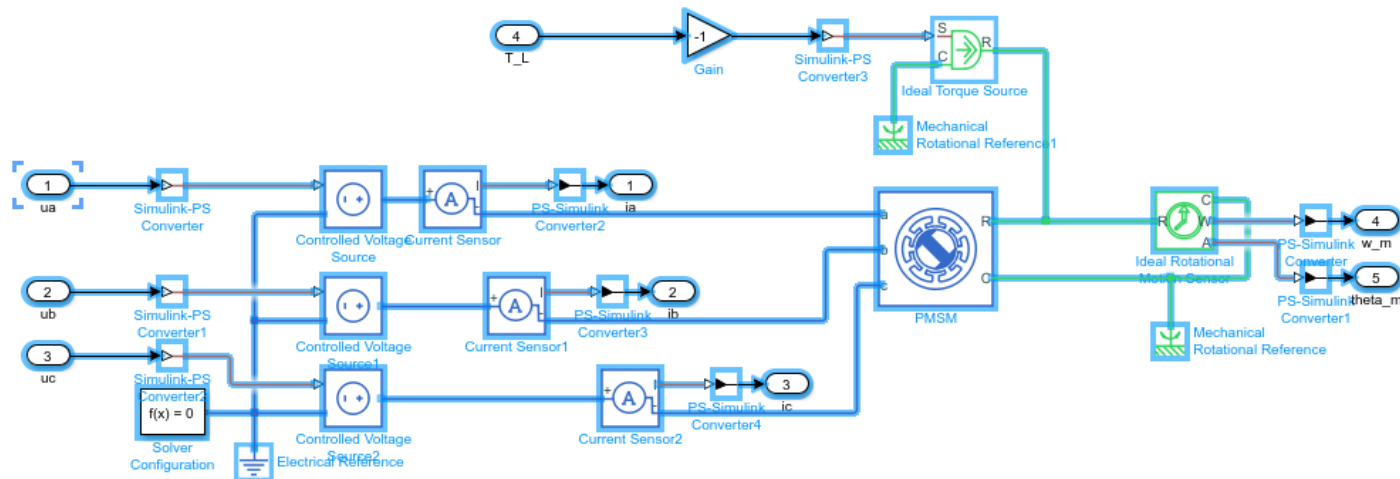
MATLAB implementation of Task 1 and Task 2

The model of SPMSM based on Simscape model



MATLAB implementation of Task 1 and Task 2

The model of SPMSM based on Simscape model



MATLAB implementation of Task 1 and Task 2

The model of SPMSM based on Simscape model

Block Parameters: PMSM

Permanent Magnet Synchronous Machine

This block represents a permanent magnet synchronous machine with sinusoidal flux distribution.

Right-click on the block and select Simscape block choices to access variant implementations of this block.

[Select a predefined parameterization](#)

Settings

Main Iron Losses Mechanical Variables

Electrical connection: Expanded three-phase ports

Winding type: Wye-wound

Modeling fidelity: Constant L_d , L_q , and PM

Number of pole pairs: zp

Permanent magnet flux linkage parameterization: Specify flux linkage

Permanent magnet flux linkage: psi Wb

Stator parameterization: Specify L_d , L_q , and L_0

Stator d-axis inductance, L_d : Ls H

Stator q-axis inductance, L_q : Ls H

Stator resistance per phase, R_s : Rs Ohm

Zero sequence: Exclude

Rotor angle definition: Angle between the a-phase magnetic axis and the d-axis

OK Cancel Help Apply

Block Parameters: PMSM

Permanent Magnet Synchronous Machine

This block represents a permanent magnet synchronous machine with sinusoidal flux distribution.

Right-click on the block and select Simscape block choices to access variant implementations of this block.

[Select a predefined parameterization](#)

Settings

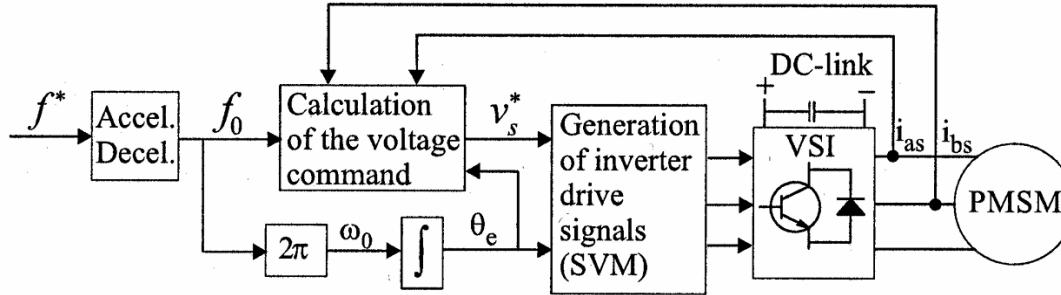
Main Iron Losses Mechanical Variables

Rotor inertia: J kg*m²

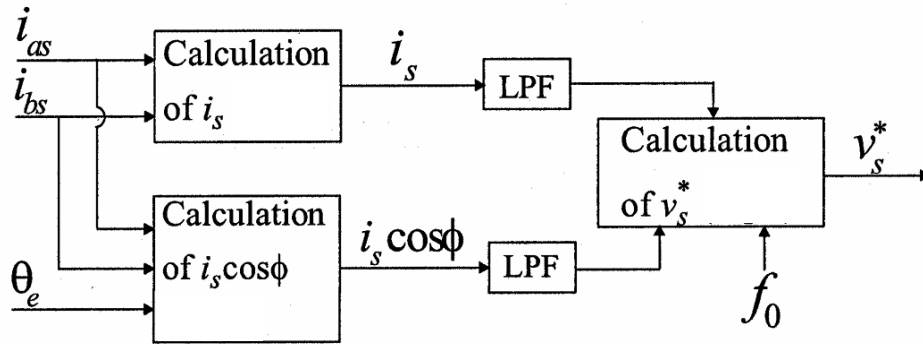
Rotor damping: 0 N*m/(rad/s)

OK Cancel Help Apply

Task 3. Implement the V/f startup strategy for SPMSM

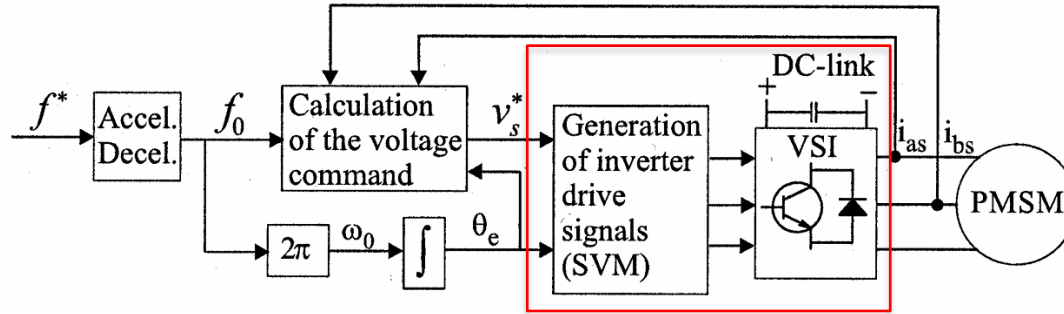


Preliminary drive configuration with voltage control method.



Calculation of the voltage command

3.1 Simplified linear model of the VSI inverter

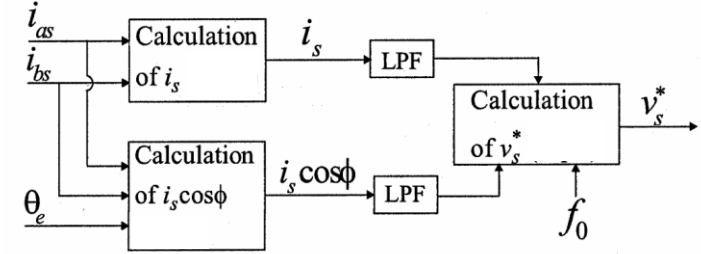
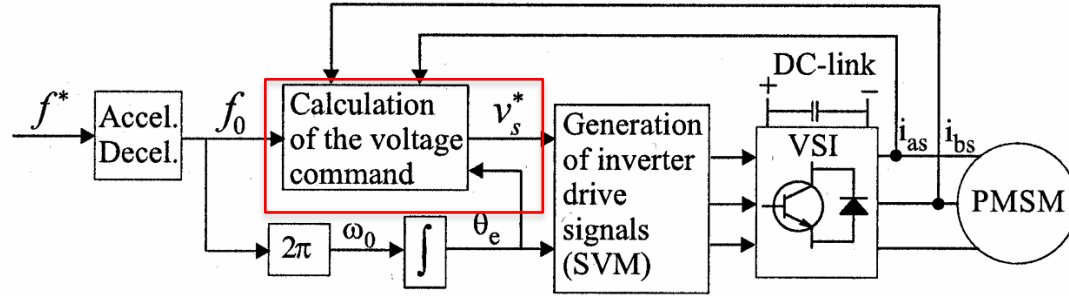


$$u_a = v_s^* \sin(\theta_e)$$

$$u_b = v_s^* \sin\left(\theta_e - \frac{2\pi}{3}\right)$$

$$u_c = v_s^* \sin\left(\theta_e + \frac{2\pi}{3}\right)$$

3.2 Calculation of the voltage command

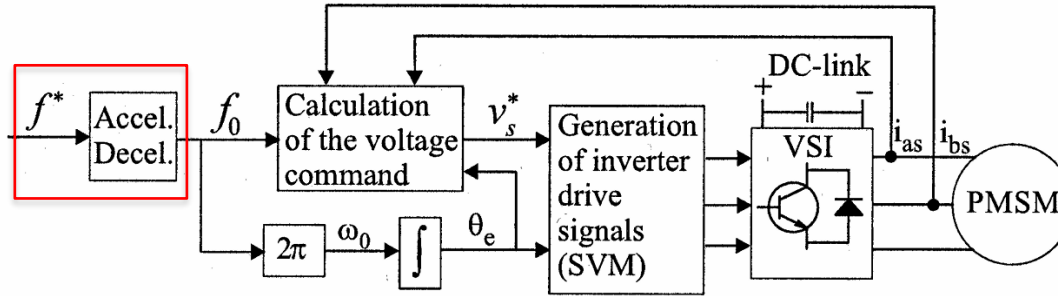


$$i_s \cos \phi = \frac{2}{3} \left[i_{as} \cos \theta_e + i_{bs} \cos \left(\theta_e - \frac{2\pi}{3} \right) - (i_{as} + i_{bs}) \cos \left(\theta_e + \frac{2\pi}{3} \right) \right]$$

$$i_s = \sqrt{\frac{1}{3} (i_{as} + 2i_{bs})^2 + (i_{as})^2}$$

$$v_s^* = R_s i_s \cos \phi + \sqrt{(2\pi f_0 \phi_f)^2 + (R_s i_s \cos \phi)^2 - (R_s i_s)^2}$$

3.3 Generation of the frequency command



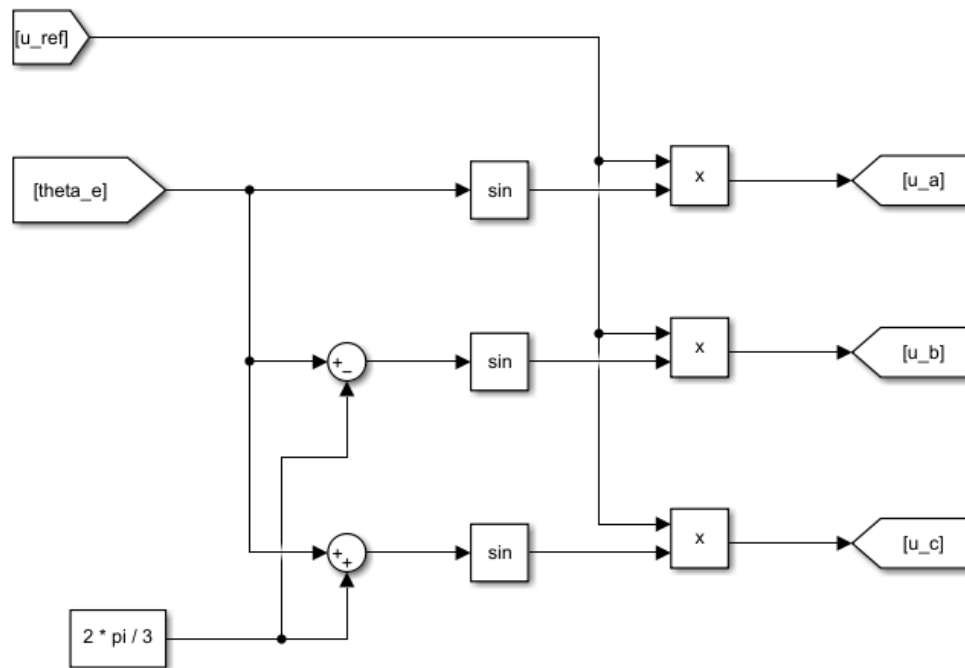
$$f^* = \frac{n_{rated}}{60} z_p$$

$$\varepsilon_{startup} = \frac{f^*}{t_{startup}}$$

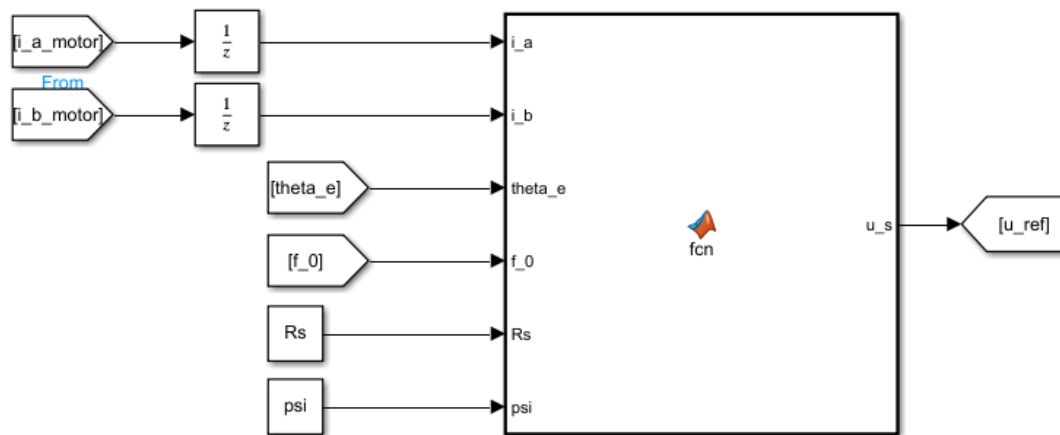
$$\begin{cases} \varepsilon(t) = \varepsilon_{startup}, & t \leq t_{startup} \\ \varepsilon(t) = 0, & t > t_{startup} \end{cases}$$

$$f_0 = \int \varepsilon(t) dt$$

Matlab implementation



Matlab implementation

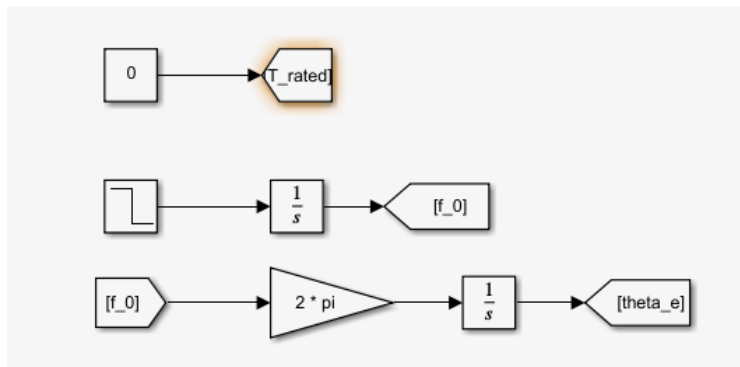


```

1 function u_s = fcn(i_a, i_b, theta_e, f_0, Rs, psi)
2
3     i_s_cos = 2 / 3 * (i_a * cos(theta_e) + i_a * cos(theta_e - 2 * pi / 3) - (i_a + i_b) * cos(theta_e + 2 * pi / 3));
4
5     i_s = sqrt(1 / 3 * (i_a + 2 * i_b)^2 + i_a^2);
6
7     u_s = Rs * i_s_cos + sqrt((2 * pi * f_0 * psi)^2 + (Rs * i_s_cos)^2 - (Rs * i_s)^2);
8
9
10 end
11

```

Matlab implementation



Block Parameters: Step

Step

Output a step.

Main Signal Attributes

Step time:

$t_startup$

Initial value:

$eps_startup$

Final value:

0

Sample time:

0

☒ Interpret vector parameters as 1-D

☒ Enable zero-crossing detection

OK Cancel Help Apply

3.4 Simulations

Simulate the startup process for both of SPMSM models under no load condition.

Simulation time: 11 sec.

Make sure that both models work the same way.

Plot the follows graphs for both models:

- reference frequency and motor velocity in rev/min
- phase currents i_a, i_b, i_c, A
- reference voltage v_s^*, V
- phase voltages v_a, v_b, v_c, V
- dq -voltages, V
- dq -currents, A

Report requirements

Report must contain:

- your name in English, your HDU and ITMO numbers, your photo.
- good resolution screenshots of your models of SPMSM
- good resolution screenshots of your startup scheme
- good resolution figures from simulations (white background, legend, unit of measurements).
- listing of your m-file with calculations.
- conclusion.

Reports should be uploaded to the folder in DingTalk chat.

[HDU-AT3-24] Electrical Machines/Student_works/PMSM_Reports

Report requirements

Deadline for report: 2024/10/22

Penalties:

- inaccurate figures: - 2 points
- skipping the deadline: - 4 points
- no conclusions or inadequate conclusion: - 2 points

IN THE CASE OF PLAGIARISM IN ANY PART OF THE LAB, ALL PARTS OF DC LAB WILL BE EVALUATED BY ZERO POINTS. THE FACT OF PLAGIARISM WILL BE REPORTED TO THE FACULTY.

Thank you!

amamatov@itmo.ru