Table of Contents

1. Vehicle Model i
   1.1. Introduction i
   1.2. Assistance systems i
   1.3. Vehicle modeling ii
   1.4. Torque vectoring xii
   1.5. Electronic Differential System xiii
2. Motor control xvi
   2.1. Dynamic d-q Model xvi
   2.2. Vector or Field-Oriented Control xx
   2.3. Fundamentals of Vector Control xxii
   2.4. Indirect Vector Control xxiii
   2.5. PI Controller xxix
   2.6. Motor parameters xxxiii
   2.7. Simulation Results xxxiv
   2.8. DTC (Direct torque control) xxxiv
   2.9. MRAS control xli
   2.10. Comparison between DTC and FOC xlv
3. Proposed Systems xlv
   3.1. Experimental FOC (DSP model) xlv
   3.2. Current measurements xlviii
   3.3. SPWM Generation xlviii
   3.4. Output results xlix
   3.5. Complete Model li
4. References lix

1. Vehicle Model

1.1. Introduction

Since the 1980s, researchers have studied and developed many advanced chassis control systems, including active steering, direct yaw moment control (DYC), and active roll control. Based on the difference between the anticipated and actual vehicle response, the active steering system adjusts the steering angle in response to driver commands to keep the vehicle traveling in the intended direction. By using differential braking and traction, direct yaw moment control can increase the lateral stability of the vehicle. Individual chassis control systems do, however, have some restrictions. For example, active steering control cannot produce more tire lateral forces in the near-saturation region.

1.2. Assistance systems

Different advanced or simple control systems aid the driver in maintaining control, stability, and maneuverability of the vehicle. These systems increase passenger safety or reduce the amount of effort required by the driver to operate the vehicle. Torque vectoring and electronic stability systems are two examples of assistance systems chosen for this project.

1.2.1. Electronic stability program

It is a control system that stabilizes the vehicle by using the braking system as the control actuator. Depending on whether the vehicle is oversteering or understeering, this system typically applies brakes to one wheel to increase or decrease the yaw rate. The vehicle should maintain stability using this control action in all urgent circumstances.

1.2.2. Torque vectoring system

It is a control system used to achieve the same goals as the ESP. However, the torque vectoring system has a different application than the ESP. It is employed to alter the lateral vehicle dynamics and enhance the vehicle's stability and maneuverability. In most cases, these situations are not life or death. On the other hand, the electronic stability system regulates the stability of the vehicle in hazardous circumstances where serious damage may result. Torque vectoring should be disabled if the ESP system begins to control the vehicle. Additional yaw moment produced by the variation in the vehicle's torque distribution stabilizes or accelerates the yaw moment. The difference between the generated yaw moment and rear wheel torque is shown in Figure 1.1, This yaw moment then forces the vehicle dynamics to turn more, which results in an increment of the vehicle yaw rate.

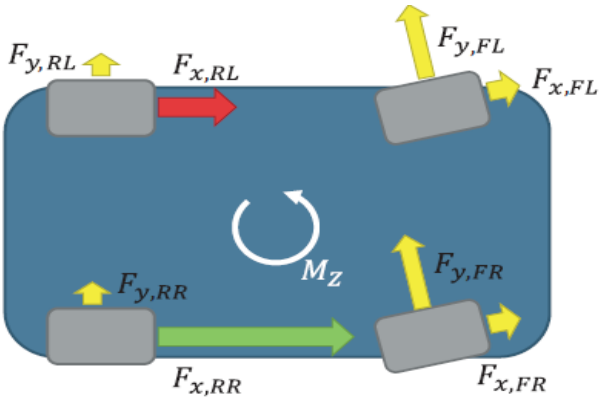


Figure.1.1:Created additional yaw

1.3. Vehicle modeling

The simulation and control algorithm development discussed in the project is aided by the vehicle dynamics discussed in this chapter. Several aspects of the vehicle dynamics are simplified during the system description. The planar model of the vehicle is used to develop the torque vectoring system and simulate cornering. The vehicle body's roll motion and vertical movement are excluded.

1.3.1. Tire model

The most important inputs for tire models that will be later taken into consideration are side-slip angle α, longitudinal slip λ, and normal load Fz of a specific tire. side-slip angle α and longitudinal slip λ are defined as follows:

$$\lambda = \frac{|v_x - v_c|}{v_x} \qquad (1.1)$$

$$\alpha = \arctan\left(\frac{v_y}{v_x}\right) \qquad (1.2)$$

where v_c is the circumferential velocity of the tire, v_x is the longitudinal velocity of the tire center, and v_y is lateral tire velocity of the tire center, both with respect to the ground.

The slip curve is a common way to express the longitudinal and lateral forces that the tire transfers, as well as how these forces depend on the longitudinal slip and side-slip angle, respectively. The longitudinal acceleration is taken to be zero and the vehicle's forward speed to be constant. As a result, only lateral tire forces and side-slip angles concerning a lateral force slip characteristic are considered. An example of this slip curve is shown in figure1.2

Nominal cornering stiffness $C_{\alpha 0}$ is the slope of the characteristic at zero side-slip angles. The characteristic is linear for small side-slip angles α, and the side force Fy generated is equal to the side-slip value multiplied by this nominal cornering stiffness

coefficient. Although the generated side force is linear up until the slip curve reaches the maximum of the friction coefficient μ_{max}, as the side-slip angle increases, the tire begins to become overloaded. The tire is unable to transmit higher forces Fy as the side-slip angle is increased further.

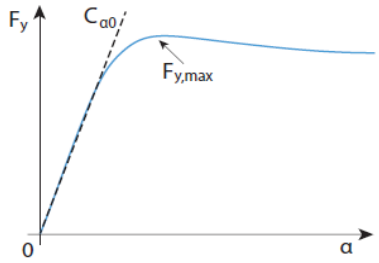


Figure 1.2 An example of typical side-slip characteristics for lateral motion

1.3.2. Kinematic vehicle model

The kinematic vehicle model is derived from a simple vehicle scheme, with the corner's center falling on the line defined by the vehicle's rear axle. Each point moves in a direction that is always perpendicular to the line that connects it to the corner's center.

The instantaneous cornering radius R is determined by finding the intersection of the front tire's normal to the x-axis and the vehicle's rear axle. The vehicle states can be calculated from figure1.3 as

$$r = \frac{v}{R} = \frac{v}{l_r}\tan\beta \qquad (1.3)$$

$$\beta = \tan^{-1}\left(\frac{l_r}{l_r + l_f}\tan\delta_f\right) \qquad (1.4)$$

where l_f and l_r are the distances from the car's center of gravity to its front and rear tires, respectively, v is the vehicle speed, R is the cornering radius, $r(\text{or}\dot{\psi})$ is the vehicle yaw rate, and β is the side-slip angle.

The kinematic model can be used only at low speeds, this model will be the reference model for the torque vectoring system.

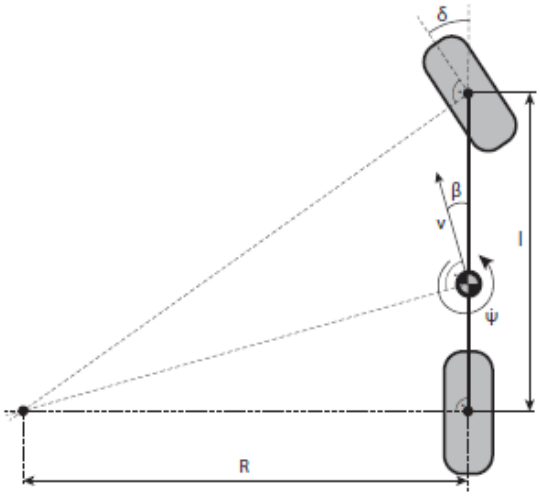


Figure 1.3 Kinematic vehicle model

1.3.3. Single-track vehicle model

To ignore load, transfer during vehicle motion, the center of gravity of the vehicle is projected into the plane of the surface. Thus, to accurately estimate the current vehicle state, only one rotatory and two translatory degrees of freedom are needed.

First, it is necessary to define the vehicle coordinate system (see Fig. 1.4). From the perspective of the driver, the vehicle's x and y axes point from the center of gravity towards the front and left sides, respectively. The z-axis then points upward to conform to the widely accepted right-handed coordinate system.

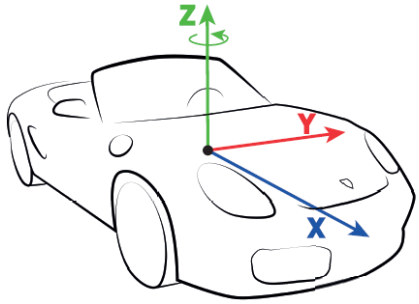


Figure 1.4:Vehicle coordinate system

Figure 1.5 introduces the single-track vehicle model for planar vehicle motion.

The number of tires is the primary distinction between the actual vehicle and the proposed scheme. One tire, which is positioned in the center of the front axis, is used in place of the left and right front tires. It is necessary to increase the cornering stiffness coefficient to account for the impact of the two original tires. On the vehicle's rear axle, the same idea is applied.

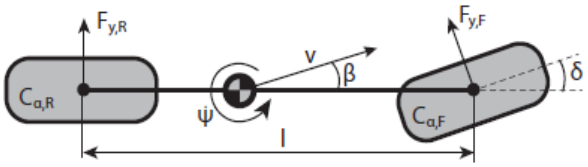


Figure 1.5 single track model

Figure 1.5 shows a vehicle moving at velocity v. The vehicle side-slip angle β is defined as the angle between the vehicle's x-axis and the velocity vector.

$$\beta = \tan^{-1}\left(\frac{v_y}{v_x}\right) \qquad (1.5)$$

where v_x and v_y are the vehicle velocities in the x and y direction of the vehicle coordinate frame respectively.

Vehicle yaw ψ is the angle formed by the fixed global axis x_0 and the longitudinal vehicle axis x. If the motion of the vehicle is linear, the vehicle acceleration v′ has the same direction as the vehicle velocity v. When a vehicle travels along a curved track, we notice centripetal acceleration, which is defined as a_c

$$a_c = \frac{v^2}{R} = v(\dot{\beta} + r) \qquad (1.6)$$

where v is the vehicle's instantaneous speed, R is the track's instantaneous radius,$\dot{\beta}$ is the angular rate of the vehicle's side slip, and r is the rate at which the vehicle yaws with respect to its z-axis. Only the second part of the equation, which includes the vehicle side-slip angle rate and yaw rate, is typically used to calculate the vehicle side acceleration because the instantaneous radius of the motion is frequently unknown and changes frequently.

By bringing all forces in the x, y, and z directions of the vehicle into equilibrium as well as all moments about the z-axis, it is possible to directly derive the differential equations of motion for the vehicle depicted in Figure 1.5. The forces of aerodynamics are neglected.

$$-m\dot{v}\cos(\beta) + mv(\dot{\beta} + r)\sin(\beta) - F_{y,F}\sin(\delta) + F_{x,F}\cos(\delta) + F_{x,R} = 0 \qquad (1.7)$$

$$-m\dot{v}\sin(\beta) - mv(\dot{\beta} + r)\cos(\beta) + F_{y,F}\cos(\delta) + F_{x,F}\sin(\delta) + F_{y,R} = 0 \qquad (1.8)$$

$$-I_z\dot{r} + F_{y,F}l_f\cos(\delta) - F_{y,R}l_r + F_{x,F}l_f\sin(\delta) = 0 \qquad (1.9)$$

m is the mass of the vehicle, v is the velocity of the vehicle, r is the yaw rate of the vehicle, I_z is the moment of inertia about the z-axis, l_f, and l_r are the distances of the center of the front and rear tire from the vehicle gravity center respectively, β is the slide-slip angle of the vehicle, δ is the front tire steering angle, $F_{x,F}$and $F_{x,R}$ are longitudinal forces of the front and rear tire respectively, $F_{y,F}$ and $F_{y,R}$ are lateral forces of the front

and rear tire respectively. The change of the side-slip angle $\dot{\beta}$ is very small compared to the yaw rate r

Equations of the chosen tire model define the tire forces $F_{y,F}$ and $F_{y,R}$. Side-slip angles of tires are significantly influenced by tire position and steering angle, which are defined as:

$$\alpha_F = \delta - arctan\left(\frac{v\,sin(\beta) + l_f r}{vcos(\beta)}\right) \quad (1.10)$$

$$\alpha_R = -arctan\left(\frac{v\,sin(\beta) - l_r r}{vcos(\beta)}\right) \quad (1.11)$$

where α_F and α_R are tire side-slip angles of the front and the rear tire respectively.

The previous equations can be approximated for small steering angles as:

$$\alpha_F = \delta - \beta - \frac{l_f r}{v_x} \quad (1.12)$$

$$\alpha_R = -\beta + \frac{l_r r}{v_x} \quad (1.13)$$

1.3.4. Linear vehicle model with constant velocity

The single-track vehicle model may be linearized using equations if the front wheel steering angle and side-slip angle are both less than 10°.

$$\mathrm{Sin(x)} \approx \mathrm{x}$$

$$\mathrm{Cos(x)} \approx 1$$

The vehicle differential equations of motion can be reformulated to

$$-m\dot{v} + F_{x,F} + F_{x,R} = 0 \quad (1.14)$$

$$-m\dot{v}\left(\dot{\beta} + r\right) + F_{y,F} + +F_{y,R} = 0 \quad (1.15)$$

$$-I_z\dot{r} + F_{y,F}l_f - F_{y,R}l_r = 0 \quad (1.16)$$

These equations establish the front and rear tire's side-slip angles. The linear model discussed above may be used to determine the lateral tire forces $F_{y,F}$ and $F_{y,R}$.

During the cornering maneuver with constant velocity, the acceleration of the vehicle $\dot{v}$ is considered to be equal to zero. The following equations are created by substituting side forces $F_{y,F}$ and $F_{y,R}$into the vehicle differential equations.

$$-m\dot{v}\dot{\beta} - m\dot{v}r + C_f\left(\delta - \beta - \frac{l_f r}{v}\right) + C_r\left(-\beta + \frac{l_r r}{v}\right) = 0 \quad (1.17)$$

$$-I_z\dot{r}+C_f\left(\delta-\beta-\frac{l_f r}{v}\right)l_f - C_r\left(-\beta+\frac{l_r r}{v}\right)l_r = 0 \qquad (1.18)$$

The linearized differential equations can be modified after small manipulation into the following state space model:

$$\begin{bmatrix}\dot{\beta}\\ \dot{r}\end{bmatrix} = \begin{bmatrix} -\frac{C_f+C_r}{mv} & -\left(1+\frac{C_f l_f - C_r l_r}{mv^2}\right) \\ -\frac{C_f l_f - C_r l_r}{I_z} & -\frac{C_f l_f^2 + C_r l_r^2}{I_z} \end{bmatrix}\begin{bmatrix}\beta\\ r\end{bmatrix} + \begin{bmatrix}\frac{C_f}{mv}\\ \frac{C_f l_f}{I_z}\end{bmatrix}\delta \qquad (1.19)$$

where the vehicle states are represented by the vehicle side-slip angle β and the vehicle yaw rate r.

1.3.5. Final Vehicle Model

Numerous factors influence the states of the vehicle but aren't shown in the current models such as resistive forces, and the aligning torques for each axle.

1.3.5.1. Resistive forces and load torque calculations

The total force required for driving a vehicle is calculated below:

$$Ftotal = Frolling + Fgradient + Faerodynamic\ drag$$

Where $Frolling$ is the force due to Rolling Resistance, $Fgradient$ is the force due to Gradient Resistance, $Faero\ is\ the$ force due to aerodynamic drag, $Ftotal$ is the total tractive force that the output of motor must overcome, to move the vehicle.

- ROLLING RESISTANCE

Rolling resistance is the resistance offered to the vehicle due to the contact of tires with the road. The formula for calculating force due to rolling resistance is given by equation.

$$Frolling = Crr * M * g$$

Where, Crr= coefficient of rolling resistance, M= mass in kg, g= acceleration due to gravity= 9.81 m/s^2

The rolling resistance coefficient is a dimensionless value that depends on the road surface and the type of tire.

- GRADIENT RESISTANCE

Gradient resistance of the vehicle is the resistance offered to the vehicle while climbing a hill or flyover or while traveling on a downward slope. The angle between the ground and the slope of the path is represented as α.

The formula for calculating the gradient resistance is given by the equation:

$$F_{gradient} = \pm \mathrm{M} * \mathrm{g} * \sin(\alpha)$$

- AERODYNAMIC DRAG

Aerodynamic drag is the resistive force offered due to the viscous force acting on the vehicle. It is largely determined by the shape of the vehicle. The formula for calculating the aerodynamic drag is given by the equation:

$$\mathrm{F_{aero}} = \frac{1}{2}\rho \mathrm{v^2 S C_d} \qquad (1.20)$$

where ρ is the air density, $\mathrm{C_d}$ is the dimensionless drag factor that depends on the vehicle shape, and S is the projected frontal area of the vehicle.

These are the three main forces that act on the vehicle when it travels at a constant speed. While accelerating and decelerating the effect of force due to inertia also acts.

$$T_{load_{total}} = F_{total} * \text{Wheel Radius}$$

The losses due transmission of power to the wheel must be included by considering the efficiency of the transmission gear system.

Motor torque is the torque produced by the electric motor driving the vehicle, which can be obtained from the motor specifications or calculations based on the motor's power and speed.

1.3.5.2. Cornering Forces Acting on a Vehicle:

Forces acting on the vehicle are explained in [2], the assumption of a small sideslip angle β and the subsequent linearization and uncoupling between lateral and longitudinal behavior allow one to use the same values of the forces on the ground, so a mono track system can be used.

Only the resultants of the side force Fy and the yaw moment Mz need to be computed because the lateral behavior is decoupled from the longitudinal one.

The expression of the total lateral forces can be expressed as the linear equation:

$$F_y = Y_\beta \beta + Y_r r + Y_\delta \delta + F_{ye} \qquad (1.21)$$

Where:

$$\begin{cases} Y_\beta = -(C_f + C_r + \frac{1}{2}\rho v^2 S C_{y,\beta}) \\ Y_r = -\frac{1}{v}(l_f C_f - l_r C_r) \\ Y_\delta = C_f \end{cases}$$

Similar to what is seen for the cornering forces, the linearized expression for the yawing moments is:

$$M_z = N_\beta \beta + N_r r + N_\delta \delta + M_{ze} \quad (1.22)$$

Where:

$$\begin{cases} N_\beta = -l_f C_f + l_r C_r + M_{z,1\alpha} + M_{z,2\alpha} + \frac{1}{2}\rho v^2 S C_{Mz,\beta} \\ Y_r = \frac{1}{v}(-l_f^2 C_f - l_r^2 C_r + M_{z,1\alpha} l_f + M_{z,2\alpha} l_r) \\ N_\delta = C_f l_f - M_{z,1\alpha} \end{cases}$$

Note: $M_{z,1\alpha}$ and $M_{z,2\alpha}$ are the aligning torques, M_{ze} is the external yaw moment, and F_{ye} is the external force(neglected).

1.3.5.3. Derivatives of stability:

The terms $Y_\beta, Y_r, Y_\delta, N_\beta, N_r$ and N_δ are nothing but the derivatives of the force with respect to the variables β, r, and δ, They are usually referred to as derivatives of stability. N_r is sometimes referred to as yaw damping, as it is a factor that, multiplied by an angular velocity, yields a moment, like a damping coefficient. In a simplified study of the handling of road vehicles, aerodynamic forces are usually neglected, as is the interaction between the longitudinal and transversal forces of the tires. In these conditions, $Y_\beta, Y_\delta, N_\beta$ and N_δ are constant while, Y_r and, N_r are proportional to 1/v. They are strongly influenced by the load and road conditions through the cornering stiffness of the tires. The final expression of the equations of motion:

$$\begin{cases} mv(\dot{\beta} + \dot{\psi}) + m\,\dot{v}\beta = Y_\beta \beta + Y_r r + Y_\delta \delta + F_{ye} & (1.23) \\ I_z \ddot{\psi} = N_\beta \beta + N_r r + N_\delta \delta + M_{ze} & (1.24) \end{cases}$$

The steering angle, along with the external force and moment F_{ye} and M_{ze}, can be thought of as an input to the system. This strategy is frequently known as the "locked controls" behavior.

β and r can be considered as state variables can be as a state equation:

$$\dot{z} = Az + B_c u_c + B_e u_e \quad (1.25)$$

where the state and input vectors are:

$$z = \begin{bmatrix} \beta \\ r \end{bmatrix}, \quad u_c = \delta, \quad u_e = \begin{bmatrix} F_{ye} \\ M_{ze} \end{bmatrix}$$

The dynamic matrix is:

$$A = \begin{bmatrix} \frac{Y_\beta}{mv} - \frac{\dot{v}}{v} & \frac{Y_r}{mv} - 1 \\ \frac{N_\beta}{I_z} & \frac{N_r}{I_z} \end{bmatrix}$$

The input gain matrices are:

$$B_c = \begin{bmatrix} \frac{Y_\delta}{mv} \\ \frac{N_\delta}{I_z} \end{bmatrix}, \qquad B_e = \begin{bmatrix} \frac{1}{mv} & 0 \\ 0 & \frac{1}{I_z} \end{bmatrix}$$

The block diagram corresponds to the state equations shown in Figure 1.6, This is the vehicle model considered for applying torque vectoring in section 1.4.

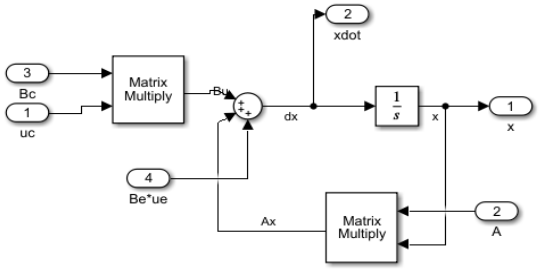


Figure 1.6: block diagram for vehicle handling model

By neglecting F_{ye} and M_{ze}, the state equation will be:

$$\dot{z} = Az + B_c u_c \qquad (1.26)$$

The actual vehicle model will be represented using eq1.26 (see Figure 1.7).

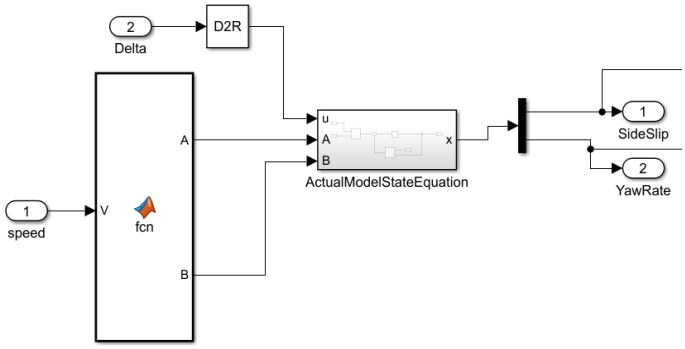


Figure 1.7: actual vehicle model

An interesting analogy exists. It is simple to obtain r from the first equation and substitute it into the second, which becomes a second-order differential equation in β if the speed is maintained constant such that the stability derivatives are constant in time. A

similar equation in r is obtained by solving the second equation in β and substituting it in the first one. The outcome is:

$$P\ddot{\beta}+Q\dot{\beta}+U\beta= S'\delta+T'\dot{\delta}- N_rF_{ye}+I_zF_{ye}-(mv-Y_r)M_{ze} \quad (1.27)$$

$$P\ddot{r}+Q\dot{r}+Ur= S''\delta+T''\dot{\delta}+ N_\beta F_{ye}-Y_\beta M_{ze}+mvM_{ze} \quad (1.28)$$

where

$$\begin{cases} P = J_z mV \\ Q = -J_z Y_b - mVN_r \\ U = N_\beta (mV - Y_r) + N_r Y_\beta \end{cases} \qquad \begin{cases} S' = -N_\delta (mV - Y_r) - N_r Y_\delta \\ S'' = Y_\delta N_\beta - N_\delta Y_\beta \\ T' = J_z Y_\delta \\ T'' = mVN_\delta \ . \end{cases}$$

Each of equations (1.27) and (1.28) is sufficient for the study of the dynamic behavior of the vehicle.

The equations are the same as the spring mass-damper system's equation of motion. Thus, a rigid motor vehicle's linearized behavior at constant speed is the same as a mass P suspended from a spring with stiffness U and a damper with damping coefficient Q, excited by the various forcing functions (the command δ and the external disturbances) listed above.

1.3.6. Vehicle parameters

Table 1 shows the parameters for the vehicle used for the simulation and implementation of the control algorithm for torque vectoring and Electronic differential systems.

M (kg)	G (m/s^2)	L (m)	Lf (m)	Lr (m)	Cf (N rad^{-1})	Cr (N rad^{-1})	W (Track Width) (m)
1500	9.81	3	1.3	1.7	67369	63411	1.56
S (m^2)	$C_{Y,\beta}$	$C_{Mz,\beta}$	Mzfa(alignment torque1)	Mzra(alignment torque2)	ρ (air density)	Iz (kg m^{-2})	
1.7	-2.2	0.6	2010	1366	1.225	2000	

Table 1: vehicle parameters

1.3.7. Vehicle Models comparison

Figure1.8 shows a comparison between the states of the discussed vehicle models, the comparison shows that the vehicle yaw rate in the kinematic and the linear vehicle model with constant velocity is larger than that of the actual model due to the presence of the resistive forces in the actual model.

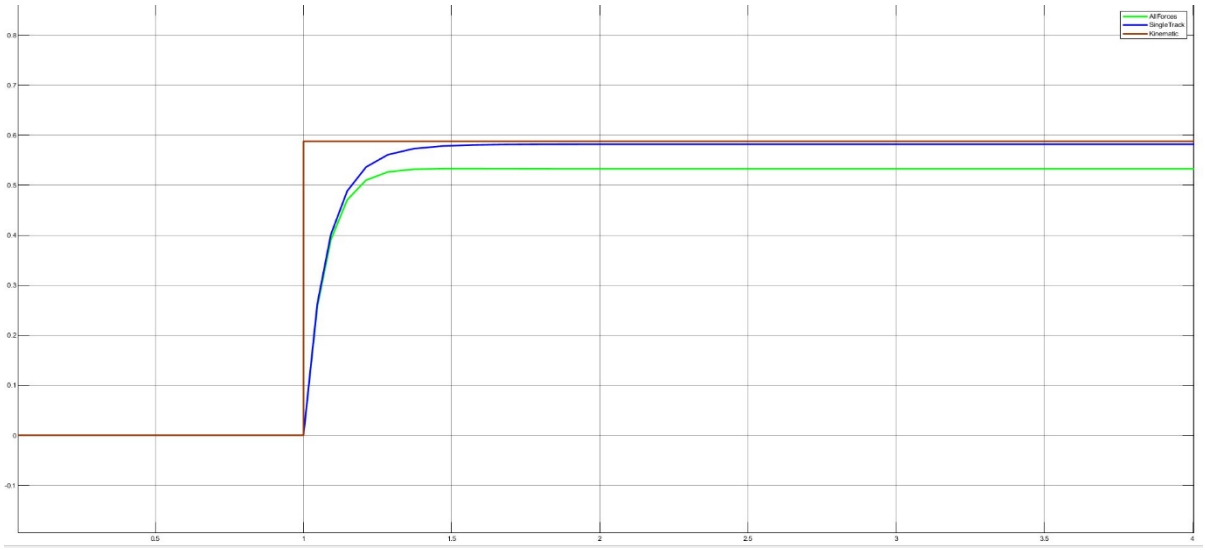


Figure1.8:Yaw rate of different vehicle models

1.4.Torque vectoring

The main idea of the torque vectoring described in section 1.2 is to modify the vehicle dynamics. The goal is to eliminate oversteer or understeer to achieve neutral vehicle behavior and improve vehicle stability. Due to the presence of factors such as aerodynamic drag and alignment torques, the yaw rate value will differ from the ideal behavior. So, an external yaw moment is required to compensate for that error.

The torque vectoring system is represented as a difference between the torques on the right and left electric motor of the vehicle. This difference of torques creates the torque vectoring yaw moment M_{ze} about the z-axis of the vehicle.

1.4.1. Feedforward Control System

Figure1.9 shows the simple feedforward controller used in the project where the steering angle and vehicle speeds are inputs and the torque difference is the controlled variable, the yaw moment generator with the input yaw rate value from the reference (Kinematic)model using eq1.28. The generated yaw moment is then converted to the torque difference according to:

$$\Delta T_r = -\Delta T_l = \frac{M_{ze} d}{2w} \qquad (1.29)$$

where d is the rear wheel diameter, w is the wheelbase, ΔT_r & ΔT_l are the torque differences for the right and left wheels respectively.

The requested torques from the motor are:

$$T_{Qr}=\Delta T_r+\frac{T_{req}}{2}, \qquad T_{Ql}=\Delta T_l+\frac{T_{req}}{2} \tag{1.30}$$

where T_{Qr} and T_{Ql} are the requested torques from the right and left motor respectively, and T_{req} is the required torque before steering.

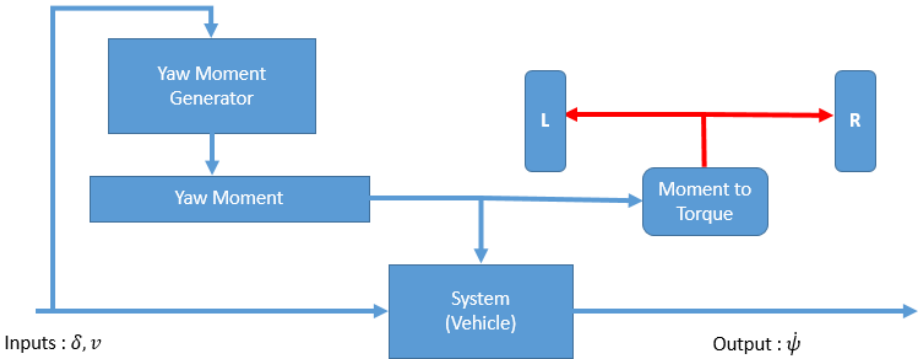


Figure 1.9:Torque vectoring feed forward controller

1.5.Electronic Differential System

1.5.1. Introduction

Ackermann steering was first invented by Georg Lankensperger and was adopted by Rudolph Ackermann for his horse-drawn carriages. He kept two wheels at different points connected using a tie rod to make the two wheels turn at different angles. So that one wheel can turn independently of the other wheel. Also, the linkages for this steering system resemble the shape of a trapezium with one fixed link and three moving links. This is Ackermann's principle of steering, which makes the car turn without any slip.

There is a fixed turning radius for every vehicle, which is the space it requires to complete a circle. The radius is calculated from an imaginary center point called the center of the turning radius. The inner wheel has to take a shorter turn, and the outer wheel has to take a long turn when compared to the inner wheel. In this case, if both wheels are turned at the same angle, there may be a possibility of slipping.

First to understand how the wheel works, a wheel has two different types of velocity. One is the rotational velocity, and the other is the translational velocity. They both have equal magnitude but are opposite in direction; hence they cancel each other. When the wheel has turned, the direction of the rotational velocity changes and there may be a possibility of slipping. To avoid that slip, the direction of the translational velocity has to be changed. But there arises an issue. While turning your car, the inner wheel travels a small distance and needs less translational velocity compared to the outer wheel.

So, both wheels must have different translational velocities and different directions for rotational velocity. This difficulty could be overcome by having different steering angles for two front wheels.

1.5.2. Electronic differential

The electric vehicle is a road vehicle, based on electric propulsion. The electric propulsion system is the heart of the new generation of EVs. It consists of a motor drive, transmission device, and wheels. The motor drive consists of the electric motor, power converter, and electronic controller, which are the core of the EV propulsion system.

Traditional traction systems use a mechanical differential to transfer the propulsion force from the internal combustion engine to the wheels. This mechanism consists of several gearings that essentially provide the same torque to both traction wheels while allowing for a range of speed values. This traction system presents losses due to friction and cannot independently control torque in each wheel. Inversely, an electronic differential avoids such losses while optimizing the profitability of the device. It also allows stronger control of vehicle traction. The figure shows the adopted scheme, this meets the EV requirements and enables replacing the mechanical differential. The traction motors are controlled by DTC (Direct Torque Control) or FOC (Field Oriented Control) through two independent inverters. The motors used are three-phase induction motors. During turns, the electronic differential must account for the speed differential between the two wheels. The system determines the necessary inner and outer wheel speeds.

When the two wheels are separately controlled by two induction motors using the vehicle speed and steering angle as input data.

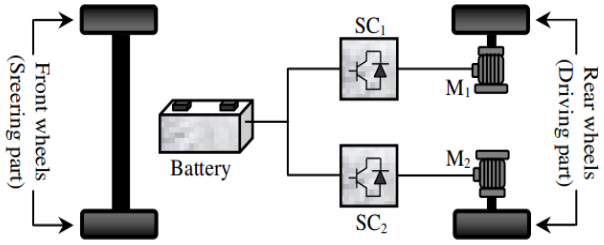


Figure 1.10 vehicle structure with 2 independent rear motors

The differential system allows the traction wheels to spin at different speeds. When the vehicle follows a curve, this condition is required because the external traction wheel moves describing a circumference whose radius is greater than that of the internal traction wheel and hence, its speed must be higher. the basic idea behind mechanical differential mechanisms is to Apply the identical torque to both traction wheels and let each wheel's speed adopt the necessary value to balance the opposing torque. This task is easily achievable through torque control of the induction motors.

Considering that a vehicle does a curve maneuver at a low speed, neglecting consequence lateral forces and slipping over the traction wheels, the geometric model proposed by Ackerman and Jeantad shown in figure1.10 can be used.

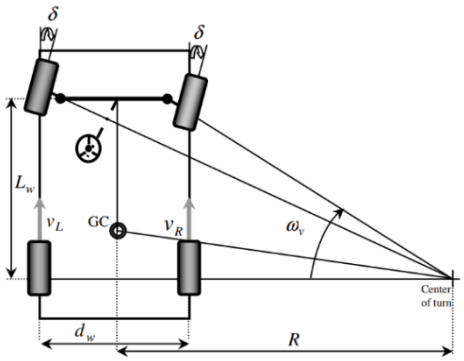


Figure 1.11 Ackerman geometry model

Using it to determine the rotation radius (R) from the steering angle (δ) and in turn the angular speed values that each traction wheel must adopt ($\omega_{r_{-L}}, \omega_{r_{-R}}$).the linear speed of each wheel is expressed as a function of the vehicle speed and the radius of the curve, equations (1.31) and (1.32)

$$V_L = \omega_v \left(R + \frac{d_w}{2}\right) \qquad (1.31)$$

$$V_R = \omega_v \left(R + \frac{d_w}{2}\right) \qquad (1.32)$$

The radius of curve depends on the wheelbase and steering angle:

$$R = \frac{L_w}{tan\,\delta} \qquad (1.33)$$

Substituting (1.33) in equations (1.31) and (1.32), we obtain the angular speed in each wheel drive equation (1.34) and (1.35):

$$\omega_{r_{-L}} = \frac{L_w + \frac{dw}{2} * tan\,\delta}{L_w}\omega_v \qquad (1.34) \qquad \omega_{r_{-R}} = \frac{L_w - \frac{dw}{2} \times \tan\delta}{L_w}\omega_v \qquad (1.35)$$

The difference between the angular speeds of the wheel drives is expressed by equation (1.36).

$$\Delta\omega = \omega_{r_{-L}} - \omega_{r_{-R}} = \frac{d_w * tan\,\delta}{L_w}\omega_v \qquad (1.36)$$

The signal of the steering angle indicates the curve direction.

$\delta > 0$ ⟹ Turn right.

$\delta = 0$ ⟹ Straight ahead

$\delta < 0$ ⟹ Turn left.

A steering angle is applied to the wheels by the driver when the car reaches the start of a curve. The electronic differential instantaneously affects the two motors, decelerating the inner wheel and accelerating the outer wheel. The driving wheel angular speeds are:

$$\omega_{r_{-L}} = \omega_v + \frac{\Delta\omega}{2} \qquad (1.37)$$

$$\omega_{r_{-R}} = \omega_v - \frac{\Delta\omega}{2} \qquad (1.38)$$

2. Motor control

2.1.Dynamic d-q Model

2.1.1. Induction machine equivalent models

The machine's exact equivalent circuit, which is only applicable in steady-state conditions, has been taken into consideration. In addition, the dynamic d-q model of the machine serves as the foundation for high-performance drive control, such as vector- or field-oriented control.

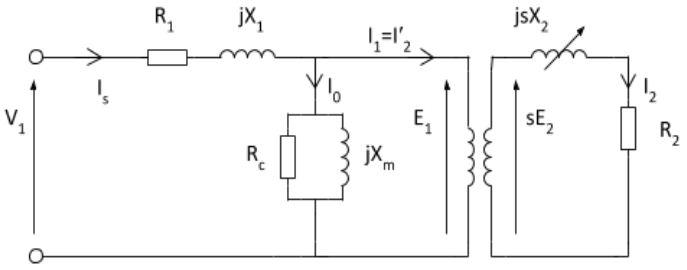


Figure 2.1 Steady state exact equivalent circuit

Since the three-phase rotor windings move relative to the three-phase stator windings, the dynamic performance of an ac machine is somewhat complex.

Lather, Kron suggested converting the stator and rotor variables to a reference frame that rotates simultaneously with the magnetic field. A transformation of stator variables to a revolving reference frame that is fixed on the rotor was proposed by D. S. Brereton.

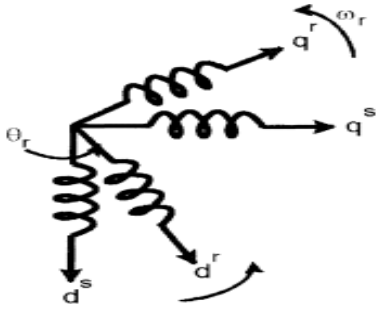


Figure 2.2 equivalent 2-phase machine

2.1.2. Axes Transformation

Consider a three-phase induction machine that is symmetrical and has immovable as-bs-cs axes spaced at a 2/3 angle. We intend to convert the variables from the three-

phase stationary reference frame (as-bs-cs) into the two-phase stationary reference frame (d^s- q^s), and then to convert them into the synchronously rotating reference frame (d^e- q^e). and vice versa.

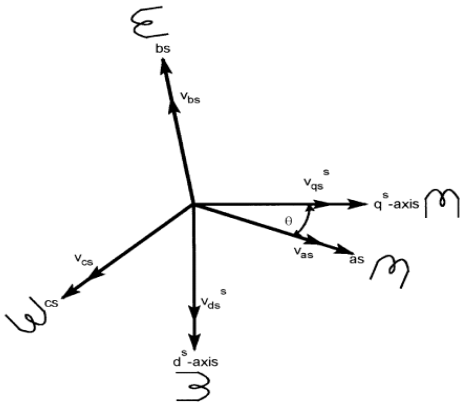


Figure 2.3 abc to d^s-q^s

The voltages V_{ds} and V_{qs} can be resolved into *as-bs-cs* components and can be represented in the matrix form as

$$\begin{bmatrix} Vas \\ Vbs \\ Vcs \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 1 \\ \cos(\theta - 120°) & \sin(\theta - 120°) & 1 \\ \cos(\theta + 120°) & \sin(\theta + 120°) & 1 \end{bmatrix} \begin{bmatrix} Vqs^s \\ Vds^s \\ Vos^s \end{bmatrix} \qquad (2.1)$$

The corresponding inverse relation is:

$$\begin{bmatrix} Vqs^s \\ Vds^s \\ Vos^s \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta & \cos(\theta - 120°) & \cos(\theta + 120°) \\ \sin\theta & \sin(\theta - 120°) & \sin(\theta + 120°) \\ 0.5 & 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} Vas \\ Vbs \\ Vcs \end{bmatrix} \qquad (2.2)$$

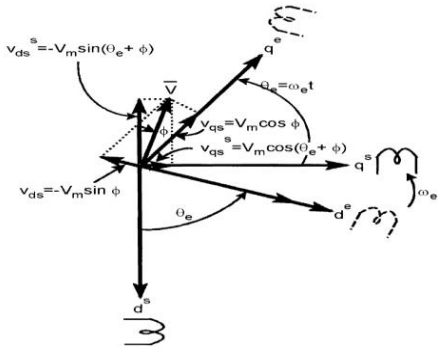


Figure 2.4 d^s-q^s to d^e-q^e

The synchronously spinning d^s-q^e axes are depicted in the previous figure. These axes rotate at synchronous speed with regard to the $d^s - q^s$ axes and the angle θe =ωet.

$$V_{qs} = V_{qs}{}^{s} \cos\theta_e - V_{ds}{}^{s} \sin\theta_e \qquad (2.3)$$

$$V_{ds} = V_{qs}{}^{s} \sin\theta_e - V_{ds}{}^{s} \cos\theta_e \qquad (2.4)$$

For convenience, the superscript *e* has been dropped from now on from the synchronously rotating frame parameters. Again, resolving the rotating frame parameters into a stationary frame, the relations are:

$$V_{qs}{}^{s} = V_{qs}\, cos\,\theta_e + V_{ds}\, sin\,\theta_e \qquad (2.5)$$

$$V_{qs}{}^{s} = -V_{qs}\, sin\,\theta_e + V_{ds}\, cos\,\theta_e \qquad (2.6)$$

Assume that the three-phase stator voltages are sinusoidal and balanced:

$$V_{qs}{}^{s} = V_m\, cos(\omega_e t + \varphi) \qquad (2.7)$$

$$V_{ds}{}^{s} = -V_m\, sin(\omega_e t + \varphi) \qquad (2.8)$$

According to equations (2.7) and (2.8), $V_{qs}{}^{s}$ and $V_{ds}{}^{s}$ are balanced, two-phase voltages with equal peak values.

A complex space vector (also known as a phasor) can be used to combine and express the variables in a reference frame.

$$\bar{V} = V_{qds}{}^{s} = V_{qs}{}^{s} - jV_{ds}{}^{s} = V_m[cos(\omega_e t + \varphi) + j\, sin(\omega_e t + \varphi)] = \hat{V}_m e^{j\omega} e^{j\omega et}$$

$$= \sqrt{2}\, Vs\, e^{j(\theta e + \omega)} \qquad (2.9)$$

which shows that from the initial (t = 0) angle to the q^e-axis, the vector $\bar{V}$ rotates anticlockwise at speed we. Additionally, equation (9) shows that for a sinusoidal variable.

The peak value ($\hat{V}m$), which is $\sqrt{2}$ times the rms phasor magnitude (*V*s), is the vector magnitude.

Note that the vector magnitudes in stationary and rotating frames are equal, that is:

$$|\bar{V}| = \hat{V}_m = \sqrt{V_{qs}{}^{s2} + V_{ds}{}^{s2}} \qquad (2.10)$$

2.1.3. Synchronously Rotating Reference Frame – Dynamic Model

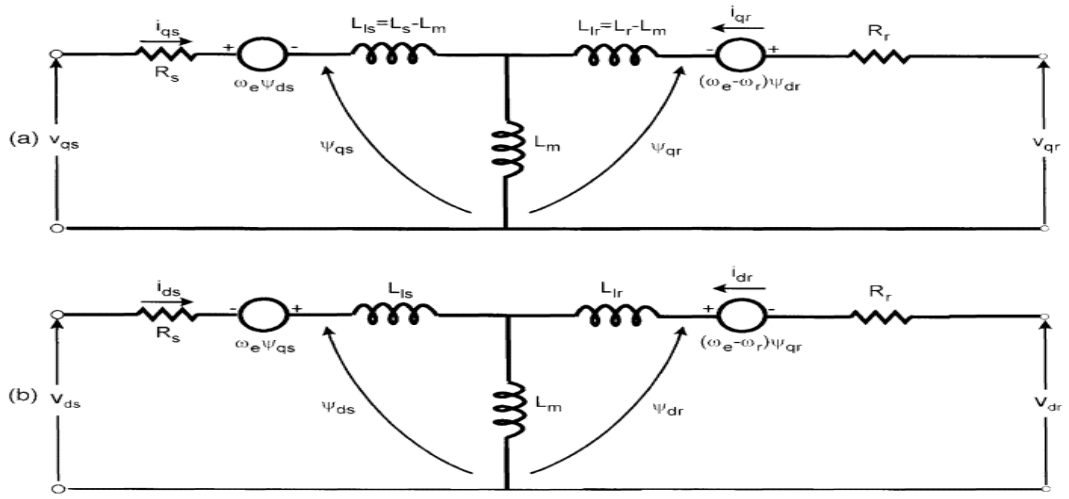


Figure 2.5 Dynamic equivalent circuit

We must represent the equivalent two-phase machine in a synchronously rotating d^e-q^e frame. The following equations for stator circuits can be written:

$$V_{qs} = R_s i_{qs} + \frac{d}{dt}\Psi_{qs} + \omega_e \Psi_{ds} \qquad (2.11)$$

$$V_{ds} = R_s i_{ds} + \frac{d}{dt}\Psi_{ds} - \omega_e \Psi_{qs} \qquad (2.12)$$

Where ${\Psi_{qs}}^s$ and ${\Psi_{ds}}^s$ are q-axis and d-axis stator flux linkages, respectively.

Hence, when $\omega_e = 0$, the equations return to their stationary form. Keep in mind that the d^e-q^e axis' flux coupling causes an emf there as well.

The stator is referred to as the source of all variables and parameters. The d-q axes fixed to the rotor move at a speed ωe - ωr in relation to the synchronously rotating frame since the rotor moves at speed ωr. Consequently, the rotor equations should be modified as follows in the d^e-q^e frame:

$$V_{qr} = R_r i_{qr} + \frac{d}{dt}\Psi_{qr} + (\omega_e + \omega_r)\Psi_{dr} \qquad (2.13)$$

$$V_{dr} = R_r i_{dr} + \frac{d}{dt}\Psi_{dr} + (\omega_e + \omega_r)\Psi_{qr} \qquad (2.14)$$

This figure presents the equivalent circuits for the d^e-$q^e a$ dynamic model that meets Equations (2.11)– (2.12) and (2.13)– (2.14). One unique benefit of the machine's d^e-q^e dynamic model is that as was previously described, all sinusoidal variations in the stationary frame translate into dc values in the synchronous frame.

2.2. Vector or Field-Oriented Control

2.2.1. Introduction

High reliability, good control characteristics, low maintenance requirements, low investment, and low running costs are among the important features that are required from a modern drive. For simple drives, the industry has relied mainly on squirrel-cage induction machines. A major disadvantage of this type of machine was its inability to be controlled efficiently.

Vector control denotes an independent (decoupled) control of the flux and the torque in AC machines. This is achieved by controlling the amplitude of the stator current space vector and its position with respect to the chosen flux space vector. The stator currents of a three-phase AC electric motor are identified as two orthogonal components that can be expressed with a vector in a process known as vector control, also known as field-oriented control (FOC). The motor's magnetic flux is determined by one component, and its torque by another.

Due to the superiority of FOC's motor size, cost, and power consumption reduction, it is becoming more and more appealing for applications requiring lower performance. It is anticipated that single-variable scalar volts-per-Hertz (V/f) control will eventually be practically universally replaced as microprocessor processing capability increases.

2.2.2. DC drive Analogy

Ideally, a vector-controlled induction motor drive operates like a separately excited dc motor drive. In a dc machine, neglecting the armature reaction effect and field saturation the developed torque is given by:

$$T_e = K_t I_a I_f \qquad (2.15)$$

where: I_a is the armature current, I_f is the field current and K_t is motor torque constant.

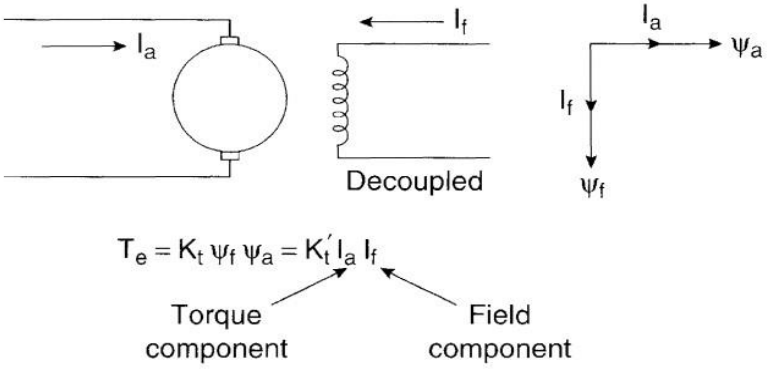


Figure 2.6 separately excited DC motor

In the dc-motor drive, the commutator and brushes ensure that the armature-current I_a produced magneto motive force (mmf) is at a right angle to the field flux Ψ_f produced by the stator. A DC machine is constructed in such a way that the field flux Ψ_f produced by the current I_f is perpendicular to the armature flux Ψ_a produced by the armature current I_a. The nature of these stationary space vectors is either decoupled or orthogonal. Decoupling means that when the field current I_f is changed, only the field flux Ψ_f is impacted. When torque is adjusted by adjusting the current I_a, the flux Ψ_f is unaffected, allowing us to use the rated Ψ_f .

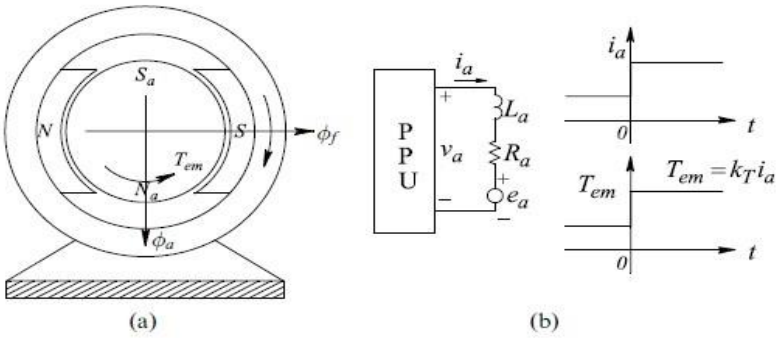


Figure 2.7 Input Step change in current and torque

To change T_e as a step, the armature current I_a is changed (at least, attempted to be changed) as a step, as shown in Fig. b.

DC machine-like performance can also be extended to an induction motor if the machine control is considered in a synchronously rotating reference frame (d_e-q_e), where the sinusoidal variables appear as dc quantities in steady state.

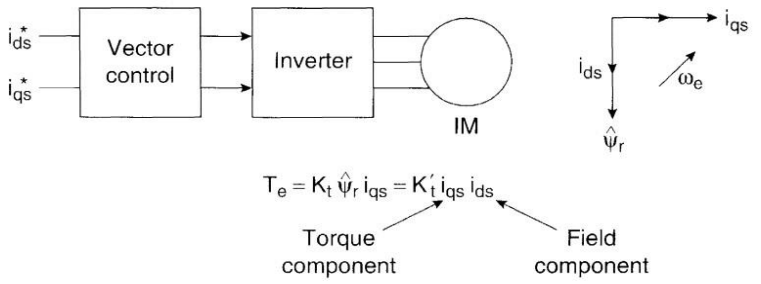


Figure 2.8 vector controlled induction motor

In a reference frame that rotates synchronously, the I_{ds} and I_{qs} represent the direct axis component and quadrature axis component of the stator current, respectively.

With vector control, I_{ds} and I_{qs} are comparable to the field current I_f and armature current I_a of a dc machine, respectively. Therefore, the torque can be expressed as:

$$T_e = K_t I_{qs} I_{ds} \qquad (2.16)$$

2.3. Fundamentals of Vector Control

2.3.1. Equivalent Circuit at Vector Control

The figure below shows the complex form of d_e-q_e equivalent circuits in steady-state conditions, where rms values V_s and I_s are replaced by corresponding peak values (sinusoidal vector variables), as shown. The rotor leakage inductance L_{lr}, has been neglected for simplicity, which makes the rotor flux Ψ_r the same as the air gap flux Ψ_m . The stator current can be expressed as

$$I_s = \sqrt{{I_{qs}}^2 + {I_{ds}}^2} \qquad (2.17)$$

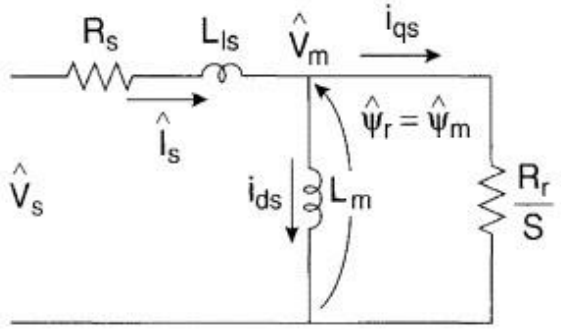


Figure 2.9 qds equivalent circuit in steady state

Where I_{ds} is magnetizing component of stator current flowing through the inductance L_{lr} and I_{qs} is torque component of stator current flowing in the rotor circuit.

Figure below shows the phasor (or vector) diagrams in d_e-q_e frame with peak values of sinusoids and air gap voltage aligned on the q-axis.

Figure (a) indicates an increase of the I_{qs} component of the stator current to increase the torque while maintaining the flux Ψ_r constant, whereas (b) indicates a weakening of the flux by reducing the I_{ds} component.

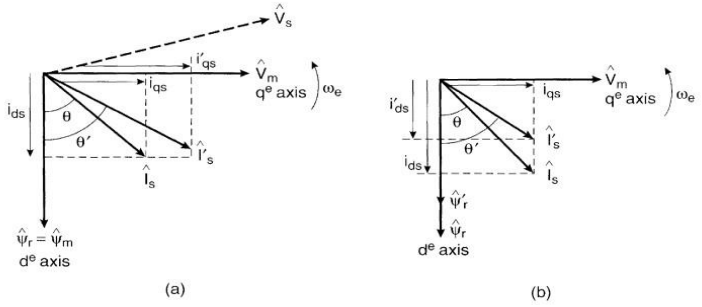


Figure 2.10 steady state phasors

2.3.2. Vector Control implementation model

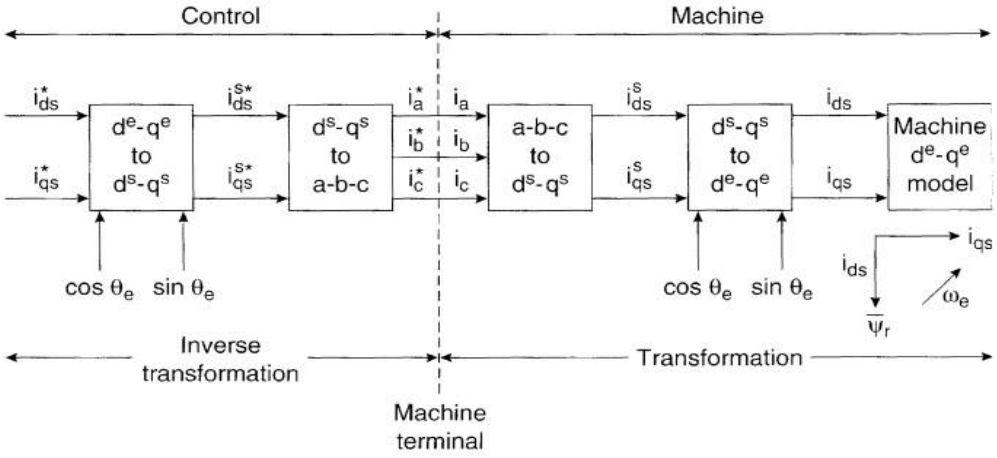


Figure 2.11 Vector control implementation principle

With the aid of the previous Figure, the basics of vector control implementation may be described. when the reference frame for the machine model rotates synchronously. The controller's matching command currents $i_a{}^*$, $i_b{}^*$, and $i_c{}^*$ direct the inverter to produce currents i_a, i_b, and i_c. The 3-phase / 2-phase transformation transforms the machine terminal phase currents i_a, i_b, and i_s into $I_{ds}{}^s$ and $I_{qs}{}^s$. components. These are then applied to the d_e - q_e machine model after being transformed into synchronously rotating frames by the unit vector components cos θ_e and sin θ_e.

The prior Chapter contains the transformation equations. In order for the control currents $I_{ds}{}^*$and $I_{ds}{}^*$ to match to the machine currents I_{ds} and I_{qs}, the controller performs an inverse transformation in two phases. Additionally, the unit vector ensures that the I_{ds} current and the flux vector Ψ_r are correctly aligned and I_{qs} perpendicular to it before incorporating them into the d_e -q_e model.

There are essentially two general methods of vector control. One, called the direct or feedback method, and the other, known as the indirect or feedforward method. The methods are different essentially by how the unit vector ($\cos\theta_e$ and $\sin\theta_e$) is generated for the control.

2.4. Indirect Vector Control

The indirect vector control approach is very similar to direct vector control, with the exception that the unit vector signals ($\cos\theta_e$ and $\sin\theta_e$) are produced in a feedforward manner. Industrial applications frequently use indirect vector control. With the use of a phasor diagram, the figure below shows the core idea of indirect vector control. The d_r - q_r axes, which are fixed to the rotor but not the stator, are moving as illustrated at speed ω_r. The d_s - q_s axes are fixed to the stator. By the positive slip angle θ_{sl} corresponding to the slip frequency ω_{sl}, the synchronously rotating axes d_e - q_e are

rotating in front of the d_r - q_r axes. Since $\omega_e=\omega_r + \omega_{sl}$ and the rotor pole is pointed at the d_e-axis, we may write

$$\theta_e = \int \omega_e \, dt \qquad (2.18)$$

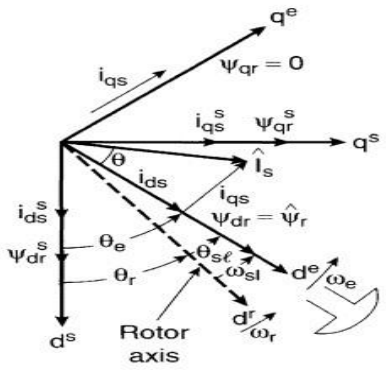


Figure 2.12 Phasor diagram of indirect vector control

2.4.1. Model of a motor with the d-axis aligned along the rotor flux linkage Ψ_r

The phasor diagram suggests that for decoupling control, the stator flux component of current I_{ds} should be aligned on the d_e axis, and the torque component of current I_{qs} should be on the q_e axis.

For decoupling control, we can now make a derivation of control equations of indirect vector control with the help of d_e - q_e equivalent circuits (discussed in the previous chapter). The rotor circuit equations can be written as

$$I_{qr}R_r + (\omega_e - \omega_r)\Psi_{dr} + \frac{d\,\Psi_{dr}}{dt} = 0 \qquad (2.19)$$

$$I_{dr}R_r + (\omega_e - \omega_r)\Psi_{qr} + \frac{d\,\Psi_{dr}}{dt} = 0 \qquad (2.20)$$

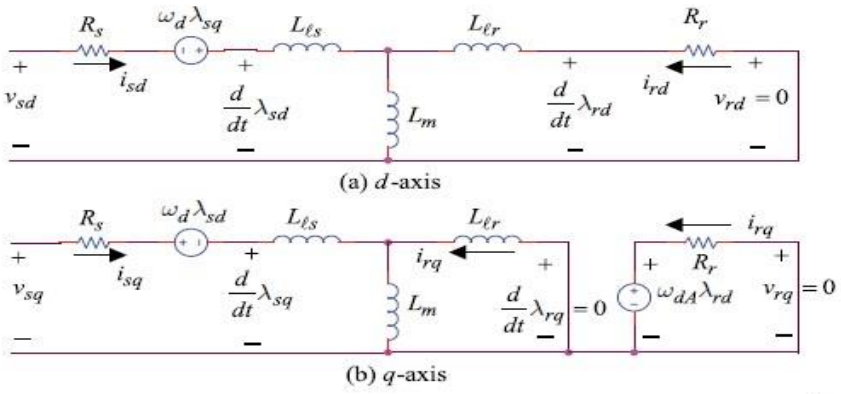


Figure 2.13 Dynamic circuits with the d-axis aligned along the rotor flux linkage Ψ_r

From the above circuits and for Decoupled control

$$\Psi_{qr} = 0 \qquad \frac{d\Psi_{dr}}{dt} = 0 \qquad \Psi = \Psi_{dr}$$

By substituting in (2.19) & (2.20)

$$I_{qr}R_r + (\omega_e - \omega_r)\Psi_{dr} = 0 \qquad (2.21)$$

$$I_{dr}R_r + \frac{d\Psi_{dr}}{dt} = 0 \qquad (2.22)$$

From d_e - q_e equivalent circuits the rotor flux linkage expressions can be given as:

$$\Psi_{qr} = I_{qs}Lm + I_{qr}L_r = 0 \qquad (2.23)$$

$$\Psi_{dr} = I_{ds}Lm + I_{dr}L_r = \Psi_r \qquad (2.24)$$

2.4.2. Slip Speed and Unit Vector calculations

From the above Equations:

$$I_{qr} = -I_{qs}\frac{L_m}{L_r} \qquad (2.25)$$

$$I_{dr} = \frac{\Psi_r}{L_r} - I_{ds}\frac{L_m}{L_r} \qquad (2.26)$$

By substituting from (2.25) into (2.21):

$$-I_{qs}\frac{L_m}{L_r}R_r + (\omega_e - \omega_r)\Psi_r = 0 \qquad (2.27)$$

$$\omega_{slip} = (\omega_e - \omega_r) = \frac{L_m}{\tau_r}\frac{I_{qs}}{\Psi_r} \qquad (2.28)$$

Where $\tau_r = \frac{L_r}{R_r}$, Rotor time constant

$$\omega_e = \omega_r - \omega_{slip}$$

Where ω_r (electrical speed of rotor) = $\frac{P}{2} * \omega_m$ (measured mechanical speed of rotor)

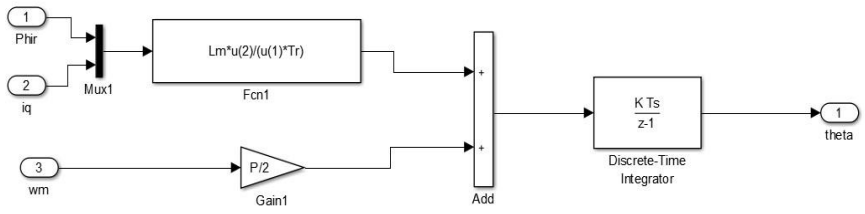


Figure 2.14 theta calculation

2.4.3. The rotor flux linkage Ψ_r calculations

By substituting from (2.26) into (2.27):

$$\Psi_r = I_{ds} L_m - \frac{d\Psi_r}{dt} \frac{L_r}{R_r}$$

In S-domain:

$$\Psi_r(s) = \frac{L_m}{1+s\tau_r} I_{ds}(s) \qquad (2.29)$$

Figure 2.15 rotor flux calculation

If rotor flux Ψ_r= constant which is usually the case

$$\Psi_r = L_m I_{ds} \qquad (2.30)$$

2.4.4. V_{qs} & V_{ds} calculations

The V_{qs} & V_{ds} are reference voltages which will be inverse transformed (discussed in the previous chapter) to be used to at SVPWM to generate gate pulses used in the Universal Bridge (power electronic inverter).

A dc machine-like electro-mechanical model of an ideal vector-controlled drive can be derived using the following equations:

$$T_e = \frac{3}{2} \frac{P}{2} \frac{L_r}{L_m} \Psi_r I_{qs} \qquad (2.31)$$

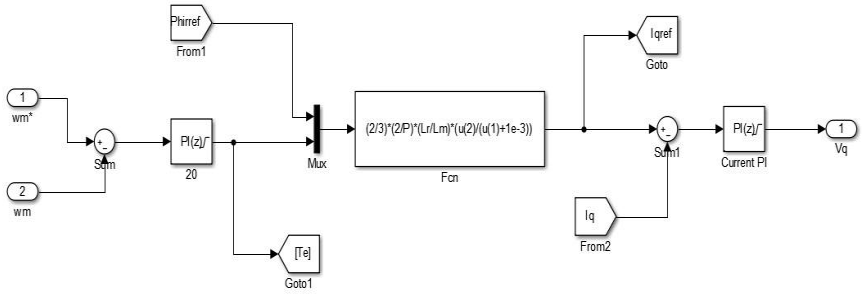


Figure 2.16 Vq calculation

V_{qs} calculations using the above equations and a PI controller, where the tuning of the PI controller gains will be explained in the next chapters.

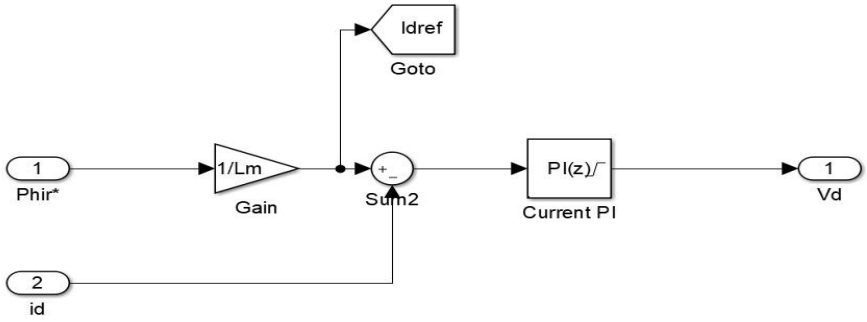


Figure 2.17 Vd calculation

V_{ds} calculations using the above equations and a PI controller, where the tuning of the PI controller gains will be explained in the next chapters.

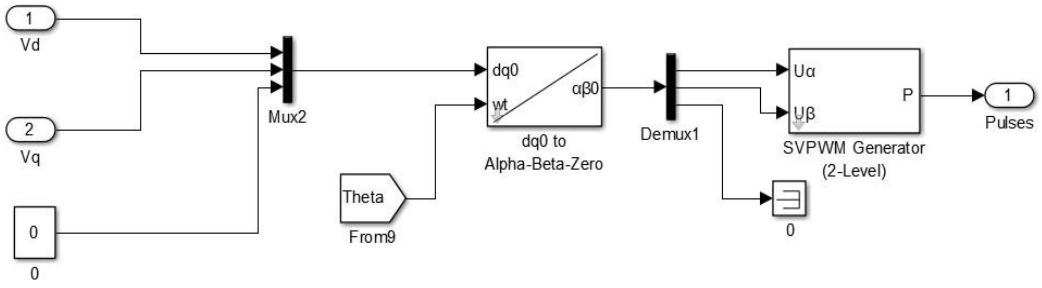
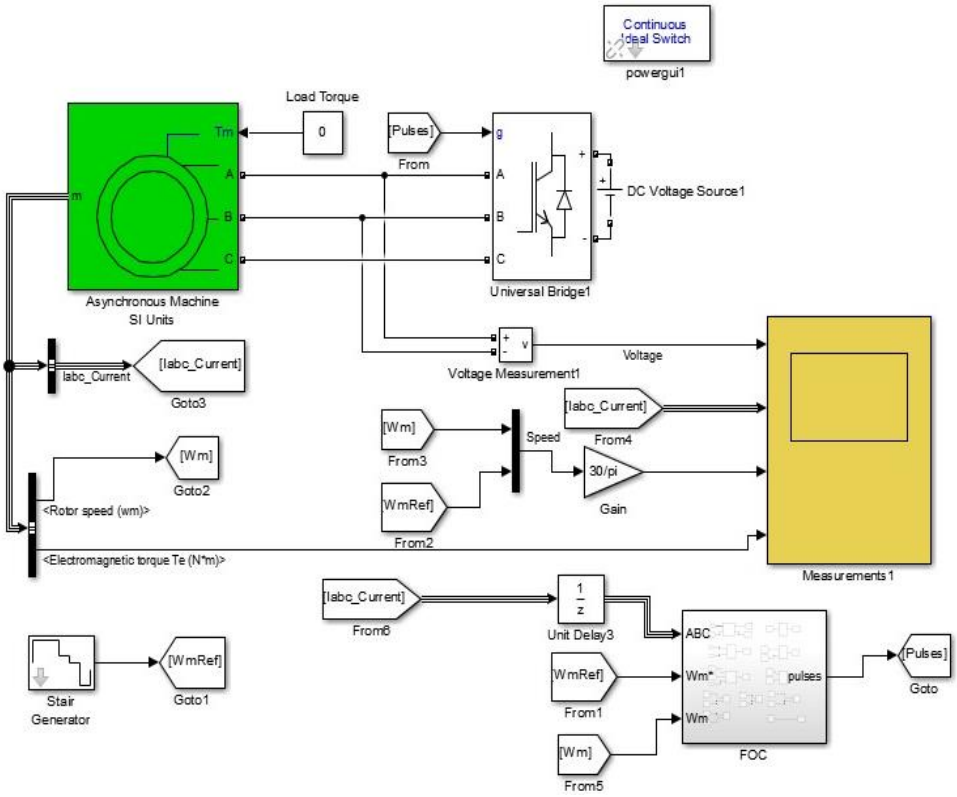


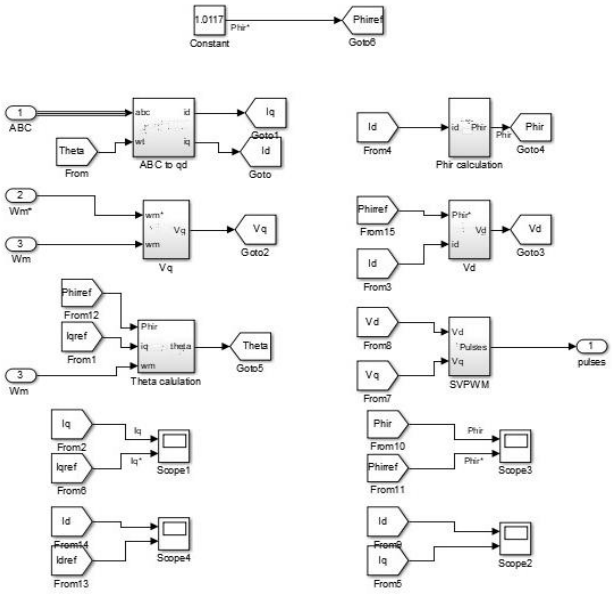
Figure 2.18

Obtaining a Pulse train after an inverse transformation (inverse Park transformation) of V_{ds} & V_{qs}.

2.4.5. Model configuration



Simulation Model of F.O.C



F.O.C block

2.5.PI Controller

A PI (proportional-integral) controller is a type of feedback control system commonly used in engineering applications to regulate the behavior of a system. This controller works by comparing the current output of a system to a desired setpoint and then adjusting the system's inputs accordingly to minimize the error between the two. The PI controller uses both proportional and integral terms to achieve the desired control response, where the proportional term provides a response to the current error and the integral term provides a response to the accumulated historical error.

2.5.1. PI for the current loop

The PI controller is commonly used in vector control and used to regulate the Id and Iq currents. The PI controller takes the error between the desired current and the actual current as input and produces a control signal to adjust the voltage applied to the motor. The proportional term in the PI controller responds to the current error, and the integral term responds to the accumulated error over time.

The use of PI controllers in the FOC system ensures that the motor current is accurately regulated, which is essential for achieving high-performance control of the motor. The PI controller adjusts the voltage applied to the motor to maintain the desired current level, which in turn controls the torque and speed of the motor.

$$G = \frac{1}{S\sigma L_s + R'_s}$$

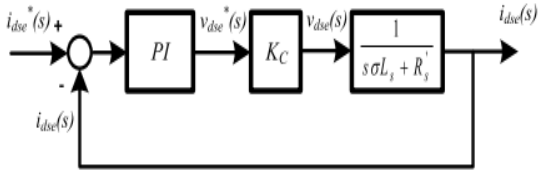


Figure 2.19 PI For Id

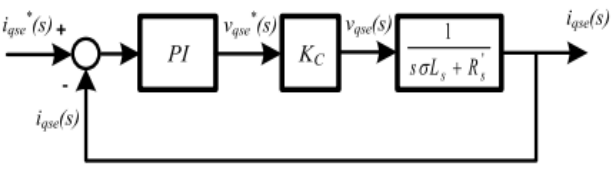


Figure 2.20 PI For Iq

$$\sigma = 1 - \frac{Lm^2}{L_s * L_r}$$

$$L_s = L_{Ls} + L_m$$

$$L_r = L_{Lr} + L_m$$

$$R'_s = R_s + R_r * \left(\frac{L_m}{L_r}\right)^2$$

The magnitude optimum technique is a method used for tuning the PI controller in a Field Oriented Control (FOC) system. The goal of this technique is to achieve the best performance in terms of response time and stability.

Optimization tuning methods for a PI controller involve using mathematical algorithms to determine the optimal values of the controller gains that will provide the best performance for the system. These methods are typically used in applications where the system requirements are complex, and a trial-and-error approach is not practical or efficient.

To apply this method, the system's transfer function is first analyzed in the frequency domain. The gain and phase margins are then computed to determine the system's stability.

This technique is useful because it ensures that the system is stable while also providing a fast response time. It also ensures that the system's performance is optimized by minimizing overshoot and settling time.

Overall, the magnitude optimum technique is a valuable tool for tuning the PI controller in an FOC system, providing optimal performance and stability.

It is possible to observe that the gains of the PI controller are related to the induction machine parameters as a DC machine from Optimization tuning.

$$K_p = \frac{w_c \sigma L_s}{K_c} \qquad (2.32)$$

$$\tau_i = \frac{K_p K_c}{R_s' * w_c} \qquad (2.33)$$

$$K_i = \frac{K_p}{\tau_i} \qquad (2.34)$$

If the power stage constant Kc is approximated to unity, for the same current bandwidth, the proportional gain parameter is proportional to the σL_s factor.

In the indirect vector sensor control, the currents components I_{ds} and I_{qs} should be sampled and updated at least at the speed that these control loops are executed to guarantee a minimum bandwidth of both controllers.

$$\omega_c = \frac{f_s}{21} * 2\,\pi \qquad (2.35)$$

The limit of PI, with respect to both I_q and I_d, is contingent upon the DC source and is equivalent to half of the voltage of the DC source (VDC/2).

2.5.2. PI for Speed loop

The PI controller in the speed loop of FOC works by comparing the actual speed of the motor to a desired speed setpoint. The difference between the two (known as

the speed error) is then fed into the PI controller, which calculates the required torque output to reduce the error. The output of the PI controller is then fed into the motor's torque control loop, which adjusts the motor's current magnitude to achieve the desired torque output. By adjusting the motor's current magnitude, the speed loop of FOC can regulate the motor's speed with high accuracy and precision.

Settling time calculation:

$$T_e - T_l = J\frac{\omega_f - w_i}{\Delta T} \qquad (2.36)$$

This equation enables the determination of an acceptable settling time for the motor, which can be utilized to tune the PI controller.

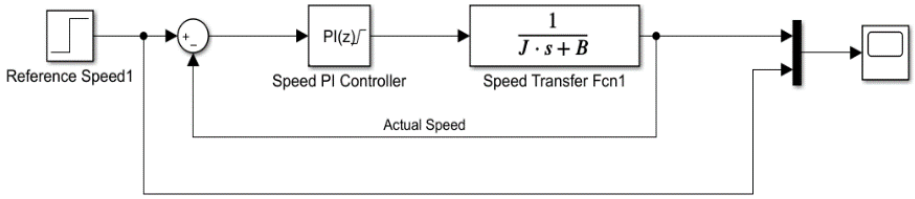


Figure 2.21: PI for Speed Loop

$$G = \frac{1}{JS + B}$$

If the effect of friction is neglected and only inertia is considered, then the resulting analysis or model would be simplified and focused solely on the system's kinetic properties and behavior.

$$G = \frac{1}{JS} = \frac{1}{T_2 S} \qquad (2.37)$$

According to the transfer function, the plant is a 1st order system. Therefore, a PI controller is sufficient to follow a constant reference with no permanent error. The proportional and integral terms are tuned using the Symmetrical optimum criterion.

The motor speed controller generates only a torque reference and, therefore, must be cascaded with a torque controller to produce an effect on the motor.

$$T_{sens} = 0$$

$$T_{ctrl} = N * T_{sw}$$

$$T_{PWM} = \frac{T_{sw}}{2}$$

$$T_{sw} = \frac{1}{F_{sw}}$$

The total delay of the outer loop is then the sum of the small-time constants:

$$T_{tot} = T_s + T_{ctrl} + T_{PWM}$$

$$T_N = 4T_{tot}$$

$$T_i = 8 * \frac{(T_{tot})^2}{T_2}$$

$$K_P = \frac{T_N}{T_i}$$

$$K_i = \frac{1}{T_i}$$

The limits of the speed loop PI controller are determined by the maximum torque, which can be twice the rated torque.

	Symbol	Definition
Sensing delay	T_{sens}	Delay in the measured quantity, due to finite sensor and analog chain bandwidth, and possibly filtering delay
Control delay	T_{ctrl}	Delay between sampling instant and duty-cycle update instant in the PWM modulator
Modulator delay	T_{PWM}	Average delay between duty-cycle update in the PWM modulator and resulting change in modulator output
Total loop delay	T_{tot}	Sum of the above delays, representing the total delay of the control system
Switching Frequency	F_{sw}	The switching frequency of the inverter
	N	the ratio of the execution rates of the speed control and current control loops.

Table 2.1

2.6. Motor parameters

Squirrel-cage induction machine 100 HP (75 KW) 400V 50Hz 1484Rpm.

Pn (VA)	Vn (Rms)	Frequency	R_s	L_{ls}	R_r	L_{lr}	L_m	J	Pole pairs
7.5e4 VA	400 V	50 Hz	0.03552 Ω	0.000335 H	0.02092 Ω	0.000335 H	0.0151 H	1.25 kg.m^2	4

Table 2.2

The equations presented in the preceding discussion can be used to calculate the values of the proportional gain (K_P) and integral gain (K_i) for both the current loop and the speed loop in a motor control system. By solving the equations for the desired closed-loop performance specifications, the appropriate values of K_P and K_i can be determined for each loop.

The speed controller is executed 20 times slower than the current control (N=20).

This way, the torque controller has enough time at its disposal to regulate the stator currents before the reference from the speed control is updated.

PI for current loop	Value	PI for Speed loop	Value
K_P	0.992768321995467	K_P	152.4390244
K_i	83.2165403436252	K_i	9295.062463

Table 2.3

2.7.Simulation Results

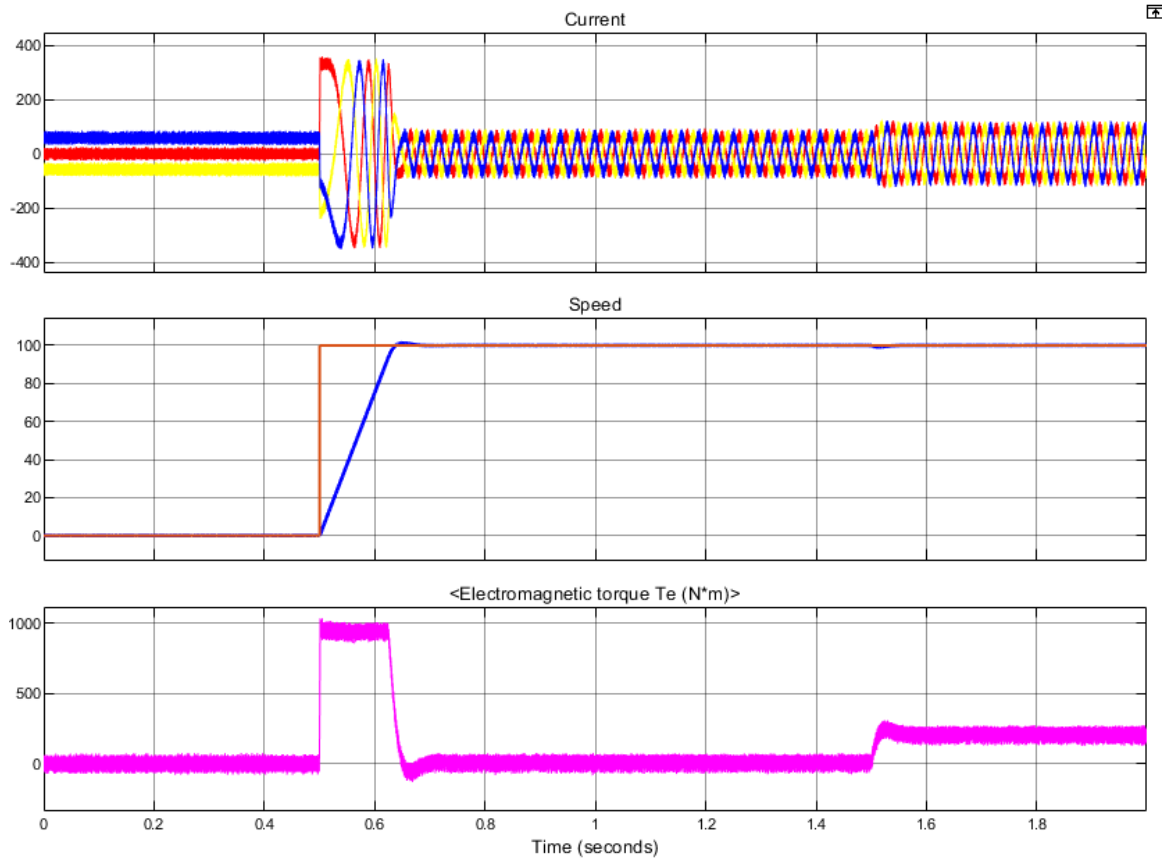


Figure FOC Simulation Result

2.8.DTC (Direct torque control)

Direct Torque Control (DTC) is a popular control technique used in AC machines, particularly in induction motors. It offers fast dynamic torque response and low dependence on machine parameters, making it a suitable choice for various applications. However, DTC has some limitations, such as the complexity of the control algorithm and the torque and flux oscillations caused by the variable duration of switching.

To overcome these limitations, a variant of DTC called DTC Constant Frequency Modulation (DTC-SVM) has been introduced. DTC-SVM imposes a constant modulation frequency, which reduces torque and flux oscillations while simplifying the control algorithm and reducing the computational burden. This technique has proven to be effective in enhancing the performance of DTC and making it a more practical option for real-time control applications.

2.8.1. Control Strategy of DTC

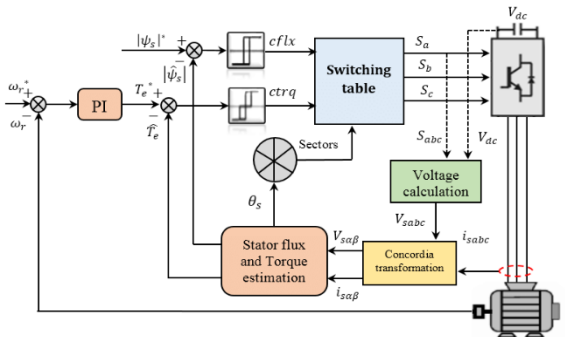


Figure2.22 Direct torque control block diagram

- Stator flux (Ψ_s) and torque (T_e) are calculated from the estimator.
- Reference torque ($T_e{}^*$) is calculated from reference speed and actual speed of the motor.
- The reference stator flux magnitude ($\Psi_s{}^*$) and the reference torque ($T_e{}^*$) are compared with the estimated flux magnitude (Ψ_s) and the estimated torque (T_e). The flux error and torque error are delivered to the hysteresis controller.

The direct torque control (DTC) is based on the direction of the stator flux and employs the voltage vector instantaneous values. A three-phase inverter can provide eight instantaneous base voltage vectors, two of which are null vectors. These vectors are selected from the commutation table based on flux and torque errors as well as the location of the stator flux vector. DTC is therefore distinguished as a technology suitable for operating alternating current devices without using mechanical sensors.

The model yields the following values at each moment based on measurements of the DC voltage at the inverter's input and the currents of the stator phases:

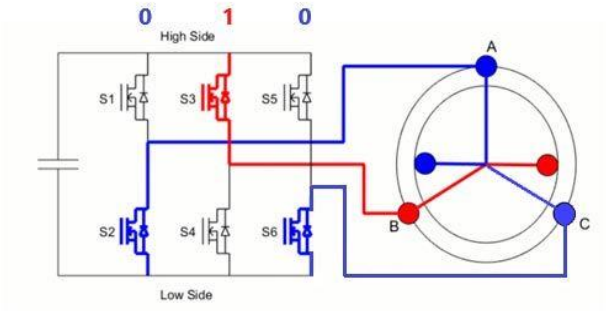


Figure2.23 Inverter connected to the stator of the motor.

1. The motor's real stator flux.
2. The actual torque it generates.
3. The position of the sector.

2.8.2. Voltage sectors

A typical two-level voltage inverter accomplishes seven unique phase plane locations, which correspond to the inverter's eight voltage sequences. The various combinations of the three values (S1, S2, and S3) allow for the generation of eight places of the vector VS, two of which correspond to the null vector.

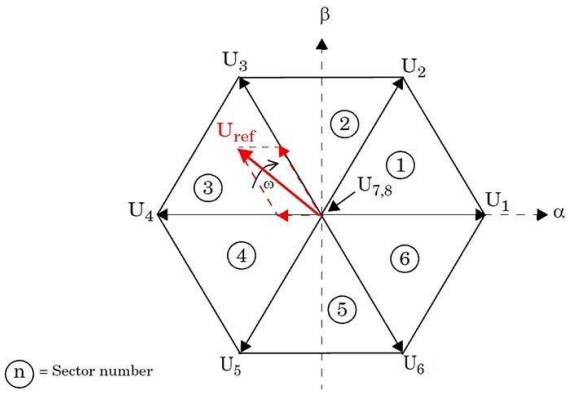


Figure 2.24 Space vector hexagon for two-level inverter.

Space Vector	S1	S3	S5
U1	1	0	0
U2	1	1	0
U3	0	1	0
U4	0	1	1
U5	0	0	1
U6	1	0	1
U7	0	0	0
U8	1	1	1

Figure 2.25 Table of switches.

2.8.3. ESTIMATORS

2.8.3.1. Estimation of stator flux

The flux can be estimated from the stator current and the stator voltage of the motor from this equation:

$$\Psi_s = \int_0^t (V_s - R_s I_s)\, dt \qquad (2.38)$$

And we can get alpha and beta components of the stator flux from these equations:

$$\Psi_{s\alpha}=\int_0^t (V_{s\alpha}-R_sI_{s\alpha})\,dt \qquad (2.39)$$

$$\Psi_{s\beta}=\int_0^t (V_{s\beta}-R_sI_{s\beta})\,dt \qquad (2.40)$$

The voltages $(V_{s\alpha})$and $(V_{s\beta})$ are obtained from Concordia transformation.

Similarly, The voltages $(I_{s\alpha})$and $(I_{s\beta})$.

$$\Psi_s=\sqrt{\Psi_{s\alpha}^2+\Psi_{s\beta}^2} \qquad (2.41)$$

And we can find the angle α_s from the equation:

$$\alpha_s=\tan^{-1}\left(\frac{\Psi_{s\beta}}{\Psi_{s\alpha}}\right) \qquad (2.42)$$

2.8.3.2. Estimation of torque

The torque (T_e) can be estimated from the stator flux and stator current. Their Alpha beta components from this equation:

$$T_e=p(\Psi_{s\alpha}I_{s\beta}-\Psi_{s\beta}I_{s\alpha}) \qquad (2.43)$$

2.8.4. Hysteresis controllers

2.8.4.1. Flux Hysteresis controller

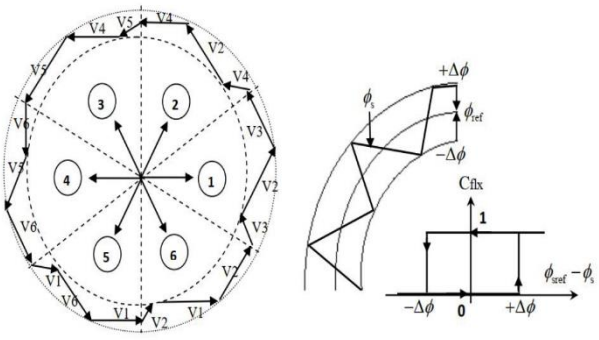


Figure2.26 hysteresis and selection of the corresponding voltage vectors.

With this type of controller, we can easily control and trap the extremity of the flux vector in a circular crown. The output of the hysteresis controller is represented by a Boolean variable (C_{flx}), $(C_{flx}=0)$ indicates that the amplitude of the flux must be

decreased, and ($C_{flx} = 1$) indicates that the amplitude of the flux must be increased to maintain the following equation:

$$\left|\Psi_s^* - \Psi_s\right| \leq \Delta\Psi_s \qquad (2.44)$$

Ψ_s^*: is the reference flux.

Ψ_s: is the estimated flux.

$\Delta\Psi_s$: is the flux hysteresis band width.

- if $\Delta\Psi_s > \varepsilon_\psi$ so $C_{flx} = 1$
- if $0 \leq \Delta\Psi_s \leq \varepsilon_\psi$ and $\frac{\Delta\Psi_s}{dt} > 0$ so $C_{flx} = 0$
- if $0 \leq \Delta\Psi_s \leq \varepsilon_\psi$ and $\frac{\Delta\Psi_s}{dt} < 0$ so $C_{flx} = 1$
- if $0 \leq \Delta\Psi_s < -\varepsilon_\psi$ so $C_{flx} = 0$

2.8.4.2. Torque Hysteresis controllers

The difference between flux controller and torque controller is that the torque can be positive or negative depending on the direction of rotation of the motor.

$$\left|T_e^* - T_e\right| \leq \Delta T_e \qquad (2.45)$$

T_e^*: is the reference torque.

T_e: is the estimated torque.

ΔT_e: is the toque hysteresis band width.

2.8.5. Types of hysteresis controllers

2.8.5.1. Two-level hysteresis controller

This controller can be used in flux vector control. The two-level controller can have used also if the torque control is only one direction of rotation. Only the voltage vectors V_{i+1} and V_{i+2} and zero vectors can be selected to evolve the flux vector (where [i] is the number of sector). The selection of null vectors is used to make a reduction in flux or in torque.

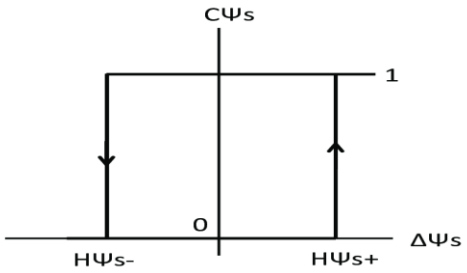


Figure 2.27 two-level hysteresis controller (flux controller)

2.8.5.2. Three-level hysteresis controller

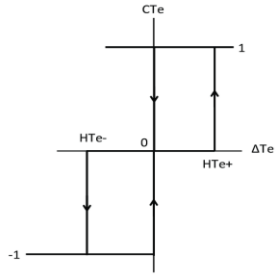


Figure 2.28 three-level hysteresis controller (torque controller)

It can be used to control the torque either for positive or negative torque. For the motors that need to rotate in both directions.

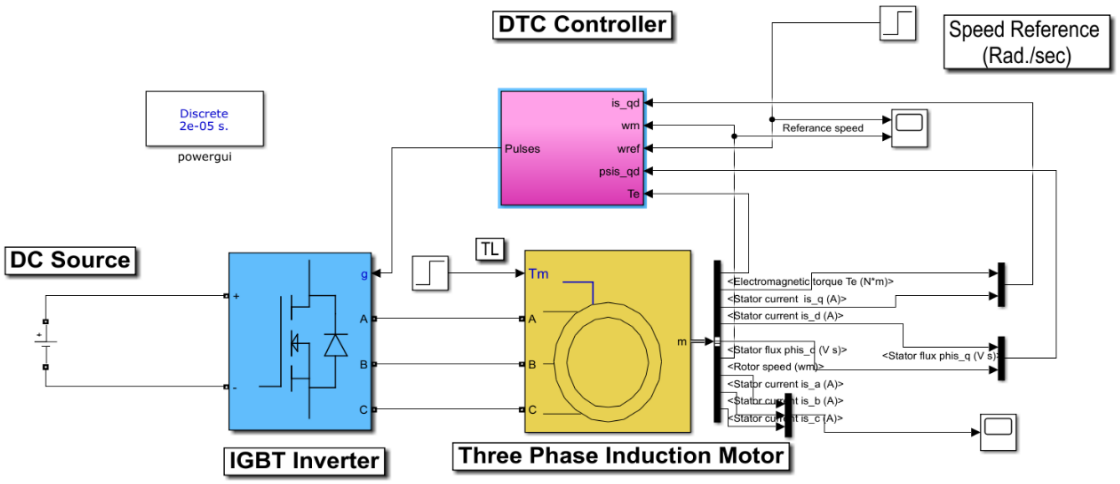
The controller output, represented by the Boolean variable C_{trq} indicates directly whether the torque amplitude must be increased in absolute value ($\{C_{trq} = 1\}$for a positive setpoint and $\{C_{trq} = -1\}$ for a negative setpoint) or reduced ($C_{trq} = 0$).

2.8.6. Switching Table of Inverter Voltage Vectors

C_{flx}	C_{trq}	S1	S2	S3	S4	S5	S6
1	1	V2	V3	V4	V5	V6	V1
	0	V0	V7	V0	V7	V0	V7
	-1	V6	V1	V2	V3	V4	V5
0	1	V3	V4	V5	V6	V1	V2
	0	V7	V0	V7	V0	V7	V0
	-1	V5	V6	V1	V2	V3	V4

Table 2.4 Switching Table

2.8.7. Model Configuration



Simulation Model of DTC

2.8.8. Simulation Results

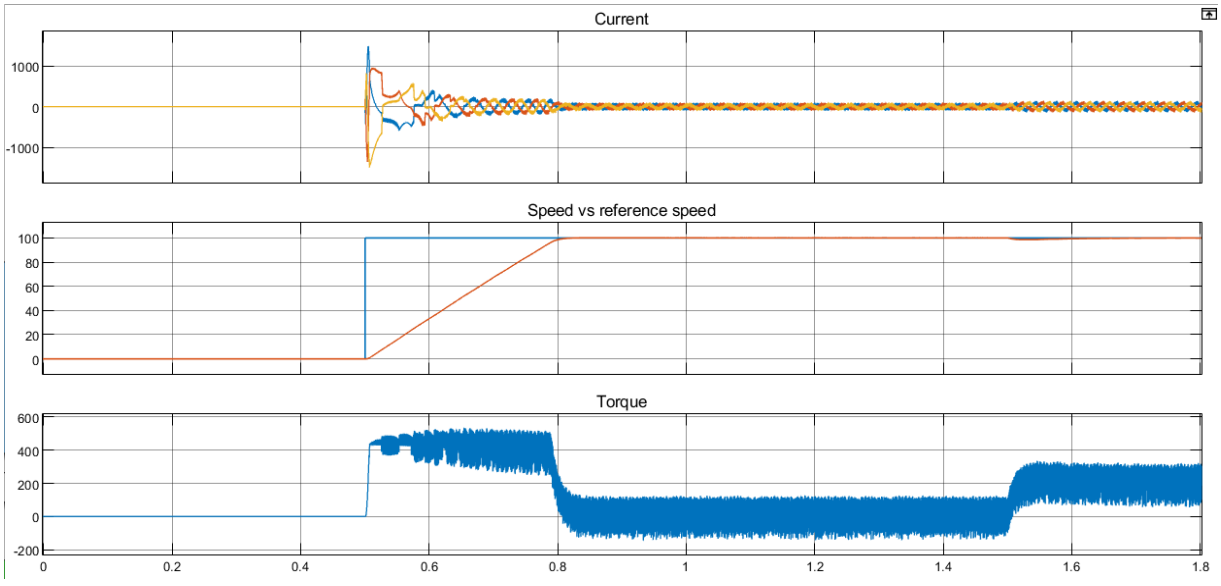


Figure DTC Simulation Result

2.8.9. DTC Disadvantages

- The existence of low-speed problems (influence of the resistive term).
- In this type of control, we need to have estimates of stator flux and motor torque.
- The existence of torque oscillations.
- The switching frequency is not constant (because of using hysteresis controller), which leads to some content rich in harmonics which increases the losses and THD and leads to acoustic noise and torque oscillations which can excite mechanical resonances.

However, the DTC is a command which is based on the estimation of the stator flux and the electromagnetic torque. Only the variation of the resistance of the stator, due to changes in temperature or operation at low rotational speeds, degrades the performance of the DTC control.

2.9.MRAS control

The speed can be calculated by the model referencing adaptive system (MRAS), where the output of a reference model is compared with the output of an adaptive model until the errors between the two models vanish to zero. The MRAS observer employs two independent expressions for the time derivative of rotor fluxes, obtained from equations of the IM model in the stationary reference frame α-β.

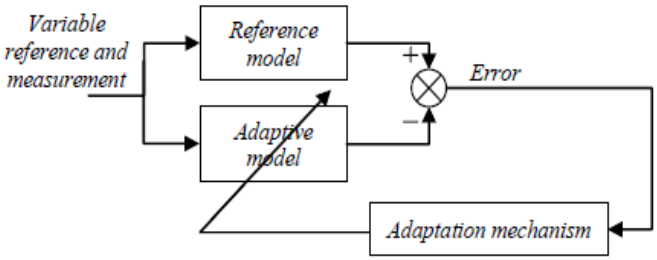


Figure 2.29 MRAS model

In our case, it is based on two models of flux vector estimation which are the Voltage model and the Current model. The observer makes use of two motor models of different structures to estimate the state variables based on different inputs and adaptation mechanisms.

2.9.1. Voltage model

This reference model takes machine stator voltages and currents as inputs to calculate the rotor flux signals.

$$\frac{d}{dt}(\Psi_{dr}{}^{s})=\frac{L_r}{L_m}V_{ds}{}^{s}-\frac{L_r}{L_m}(R_s+\sigma L_s S)I_{ds}{}^{s} \qquad (2.46)$$

$$\frac{d}{dt}\left(\Psi_{qr}{}^{s}\right)=\frac{L_r}{L_m}V_{qs}{}^{s}-\frac{L_r}{L_m}(R_s+\sigma L_s S)I_{qs}{}^{s} \qquad (2.47)$$

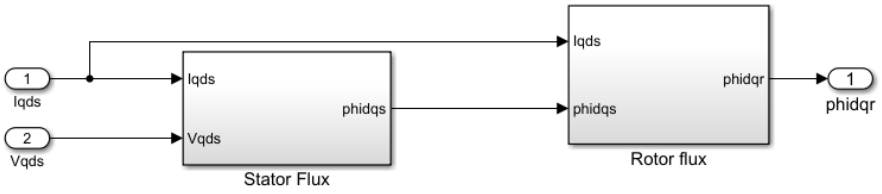


Figure 2.30 Voltage model

Figure 2.31 STATOR FLUX

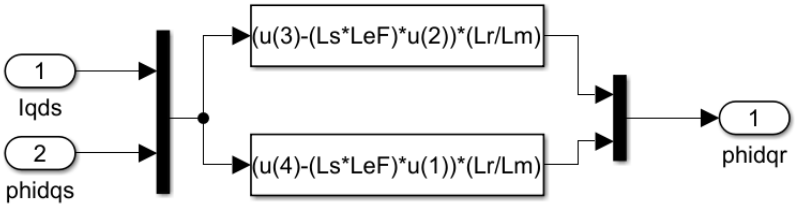


Figure 2.32 ROTOR FLUX

2.9.2. Current model

This adaptive model calculates rotor flux from the stator currents only when the speed signal is known.

$$\frac{d}{dt}(\Psi_{dr}{}^{s})=\frac{L_m}{T_r}I_{ds}{}^{s}-\omega_r\Psi_{dr}{}^{s}-\frac{1}{T_r}\Psi_{dr}{}^{s} \qquad (2.48)$$

$$\frac{d}{dt}\left(\Psi_{qr}{}^{s}\right)=\frac{L_m}{T_r}I_{qs}{}^{s}-\omega_r\Psi_{qr}{}^{s}-\frac{1}{T_r}\Psi_{qr}{}^{s} \qquad (2.49)$$

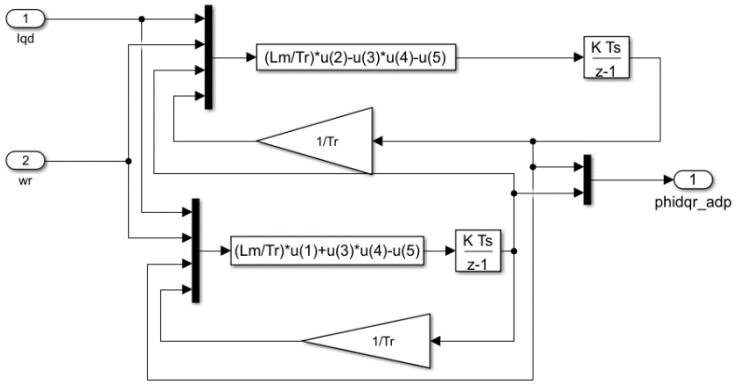


Figure 2.33 Current model

2.9.3. Adaptation mechanism

An adaptation algorithm with a P-I controller can be used to tune the speed $\widehat{\omega}r$ so that the error ξ=0. In designing the adaptation algorithm for the MRAS, it is important to take account of the overall stability of the system and to ensure that the estimated speed will converge to the desired value with satisfactory dynamic characteristics. Using Popov's criteria for hyperstability for a globally asymptotically stable system.

$$\widehat{\omega}_r=\xi(K_p+\frac{K_I}{S}) \qquad (2.50)$$

$$\xi=X-Y=\widehat{\Psi}_{dr}{}^{s}\Psi_{qr}{}^{s}-\Psi_{dr}{}^{s}\widehat{\Psi}_{qr}{}^{s} \qquad (2.51)$$

With the correct speed signal, the Fluxes from both models will match.

2.9.4. Speed estimation by model referencing adaptive control (MRAC) principle:

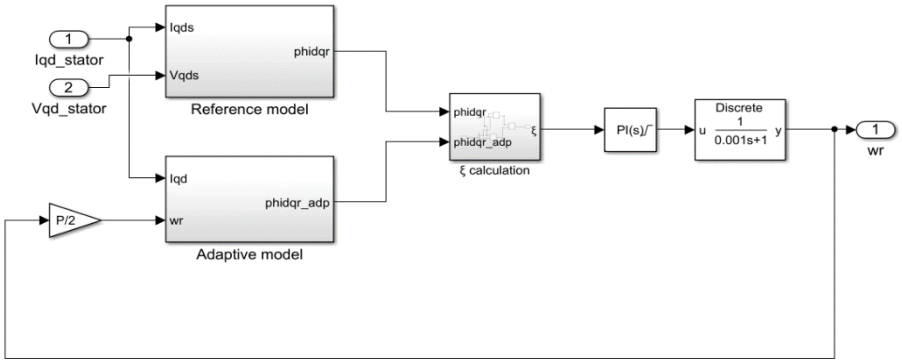


Figure 2.34 Speed estimation

2.9.5. Model Configuration

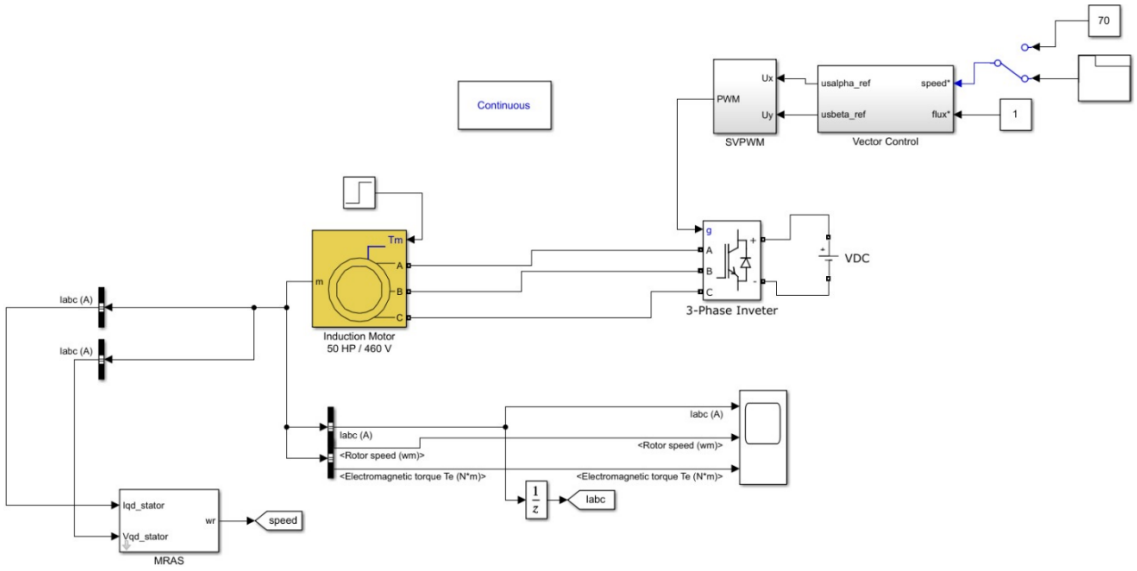


Figure Simulation Model of MRAS

2.9.6. Simulation Results

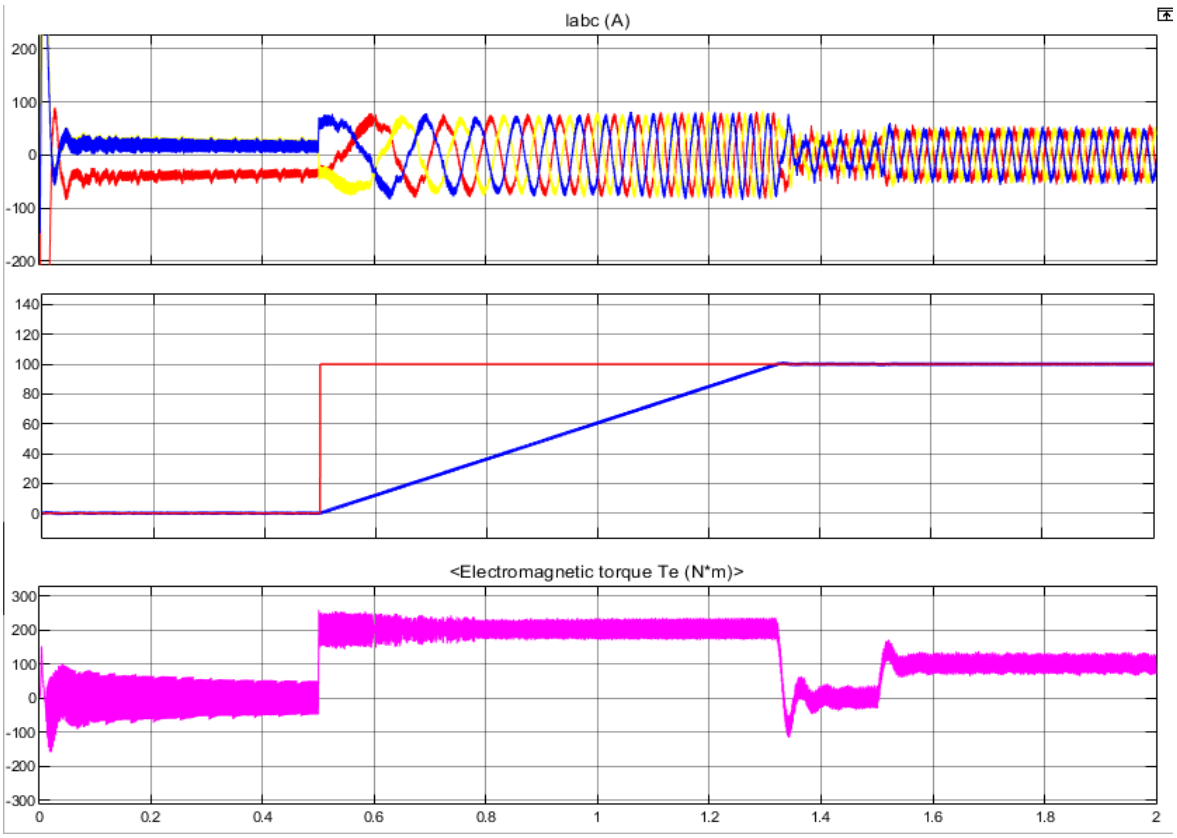


Figure MRAS Simulation Result

The motor utilized in the MRAS model differs from that employed in the DTC and FOC models, with a maximum torque capacity of 200 Nm (MRAS Model).

2.10. Comparison between DTC and FOC

Comparison of Field Oriented Control and Direct Torque Control		
	Field-Oriented Control	Direct Torque Control
Frame of Reference	Synchronously rotating (d-q).	Stationary (α, β).
Controlled Variables	Torque (T_e), Rotor Flux (φ_r).	Torque (T_e), Stator Flux (φ_s).
Control Parameters	Stator currents	Stator voltage space vector.
Sensed Variables	Rotor mechanical speed, Stator currents.	Stator voltages, Stator currents.
Estimated Variables	Rotor flux position (φ_k), Slip frequency (ω_{slip}).	Torque, Stator flux.
Controllers	Linear controllers for Stator currents.	Hysteresis controllers for Torque and Stator flux.
Torque Control	Indirectly controlled by stator currents, High dynamics, Torque ripple.	Directly controlled, High dynamics, Controlled Torque ripple.
Flux Control	Indirectly controlled by stator currents, Slow dynamics.	Directly controlled, Fast dynamics.
Parameter Sensitivity	Sensitive to rotor time constant (T_r).	Sensitive to stator resistance (R_s).
Implementation Complexity	High complexity, Coordinate transformation necessary.	Medium complexity, No coordinate Transformation.

Table 2.5 Comparison between DTC and

3. Proposed Systems

3.1. Experimental FOC (DSP model)

3.1.1. Motor Parameters

S	Vs	frequency	Rs	Rr	L_{ls}
2 HP	380 V	50 Hz	5.883 Ω	1.8993 Ω	13.577e-3 H
L_{lr}	L_m	Number of poles	inertia	Friction	
13.577e-3 H	277e-3 H	4	0.01 kg.m^2	0.001	

Table 3.1 Motor Parameters

3.1.2. Speed measurement (Encoder)

An incremental encoder interface is a type of sensor that can detect mechanical displacement without directly measuring the angular speed of a rotating shaft or object. This means that the encoder produces output pulses that need to be counted or timed to calculate the rotational speed.

The resulting values obtained from counting or timing the encoder output pulses are either a frequency or a period, which can be used to derive the rotational speed. Specifically, the speed is directly proportional to the frequency and inversely proportional to the period.

In discrete time the signals are quantized. The encoder interface samples A and B output signals regularly enough to detect every A&B state change before the following variation occurs. Upon detecting a state change, the position counts $x_p(k)$ is incremented or decremented based on whether A leads (Forward rotation) or lags B (Reverse rotation). This can be done by storing a copy of the previous A&B state and, when the state changes, comparing the current states of A&B with previous A&B states to determine the direction of rotation.

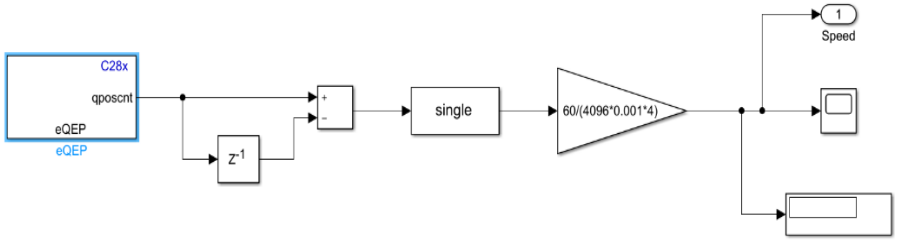
Consequently, signals QEPxA and QEPxB are both processed for computing angular speeds. However, regarding the implementation, estimating the speed from a digital position sensor is an economical strategy in motor control.

Hence, two different first-order approximations for the speed computation may be written starting from the position counter $x_p(k)$:

$$\omega(k)=\frac{x_p(k)-x_p(k-1)}{T_\omega}=\frac{\Delta x(k)}{T_\omega} \qquad (3.1)$$

where ω(k) is the speed computed at time step k, $x_p(k)$ and $x_p(k-1)$ are the position counters at time steps (k) and (k – 1) respectively, T_ω is the acquisition time window, which is the inverse of the speed calculation rate.

The angular speed must be indirectly computed by counting the encoder output pulses on QEPxA and QEPxB. When the encoder interface detects a state variation, it increments the counter $x_p(k)$. This information is available at the eQEPx block output and used to compute a frequency or period from which speed can be calculated.



Figur3.1 Speed reading from the rotary encoder

3.1.3. eQEP Block Parameters

3.1.3.1. General (Tab)

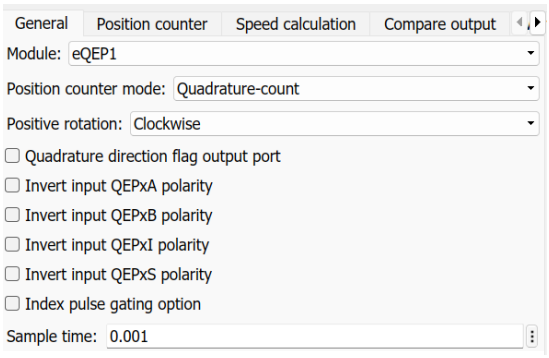


Figure 3.2 General Tab Parameters

- Select **module** eQEP2.
- Set **Position counter mode** to Quadrature-count.
- Set **Positive rotation** equal to Clockwise.
- Set **Sample time** equal to T_ω.
- Leave the other default settings.

3.1.3.2. Position counter (Tab)

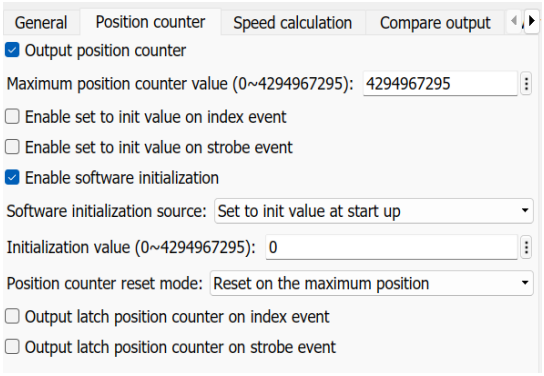


Figure 3.3 Position counter tab Parameters

- Flag Output position counter.
- Set Maximum position counter to $2^{32} - 1$.
- Flag Enable software initialization.
- Set Software initialization source equal to Set to initial value at start up.
- Set Initialization value equal to 0.
- Set Position counter reset mode equal to Reset on the maximum position.
- Leave the other default settings.

3.1.4. Scaling

Every time the position signals QEPxA e QEPxB are sampled, the pulses are detected and counted by the encoder interface. The MCU has read access to the interface and the eQEP module directly returns the position Counter through qposcnt. The latter is then used to compute the speed by taking the actual position count $x_p(k)$and the previous one $x_p(k-1)$, within a specific period T_ω. This can be obtained by using an add and delay block with sample time T_ω. The result is then multiplied by $\frac{60}{cpr*T_\omega}$.

Where cpr of our encoder is 4096, And we choose $T_\omega = 0.001$.

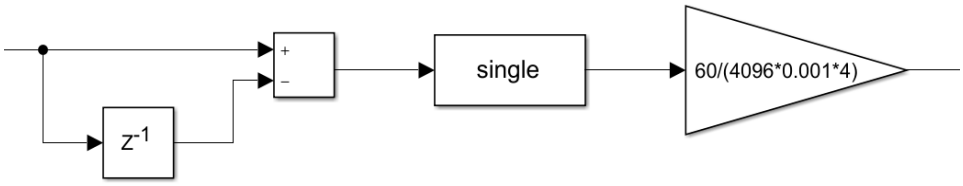


Figure 3.4 Scaling block to calculate the speed in rpm.

3.2. Current measurements

The used current sensor is LTS 25-NP. We used 2 current sensors to measure the stator current of two phases and use these readings to estimate the stator current of the third phase by using a balanced system low: $I_c = -I_b - I_a$

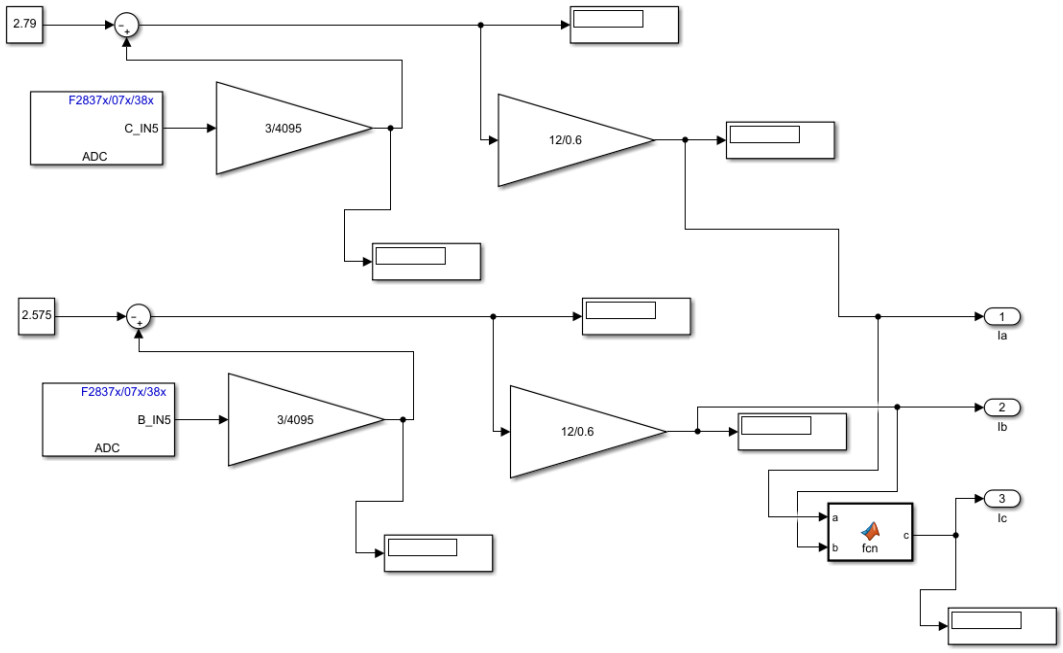


Figure 3.5 Current Measurement

After the Analog to digital converter, we need to scale the reading to a value from 0 to 3 volts because our DSP can withstand a maximum volt of 3.3 volts.

$$V_{out} = 2.5 \pm 0.6 \frac{I_p}{\pm 12}$$

To find the current we should subtract the internal offset of the current sensor The result is then multiplied by $\frac{12}{0.6}$.

$$I_p = (V_{out} - 2.5) \times \frac{12}{0.6}$$

3.3. SPWM Generation

Using ePWM block which has an internal carrier with an amplitude of "Timer period", so we need to make the switching $f_s = 5\ KHz$, And the CPU clock speed is 100 MHz.

$$\therefore Timer\ period = \frac{100 \times 10^6}{2 \times 5000} = 10000$$

So, we need to shift (Biased) the duty cycle by 1, Divide the result by 2, And the final result can be obtained by multiplying the result of division process by timer period value (10000).

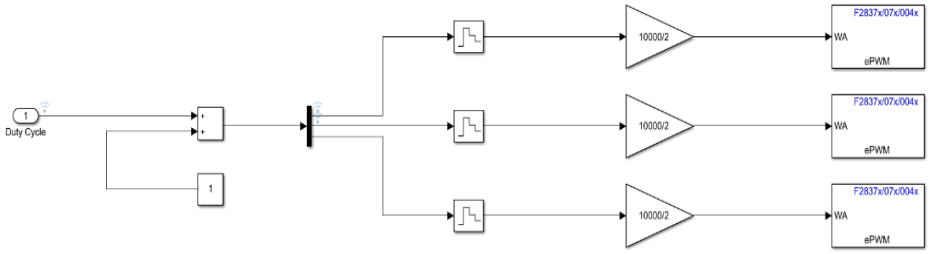
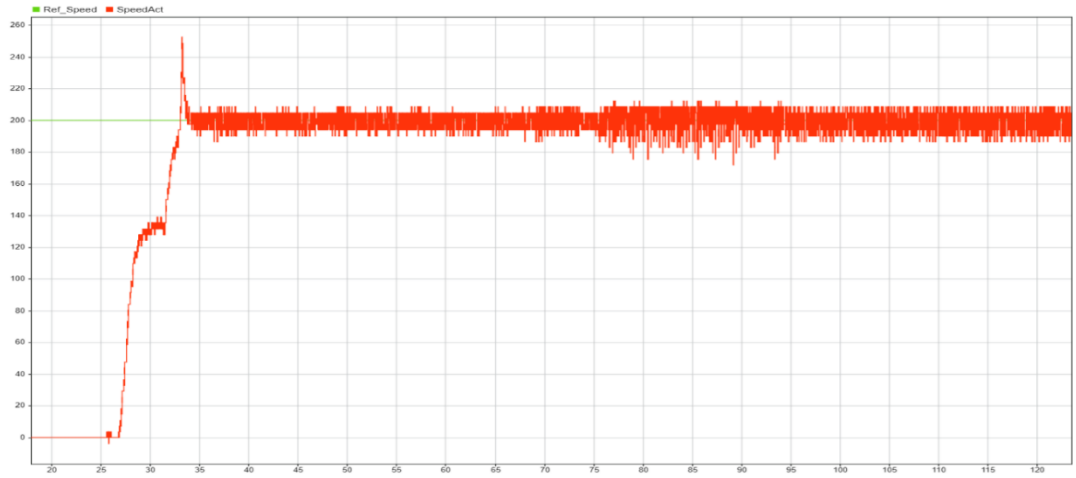


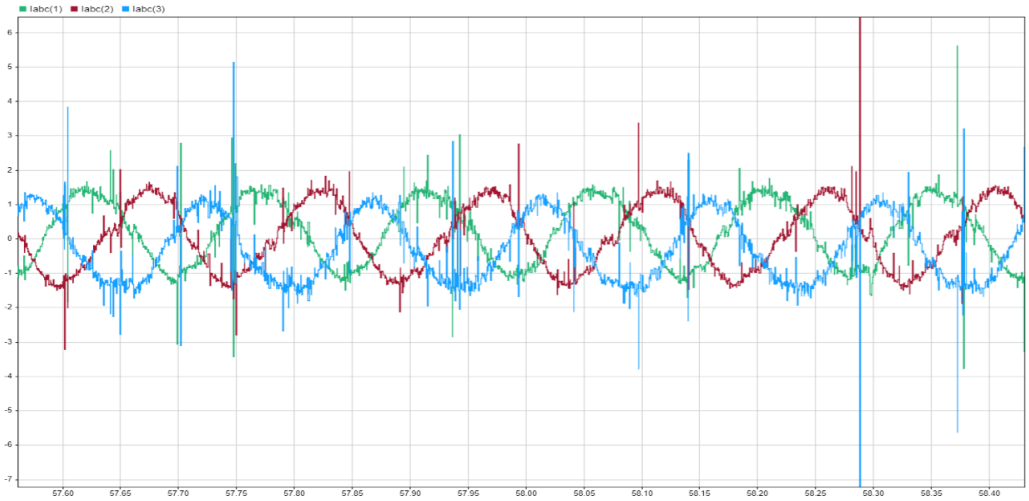
Figure 3.6 PWM Generation

3.4.Output results

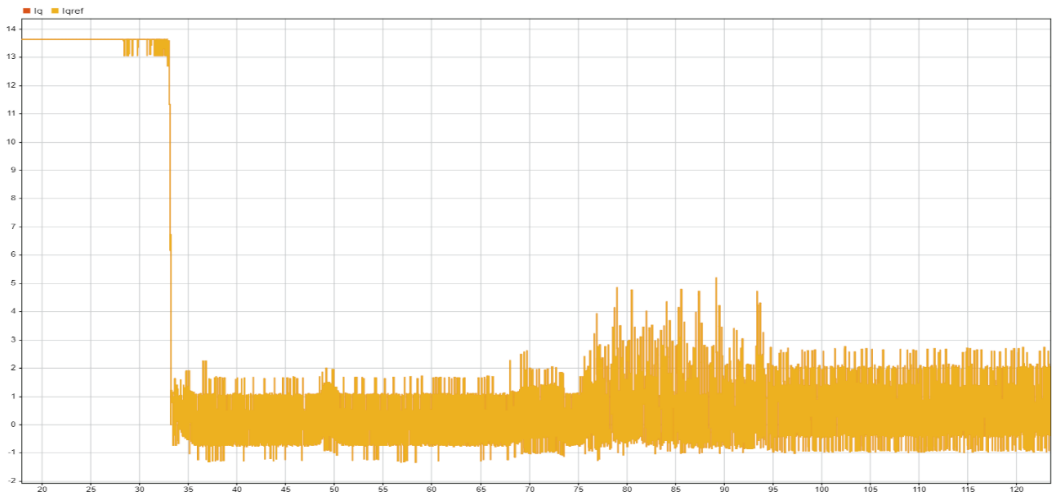
3.4.1. Speed vs reference speed



3.4.2. Stator current

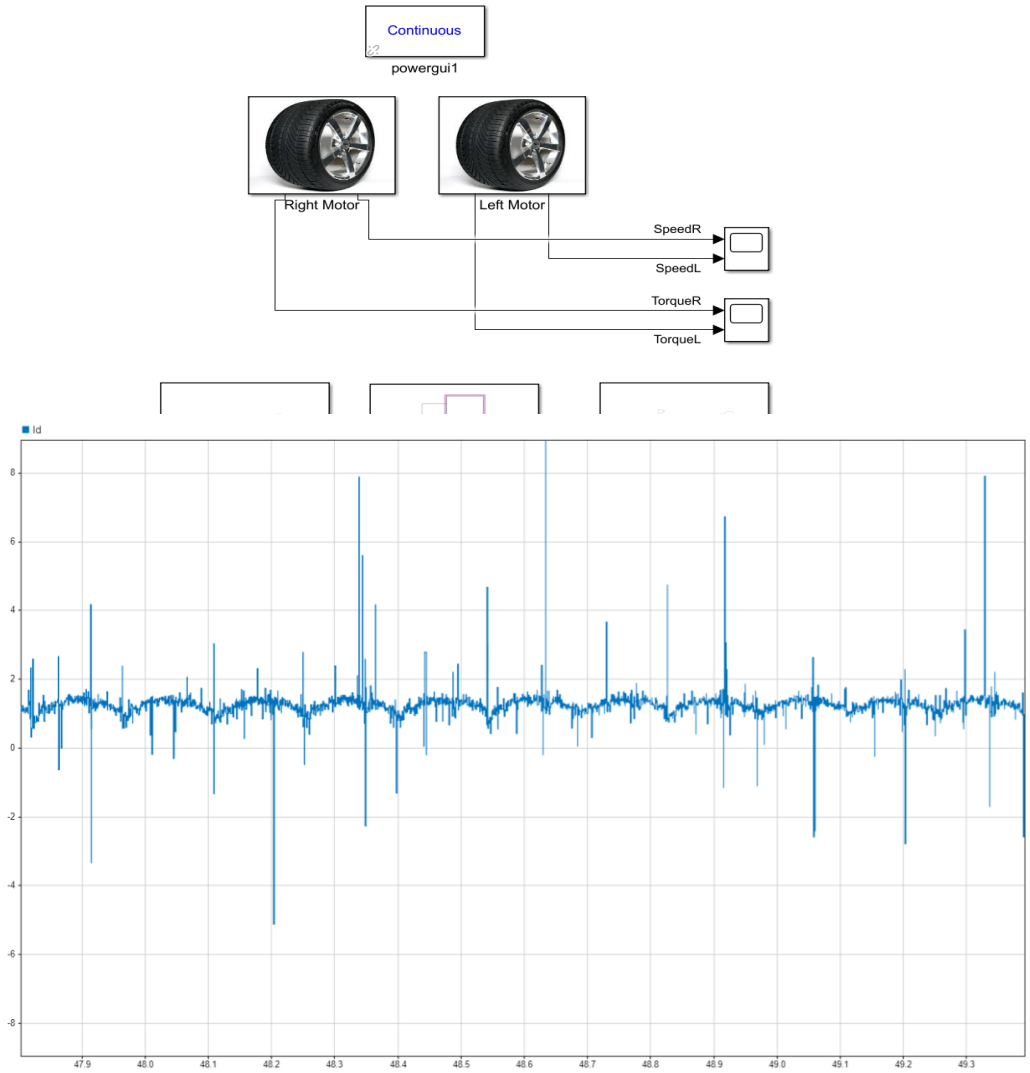


3.4.3. Quadrature Current



3.4.4. Direct Current

3.5. Complete Model



- **Complete Model:** The reference speed for the two motors shall be set to the velocity obtained from the Ackermann model, while the torque output from the torque vectoring system shall be utilized as the designated torque load for the motors.
- **Ackermann:** The Ackermann model comprises equations that can compute the reference speed for the two motors based on the input parameters of the vehicle speed and steering angle.
- **Torque vectoring:** The torque vectoring model employs the kinematic model to obtain the side slip angle and yaw rate, which are then utilized in the correction model to compute the Delta M (ΔM) required to adjust the torque output. The resulting value of ΔM is added to and subtracted from the required torque to get

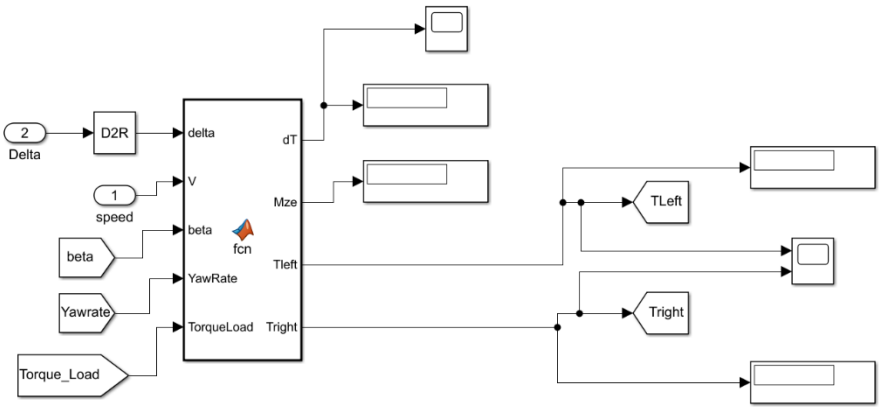


Figure 3.9 Correction Model

the torque load for the two motors. The input parameters for these models are the speed of the vehicle, steering angle, and the required torque.

- **Adjustment:** The adjustment process is utilized to determine the values of the vehicle speed, steering angle, and Torque load.
- **Right and Left Motors:** The model has a Motor, inverter, and FOC controller.

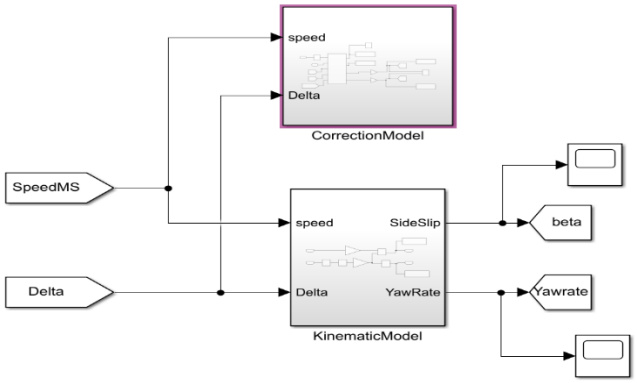


Figure 3.8 Torque Vectoring

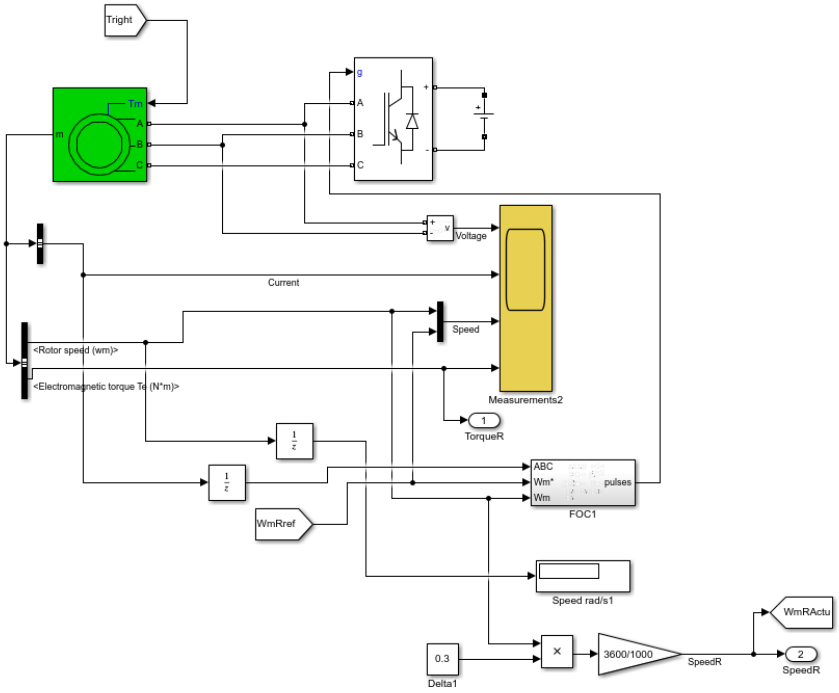
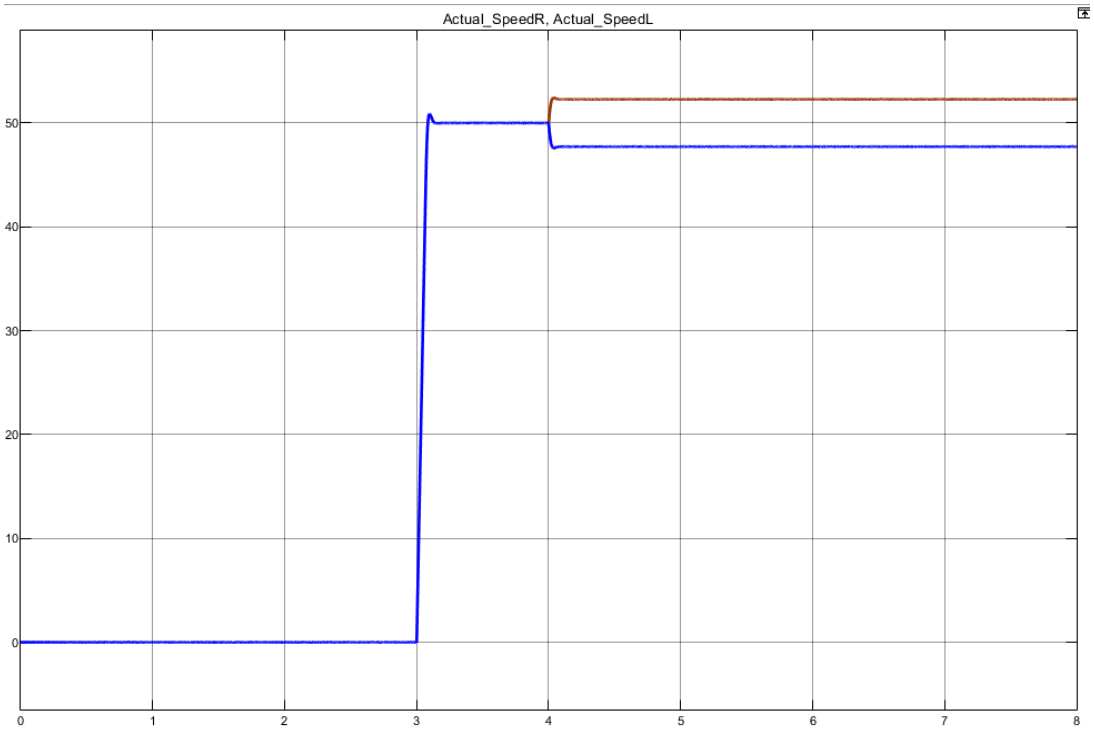


Figure 3.10 FOC Block for Right Motor

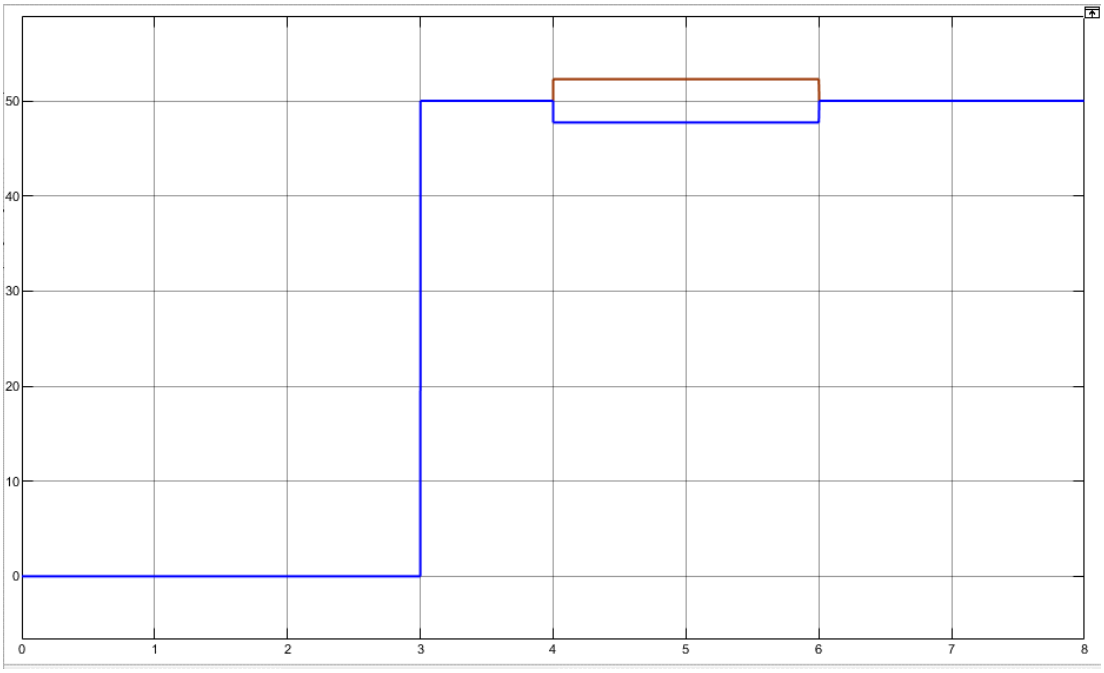
3.5.1. Simulation Result

3.5.1.1. $V = 50\frac{km}{hr} at\ t = 3, \delta[10]\ at\ t[4]$

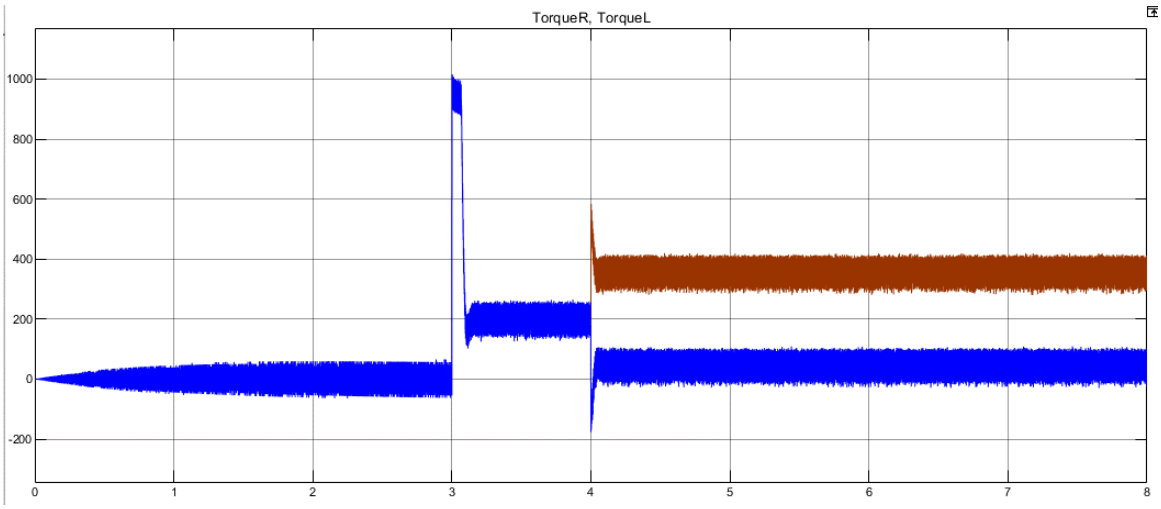
- Measured Speed of Right and Left motors



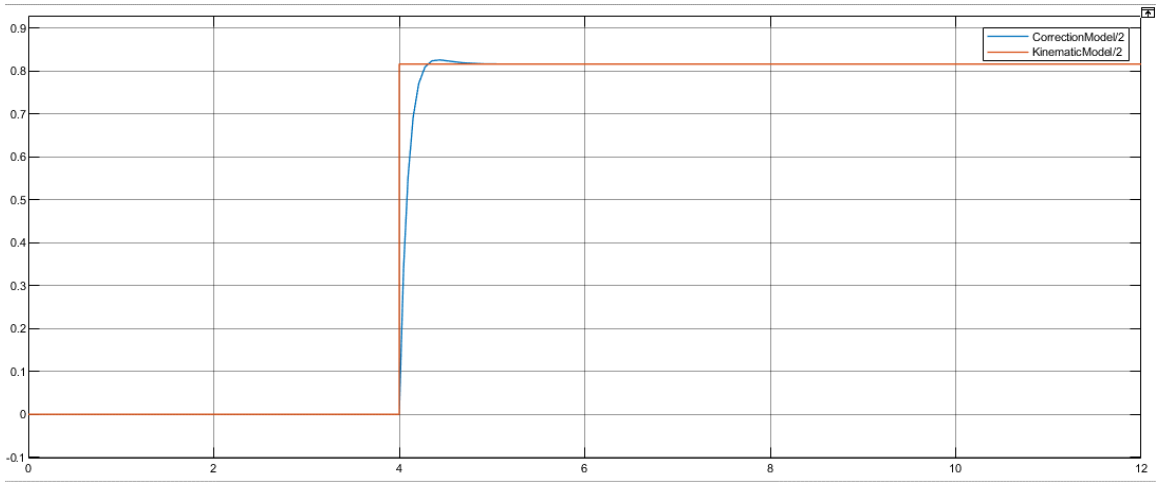
- Reference Speed of Right and Left motors



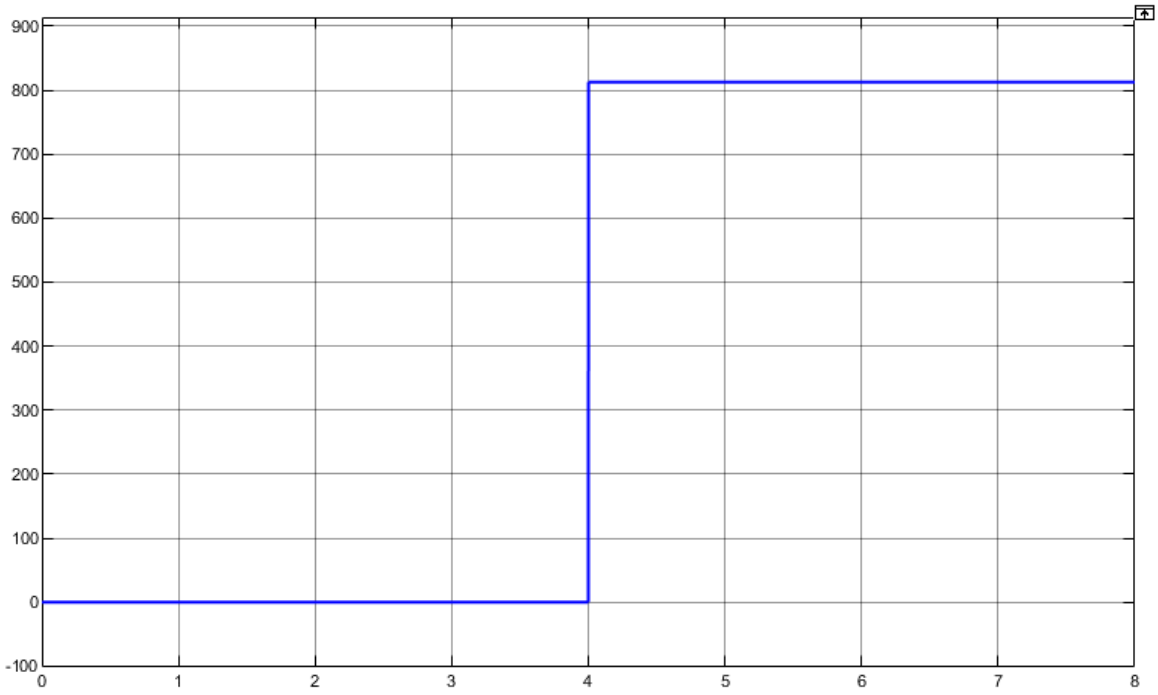
- Developed Torque of Right and Left motors



- Yaw Rate (Kinematic Model Vs Correction Model)

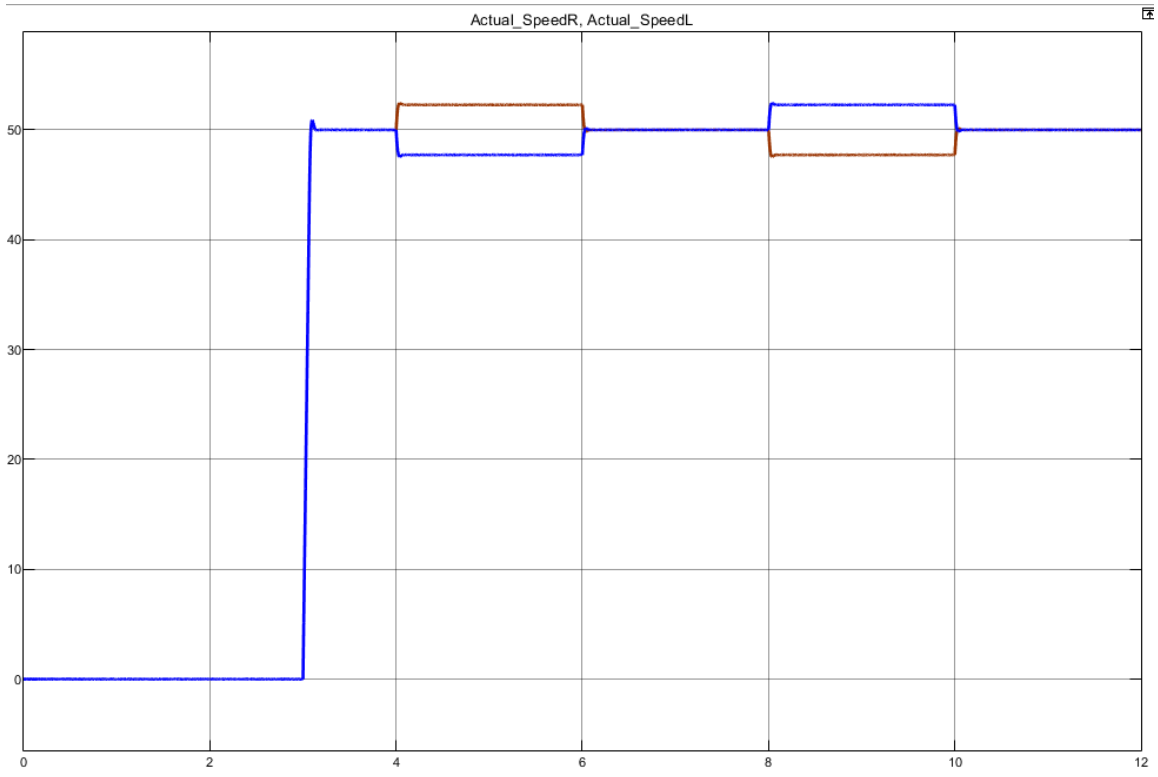


- M_{ze} (External Yaw Moment)

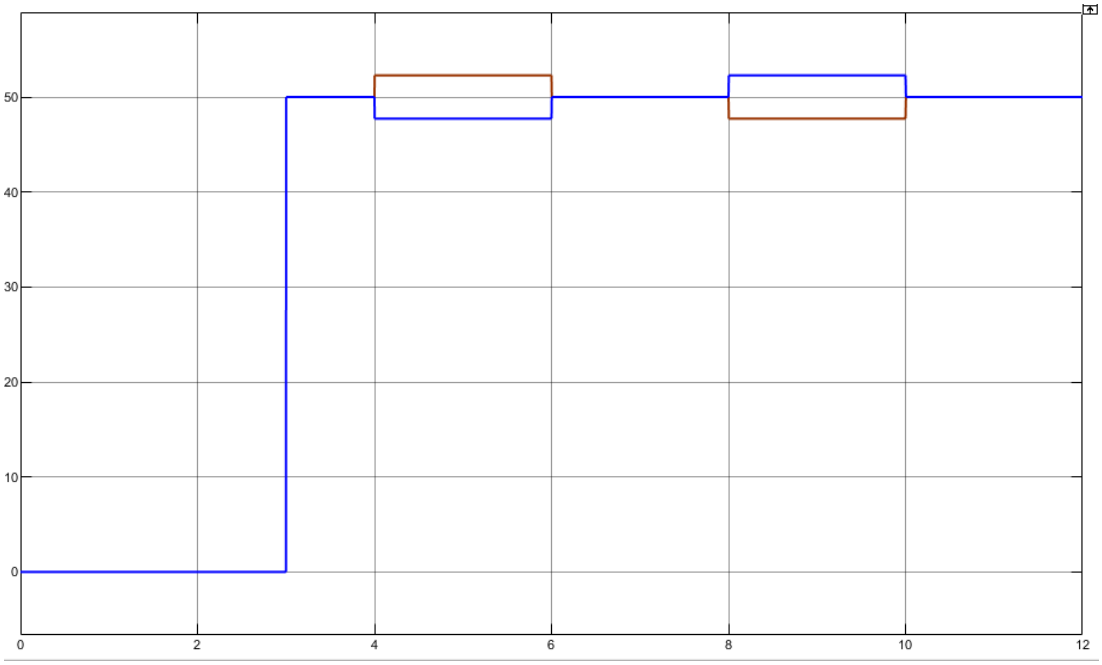


3.5.1.2. $\mathrm{V} = 50\,\frac{\mathrm{km}}{\mathrm{hr}}\,\mathrm{at\ t} = 3, \delta[0\ 10\ 0\ -10\ 0\ at\ t[0\ 4\ 6\ 8\ 10]$

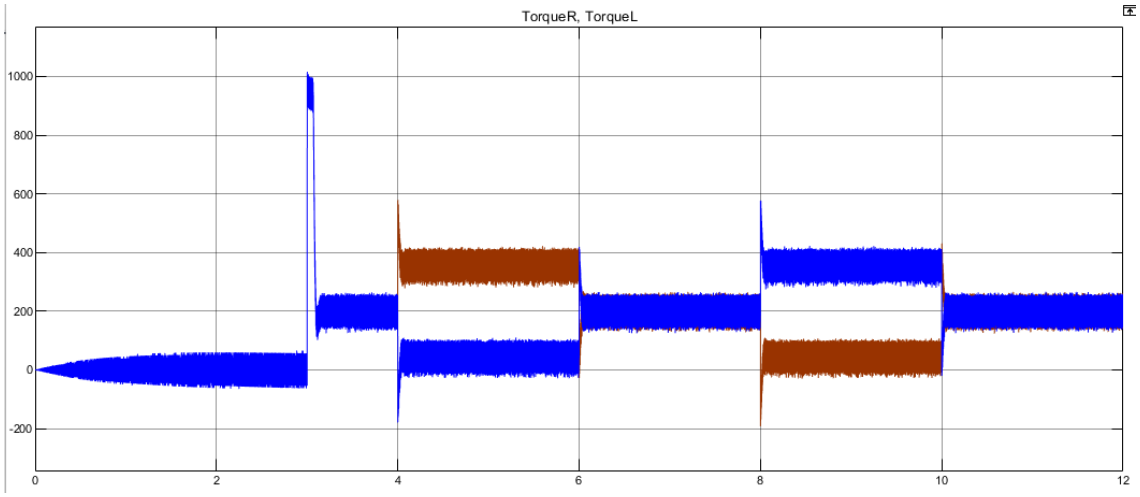
- Measured Speed of Right and Left motors



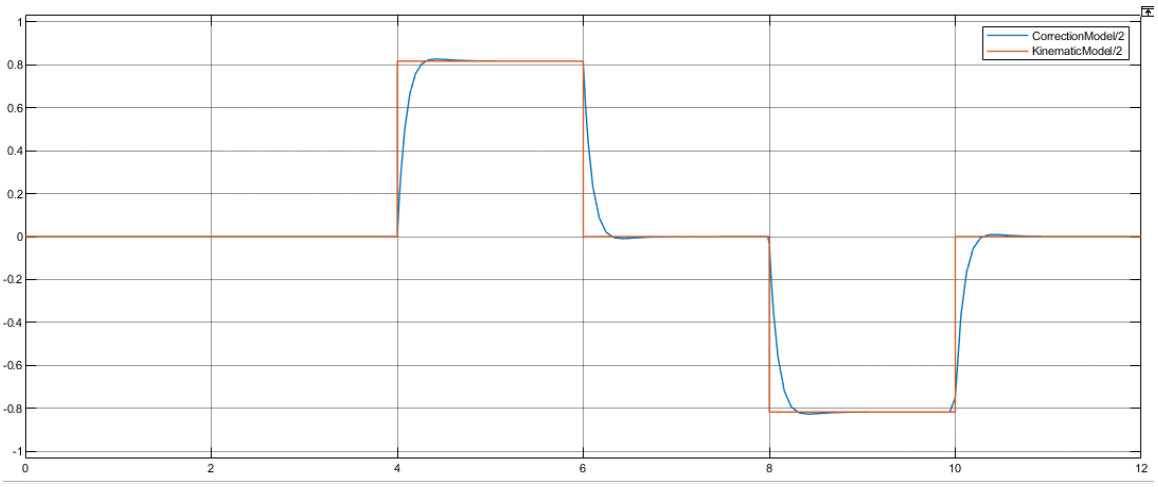
- Reference Speed of Right and Left motors



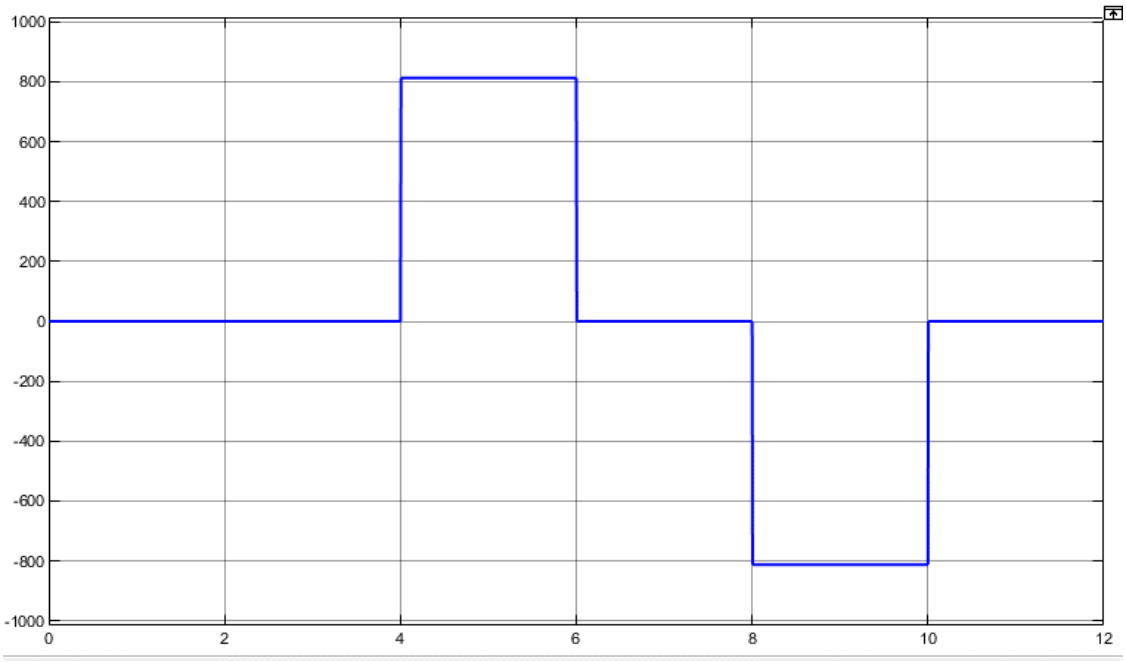
- Developed Torque of Right and Left motors



- Yaw Rate (Kinematic Model Vs Correction Model)



- M_{ze} (External Yaw Moment)



4. References

1. Mondek, M. (2018) Active torque vectoring systems for electric drive vehicles. thesis.
2. GENTA, G. (2016) Automotive chassis, volume 2: System design. SPRINGER.
3. Wight, W. (2019). Optimal direct yaw moment control of a 4WD electric vehicle.
4. He, R. and Yun, H. (2019) 'Electronic differential control of rear-wheel independent-drive electric vehicle', SAE International Journal of Vehicle Dynamics, Stability, and NVH, 4(1). doi:10.4271/10-04-01-0004.
5. Hartani, K. and Miloud, Y. (2010) 'Electric Vehicle Stability with Rear Electronic Differential Traction', EFEEA'10 International Symposium on Environment Friendly Energies in Electrical Applications, 4(1).
6. Bose, B. K.(2002) Modern Power Electronics and AC Drives. Upper Saddle River, NJ: Prentice-Hall
7. Mohan, N. (2014) Advanced Electric Drives: Analysis, control, and modeling using MATLAB/Simulink. Hoboken: Wiley.
8. Llorente, M.R. (2020) Practical control of electric machines: Model-based design and Simulation. Cham: Springer International Publishing.
9. Åström, K.J. and Hägglund, T. (2006) Advanced PID control. Research Triangle Park, NC: ISA-The Instrumentation, Systems, and Automation Society.
10. Wang, L. (2015) PID and predictive control of electric drives and power supplies using MATLAB/Simulink. Chichester: Wiley-Blackwell.
11. Sokola, M. (1998) Vector control of induction machines using improved machine models. Liverpool John Moores University.
12. Rossi, M. et al. (2022) Introduction to microcontroller programming for Power Electronics Control Applications: Coding with MATLAB and simulink. Boca Raton: CRC Press, an imprint of Taylor & Francis Group.