MARMARA UNIVERSITY

FACULTY OF ENGINEERING

CONTROL SYSTEM DESIGN OF CNC MACHINE TOOL FEED DRIVES

ÇAĞATAY KAYRAN, MEHMET EMİN AYDINOĞLU

GRADUATION PROJECT REPORT

Department of Mechanical Engineering

Supervisor

Associate Professor İbrahim Sina Kuseyri

İSTANBUL, 2022

MARMARA UNIVERSITY

FACULTY OF ENGINEERING

Control System Design of CNC Machine Tool Feed Drives

by

Çağatay Kayran, Mehmet Emin Aydınoğlu

10/06/2022- İSTANBUL

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

OF

BACHELOR OF SCIENCE
AT

MARMARA UNIVERSITY

Signature of Author(s) ....

Department of Mechanical Engineering

Certified By ....

Project Supervisor, Department of Mechanical Engineering

Accepted By..

Head of the Department of Mechanical Engineering

ACKNOWLEDGEMENT

First of all, we would like to thank our supervisor Associate Professor İbrahim Sina KUSEYRİ, for the valuable guidance and advice on preparing this thesis and giving us moral and material support.

Çağatay KAYRAN   Mehmet Emin AYDINOĞLU

January, 2022

CONTENTS

ABSTRACT

Control System Design of CNC Machine Tool Feed Drives

A typical motion control system manages position, velocity, torque, and acceleration of an axis. Often the machine has multiple axes whose position and velocity must be controlled in a synchronized fashion. Our aim in this project is to design one axis trajectory tracking position controller and evaluate the performance of the controller we designed in MATLAB/Simulink. In Simulink, we aimed to solve the vibration problem caused by environmental effects with trajectory planning and modeling.

SYMBOLS

$\ddot{\theta}_m$ **:** Motor shaft angular acceleration

$T_{load \to M}$ **:** Torque demand from the motor due to external loads reflected to motor shaft.

L_d **:** Direct-axis inductance

L_q **:** Quadrature-axis Inductance

A_{eff} **:** Complete pressure acts on the effective area

A_r **:** Land area

C_d **:** Coefficient of derivative control

F_0 **:** Suspension force at equilibrium

J_{total} **:** Inertia of all transmission elements + motor + load reflected to the motor shaft

L_k **:** Length of the capillary

R_c **:** Oil gap generates resistance

R_k **:** Restrictor flow resistance

T_{RMS} **:** Continuous Torque

T_{acc} **:** Torques required in the acceleration phase

T_{dec} **:** Torques required in the deceleration phase

T_{dw} **:** Torques required in the dwell phase

T_f **:** Viscous friction torque

T_m **:** Torque needed from the motor to achieve the motion

T_{run} **:** Torques required in the running phase

V_{BUS} **:** DC bus voltage

p_p **:** Supply pressure

r_k **:** Radius of the capillary

t_a **:** Duration of acceleration phase

t_d **:** Duration of deceleration phase

t_{dw} **:** Duration of the dwell phase

t_m **:** Duration of running phase

λ_{PM} **:** Rotor permanent magnet

ω_n **:** Natural frequency

J **:** Inertia of rotating parts

B **:** Viscous friction coefficient

F **:** Hydraulic suspense force

L **:** Land length

R **:** Insulation Resistance

b **:** Total width of the land

p **:** Pole number of the motor

v **:** Translational velocity

x **:** Table position captured by linear encoder

ζ **:** Damping ratio

η **:** Dynamic density measure of the canvas

ABBREVIATIONS

AC : Alternating Current

ADC : Analog Digital Converter

CAD : Computer Aided Design

CNC : Computer Numerical Control

CSI : Current Source Inverter

DAC : Digital Analog Converter

DC : Direct Current

DIN : Deutsches Institut für Normung (German Institute for Standardization)

HMI : Human Machine Interface

MOSFET : Metal Oxide Semiconductor Field Effect Transistor

NC : Numerical Control

PID : Proportional Integral Derivative

PWM : Pulse-width Modulation

UDI : User Defined Inputs

VFD : Variable Frequency Drives

VSI : Voltage Source Inverter

FFT : Fast Fourier Transformation

LIST OF FIGURES

LIST OF TABLES

1   INTRODUCTION

The main objective of CNC technology is to automate serial and error-free manufacturing by eliminating the need for human labor. For example, parts that are machined in one hour on a lathe can be produced in minutes thanks to CNC technology. For this reason, CNC machines are frequently used in the manufacturing sector today.

Characteristics of main spindle and feed drives for the CNC machine tool highly depend upon skillfulness of composing variable speed motors (AC or DC) and mechanical transmission elements. Original computer programs have been developed to enable interactive computer-aided design and analysis of various design variants of CNC machine tool drives.
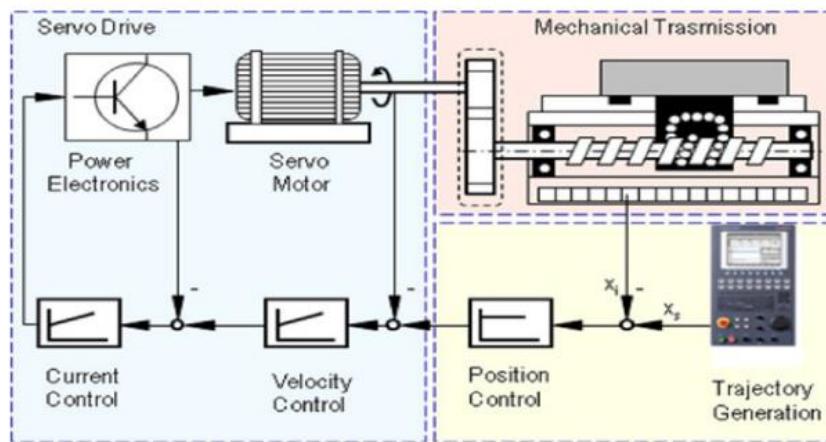
Figure 1.1 Physical components of a feed drive

Feed drives are used to place the machine tool factors carrying the slice tool and workpiece to the asked position; hence their positioning delicacy and speed determine the quality and productivity of machine tools. A general armature of feed drive tackle and its computer control structure are shown in (Figure 1.1). Feed drives are moreover powered by direct motors directly, or by rotary motors via ball screw and nut assembly as shown in (Figure 1.2). The drive train consists of a machine tool table resting on the companion and moved linearly moreover by a ball screw drive- nut or by a direct motor system. The ball screw may be connected to the rotary servo motor directly or via gear reduction for large machines. The motor is powered by amplifier electronics connected to a Computer Numerical Control (CNC) system as shown in (Figure 1.1). The table is deposited by the servo drives by following a line generation and control algorithm as shown in (Figure 1.3). An NC program generated in CAD/ CAM system is loaded to the CNC unit of the machine tool. CNC parses the NC program into tool path parts

1

which may correspond of direct, indirect, spline or other geometric movements. The feed rate entered in the NC program is combined with the acceleration and haul limits of the feed drives, and time stamped separate position commands are transferred to each drive servo by the real time line generation algorithm. The trajectory generation algorithm considers the kinematics of the machine in decoupling the spatial tool motion into each feed drive.
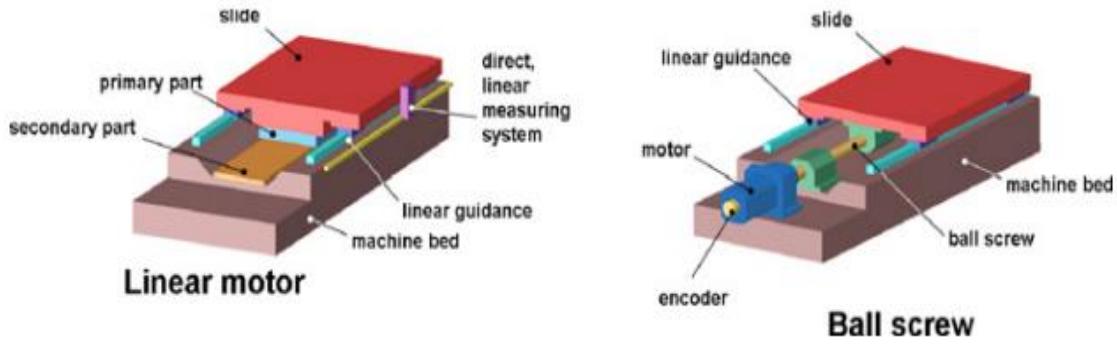
Figure 1.2 Linear and ball-screw drive mechanism

Ultramodern CNC units use haul nonstop, i.e., fifth order polynomials, to induce position commands at each separate time interval. The discrete position commands are processed by real time control laws of each servo drive, and the corresponding digital velocity commands are converted into electrical signals which are fed to the amplifier and motor of the drive. The speed and accuracy of positioning the machine tool are affected by the trajectory generation and control algorithms, mechanical drives and guides, amplifiers, motors and sensors used in each feed drive. Ongoing research challenges are also discussed in order to push the feed drive accuracy and performance to higher levels.
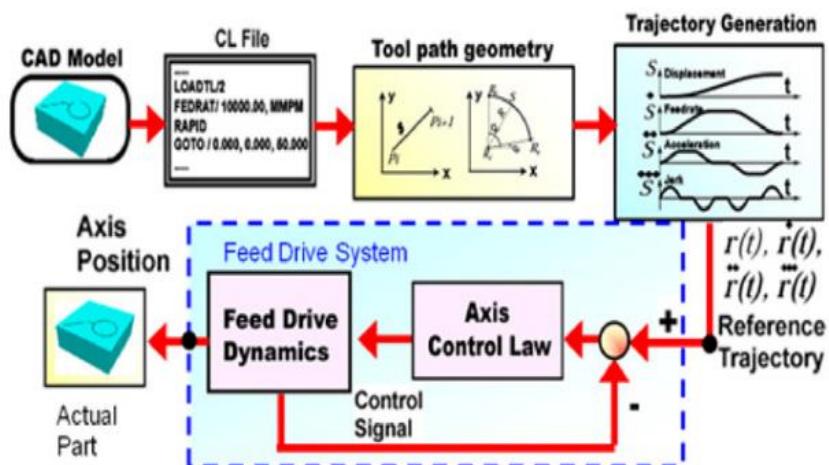
Figure 1.3 Feed drive control algorithms

2 COMPONENTS OF MOTION CONTROL SYSTEM

The complex, high-speed, high-precision control required for the multi axis coordinated motion is implemented using a specialized computer called motion controller. Working scheme as shown in (Figure 2.1), a complete motion control system consists of:

1. Human–machine interface
2. Motion controller
3. Drives
4. Actuators
5. Transmission mechanisms
6. Feedback
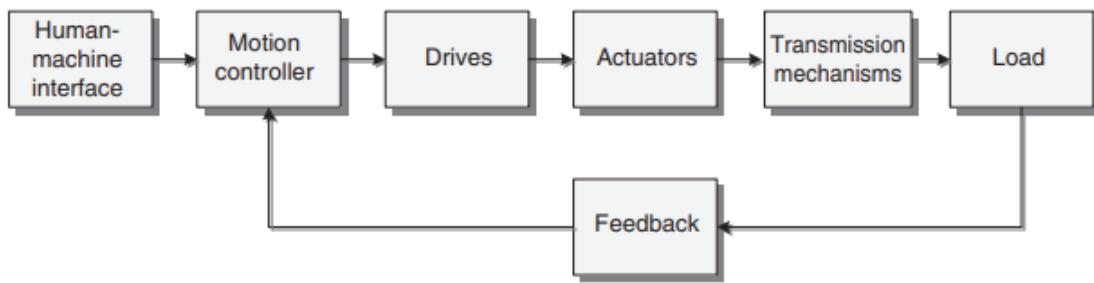
Figure 2.1 Components of a motion control system

2.1 Human-Machine Interface

An HMI, also known as Human-Machine Interface, is a device or software that allows the user to communicate with machinery and manufacturing facilities. It does this by translating large amounts of complex data into accessible information. In this way, the operator has all the necessary tools to control the production process.

The HMI is used to communicate with the motion controller. The HMI may serve two main functions: Operating the machine controlled by the motion controller and programming the motion controller.

Control panels as shown in (Figure 2.2) with pilot lights, push buttons, indicators, digital readouts, and analog gauges are common hardware-based HMIs to serve the purpose of operating a machine.

Figure 2.2 Human Machine Interface (HMI)

A computer is interfaced to the motion controller for programming purposes. The machine control programs are written, edited, downloaded, and tested using custom software provided by the controller's manufacturer. The software also includes features for testing motors, monitoring I/O signals, and fine-tuning controller gains.

2.2   Motion Controller

One of the most fundamental functions of a CNC machine is motion control. The motion controller is the "brains" of the system. It generates motion profiles for all axes, monitors I/O, and closes feedback loops. However, this motion must also be automatic, consistent, and accurate- and this is where motion-type CNC controllers fit in. They operate through two systems, namely, point-to-point control and contouring systems, which allow rapid, linear, and circular motions. The CNC controller is also in charge of regulating the feed rate and the degree of motion.

While the machine is running, it receives feedback from each axis motor. If there is a difference (tracking error) between the generated profile and the actual position or velocity of an axis, the controller generates correction commands, which are sent to the drive for that axis. The controller can also generate and manage complex motion profiles including electronic camming, linear interpolation, circular interpolation, contouring, and master–slave coordination.

Motion controllers are available in different form factors. The integrated form factor incorporates the computer, the drive electronics for the axes, and the machine I/O into a single unit. This unit is called motion controller or drive. In a modular system, the computer, the drives, and the machine I/O are separate units connected to each other via some type of communication link. In this case, just the computer is called the motion controller.

Figure 2.3 Motion Control Card for CNC Machine Tool Control

2.3   Drives

The feed drive is one of the most important parts of every CNC machines. The productivity and accuracy of the CNC machine tool highly depend on its characteristics. The feed drive's primary function is to move machine tool working parts (working table, tool unit, spindle unit, etc.) through machine axes. A separate feed drive is necessary for every machine axis. The command signals generated by the controller is small signals. The drive amplifies these signals to high-power voltage and current levels necessary to operate a motor. Therefore, the drive is also called an amplifier. The drive closes the current loop of the servo system. Therefore, it must be selected to match the type of motor to be driven. Although feed drives have very simple kinematics structures in general, their optimal design is a problem that consists of selecting servo motor and mechanical transmission elements that must satisfy some system requirements. The distinction between a drive and a controller has become increasingly blurred in recent years, as drives perform many of the complex functions of a controller. They must handle motor feedback and close not only the current but also the velocity and position loops.

Figure 2.4 AC Drive

2.4   Actuators

An actuator is a part of a device or machine that helps it to achieve physical movements by converting energy, often electrical, air, or hydraulic, into mechanical force. Defined simply, an actuator is a device that converts energy, which may be electric, hydraulic, pneumatic, etc., to mechanical in such a way that it can be controlled. The quantity and the nature of input depend on the kind of energy to be converted and the function of the actuator. Electric and piezoelectric actuators, for instance, work on the input of electric current or voltage, for hydraulic actuators, its incompressible liquid, and for pneumatic actuators, the input is air. The output is always mechanical energy.
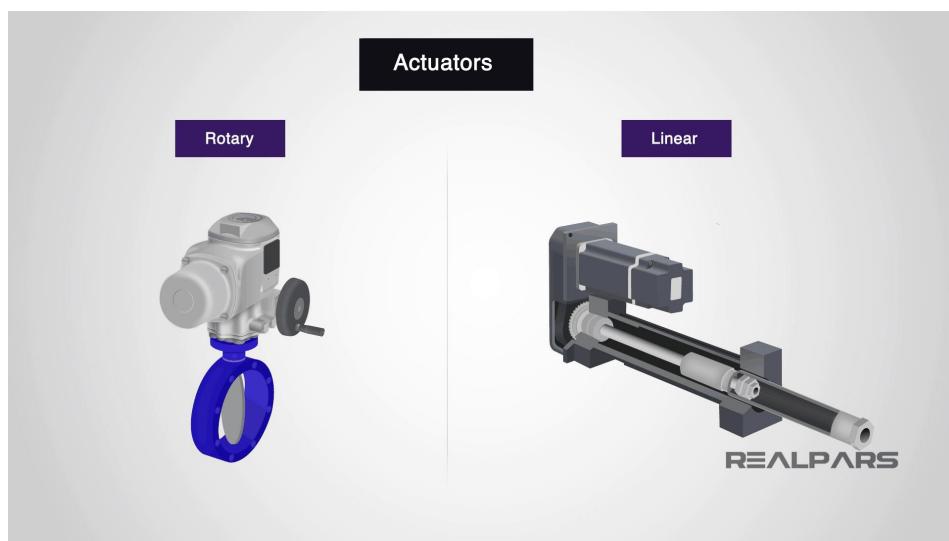
Figure 2.5 Linear and Rotary Actuators

Electric motors convert electrical energy into mechanical energy. Pumps, machine tools, fans, household appliances, disk drives, and power tools are just a few of the diverse applications of electric motors. The most common motors used in industrial motion control are three-phase AC servo and induction motors.

2.4.1   AC Induction Motor

An induction motor is an AC electric motor that produces the torque required to produce the electric current in the rotor by obtaining electromagnetic induction from the magnetic field of the resistor windings. The induction motor therefore generally does not require external excitation or self-stimulation, i.e., mechanical commutation, to transfer all or part of the energy from the resistor to the rotor, as in DC and large synchronous motors. There are two types of induction motor, ring motor and squirrel cage motor.

Three-phase squirrel-cage induction motors are generally used in industrial drives because of their robust, reliable and economical nature, while single-phase induction motors are mostly used for small loads such as household appliances. Although generally used in fixed speed service, the use of induction motors in variable speed service with variable frequency drives (VFD) is increasing. Variable frequency drives are particularly important in terms of energy

savings for current and future induction motors in variable torque centrifugal fan, pump and compressor load applications. Squirrel-cage induction motors are often used in both fixed speed and variable frequency drive (VFD) applications. Variable voltage and variable frequency drives are also used in variable speed services.

Figure 2.6 AC Induction Motor

2.4.2   AC Servo Motors

AC servo motor, also called brushed servo motor, is one of the servo motor types, and it is a type of servo motor preferred in projects that require high power and frequency value. The biggest distinguishing feature of AC servo motor types, namely brushed servo motors, is that the magnetic field remains constant in these motors and the coils are in motion.

Figure 2.7 AC Servo Motor

An AC servo motor's rotor is cylindrical and made of solid or laminated iron core and radially magnetized permanent magnets. Motors with 4-, 6- or 8-poles are common. Several magnet configurations are possible such as the surface-mounted and surface-inset magnets. By far the most common configuration is the surface-mounted version where the magnets are bonded on the surface of the rotor. Due to the simple construction, these motors tend to be relatively inexpensive. Ferrite magnets are the common choice as they are relatively inexpensive. The motor size can be reduced, and higher torque can be produced but these magnets are more expensive.

The stator of ac servo motor consists of two separate windings uniformly distributed and separated at 90°, in space. Out of the two windings, one is referred as main or fixed winding while the other one is called control winding.

A constant ac signal as input is provided to the main winding of the stator. However, as the name suggests, the control winding is provided with the variable control voltage. This variable control voltage is obtained from the servo amplifier. The stator is made of laminations, which are thin sheets of metal. The laminations are punched out of sheet metal, stacked, and welded together on the side to form the stator. The laminations reduce the energy losses due to Eddy currents compared to a solid core. Slot insulation materials and phase winding coils are inserted, and the coil ends are tied. Heat hardened varnish on the windings finishes the assembly.

2.5   Transmission Mechanisms

A transmission mechanism is used to connect the load to the motor of an axis. It helps meet the motion profile requirements. When a load is coupled to a motor through a transmission mechanism, the load inertia and torque are reflected through the mechanism to the motor. Most mechanical systems involve transmission mechanisms between the load and the motor. A transmission mechanism connects the load to the motor and helps meet the motion profile requirements.

2.5.1   Pulley and Belt

Pulleys use mechanical advantage, similar to levers, to lift up loads. Pulleys are wheel shaped with a groove that allows a cord to sit inside the groove. They can be used by hand or attached to a motorized winch to increase the amount of weight that can be lifted.

Pulleys are a simple and maneuverable way to move large objects. They are easy to transport to where they are needed and set up, but they do require somewhere stable to hang.

A single pulley changes the direction of force, making pulling down easier than lifting up. Single pulley systems are demonstrated in cranes, lifting a bucket from a well, raising a flag or adjusting window blinds. Even though there is no actual mechanical advantage with one pulley, it is referred to as having a mechanical advantage of one.

Belt drives transfer movement from one rotating pulley to another, each held on a shaft. Shafts and pulley wheels can be made out of any material, whereas pulley belts are generally made from a soft, flexible material such as rubber. Grooves on the pulleys and belts help them to grip and turn.
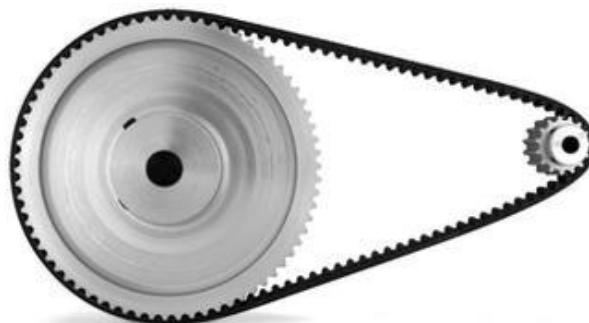
Figure 2.8 Pulleys and Belts

2.5.2 Lead screw and ball screw

Lead screws are widely used in converting rotary motion into linear motion. Two most common types of lead screws use ACME screw and ball screw. The ACME screw is difficult to back drive. In other words, motor can drive the load but the load cannot drive the motor (e.g., when the motor power is turned off in a vertical axis). The ACME screws can transmit large forces and therefore often are called power screws.

The ball-screw is currently the most commonly used in machine tool feed drives. A ball screw drive positioning stage is a precision linear component with a ball screw as its drive mechanism. A recirculating ball screw nut travels on a rotating threaded shaft with minimal friction, allowing the ball screw to support heavy axial loads. The assembly of the ball screw nut guided by a linear motion rolling guide acts as a slide table and travels along the helical raceway on the rotating shaft, converting the rotary motion of the ball screw shaft to linear motion.
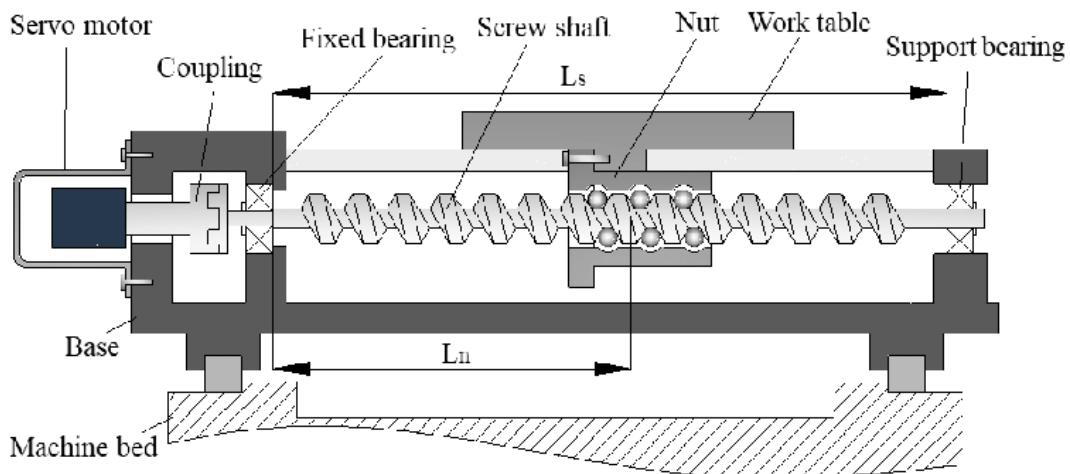
Figure 2.9 Typical structure of ball screw drive system

2.5.3 Rack and pinion

Rack-pinion-drives are using for long travel distances. By adding several racks together, very long feed travel can be realized. The resulting total stiffness of rack-pinion-drive is always independent of the length of travel distance. The total stiffness is dominated by the torsional stiffness of gear and pinion shaft as well as contact stiffness of rack-pinion-combination. The power transfer on the pinion is characterized by low revolutions and high torque. It needs additional gear steps. The whole drive line should be designed with high torsional stiffness and free of clearance. A feed transfer with clearance freedom in both movement directions could be

achieved through the separation of the pinion. (Figure 2.10) shows a feed drive with clearance elimination through pinions with helical gearing that combine with a rack. The lower pinion is moved axially through spring force on a spline-shaft shoulder, which allows both pinions bearing against opposite flank of the rack and compensating the gearing error.
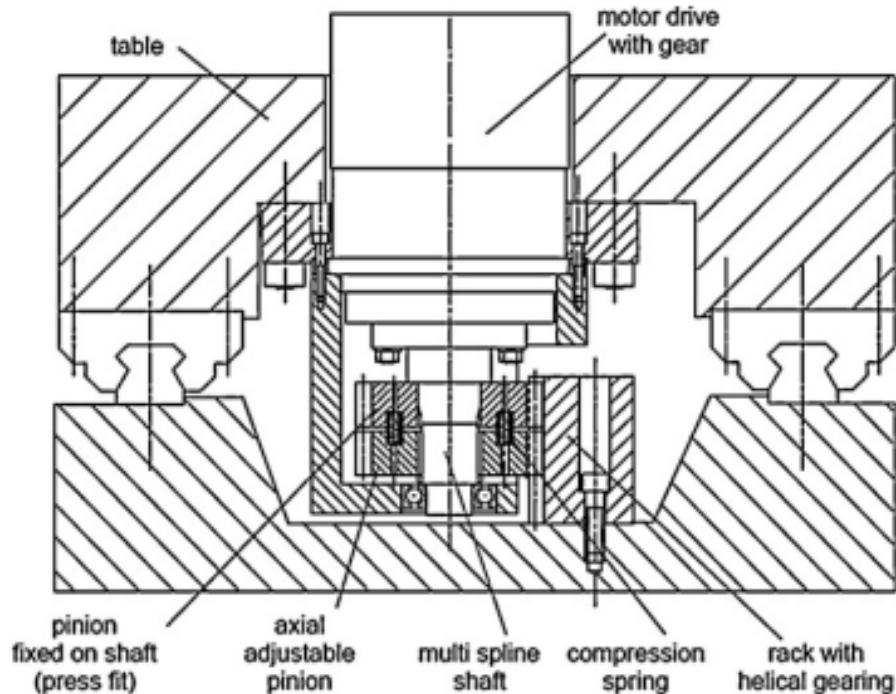
Figure 2.10 Clearance-free rack-pinion-system with split pinion.

2.6   Feedback

Feedback devices are used to measure the load's position or speed. In addition, feedback is used by the drive and controller to determine how much current should be applied to each phase of the motor. Most common feedback devices are resolvers, tachometers, and encoders. Encoders can be incremental or absolute. Selection of the feedback device depends on the desired accuracy, cost, and environmental conditions of the machine.

A different type of feedback is provided to the controller from detection sensors such as proximity switches, limit switches, or photoelectric sensors. These devices detect presence or absence of an object.
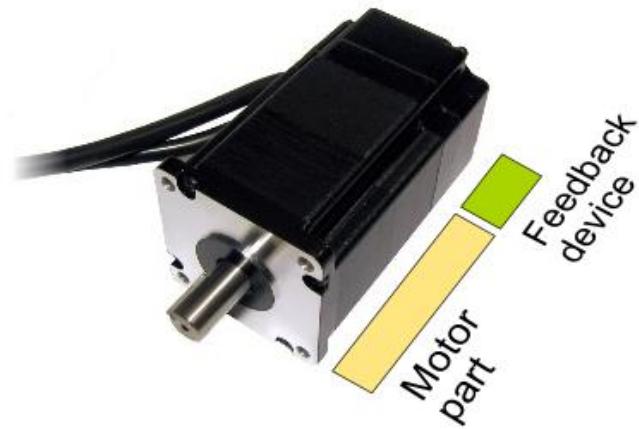
Figure 2.11 Feedback device

2.7 Machine Tool Guides

The roller-based guides have gained popularity due to their high performance and modular integration to machine tools. On the other hand, hydrostatic guides are preferred in the applications where higher accuracy, stiffness and damping are required. Aerostatic, magnetic or vacuum guides are used in the precision positioning applications where the external load is small.

The needed functional r for the longitudinal attendants are the following:

- Geometric delicacy since it's restated to the part directly.
- Stiffness to repel the machining process and the inertial forces with minimal distortion.
- Wear resistance and low disunion to avoid gripping, stick – slip marvels and aging of the shells.
- Toughness to withstand impacts from the machining process.

2.7.1 Friction guides

Friction guides have good damping, strength against impact loads and high load capacity. They are primarily used in speeds under 0.5 m/s. Uniform contact with minimum adhesion between the bed and slide is obtained by scraping and leaving uniform marks on the contact surfaces. The slide-ways are lubricated by 1 mm deep lubrication slots opened on the moving part of the guides. The guides can also be coated with few mm thick polymers in order to reduce the friction. Various configurations of friction guides are shown in (Figure 2.12).

Different pairs of materials are used to manufacture the friction guides. Cast, steel, bronze and some polymers are used as lubricant material. An important factor to ensure the controllability and smooth operation of the guide is to avoid the stick-slip phenomenon that occurs when the static friction coefficient is higher than the dynamic friction coefficient. Polymer-based materials containing additives that support lubrication have greatly reduced the sticking-slip problem.
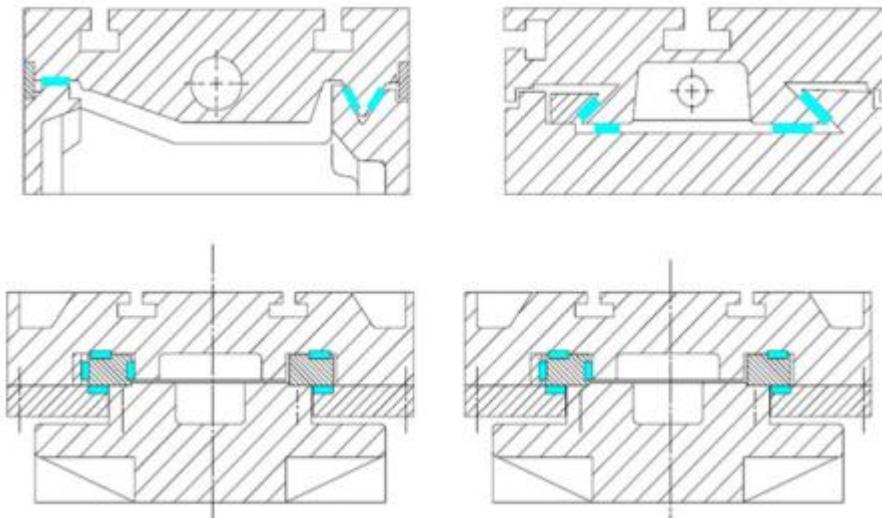
Figure 2.12 Configurations of friction guides

2.7.2   Rolling guides

Fixed roller-based guides and recirculating are widely used in existing machine tool applications. Fixed roller bearings are used when the stroke of the slide is short. The rolling elements can be steel balls, rollers or pins preloaded between two cages attached to the fixed and moving parts of the guides. They have low friction, a high load capacity, and stiffness, but they have little structural damping. Manufacturers produce recirculating rolling guides in a variety of sizes and load capacities. To speed up the design and assembly of machine tool drives, they can be equipped with integrated position sensors or racks.

The clamping systems for linear guide technologies are depicted in (Figure 2.13). The chamber between the two spring diaphragms is filled with compressed air to release the clamping action of the safety clamps. The clamping body can return to its relaxed position because the diaphragms deflect outwards. The brake blocks lift off the rail. When the chamber is not inflated, the diaphragms move back pushing apart the upper clamp body. With the horizontal strut acting as fulcrum point, the brake blocks are forced against the linear rail, thus clamping

13

the carriage. The operating pressure is 5–6 bar. The chamber beneath the spring diaphragm is filled with compressed air to activate the clamping action by clamps with air. The spring sheet is stretched and pushed upwards. The brake blocks are forced against the linear rail, clamping the carriage, with the horizontal strut acting as the fulcrum point. When the chamber is not inflated, the diaphragm returns to its relaxed position and the upper clamp body returns to its bended position. The brake blocks detach from the rail.
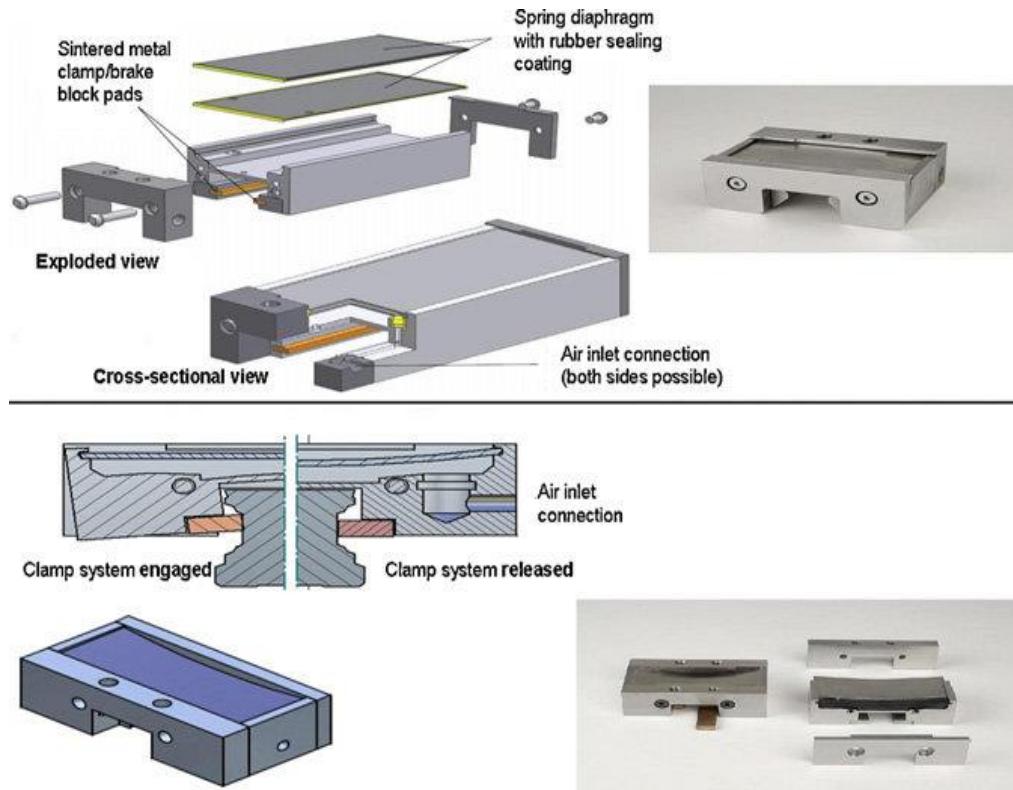
Figure 2.13 Clamping systems for linear guides

2.7.3   Hydrostatic guides

The sliding surfaces of the components are separated by cells pressurized with thin oil, hence stick slip and static friction are avoided in hydrostatic guides. Oil is released to the surrounding 'land' as it loses pressure. The distance between the 'land' and the surface which slides over it is the 'oil gap', h, which is about 10–40 mm. The oil gap generates resistance, Rc against the flow of the fluid, Q, from the oil cell to the outside.

The oil flow resistance along the gap can be estimated as:

$$R_c = \frac{\Delta p}{Q} = \frac{12\eta L}{bh^3}$$

2.1

Where η is the dynamic viscosity coefficient of the oil, L is the land length and b is the total width of the land along its perimeter. The pressure grade along the length of the land can be assumed to be direct, and the pressure acts on over to one half of the length of the land for original force vaticination. The complete pressure acts on the effective area (A_{eff}). The hydraulic suspense force, F, is calculated as follows:

$$F = P_c A_{eff} \qquad\qquad 2.2$$

The hydrostatic attendants are designed with several cells, similar that they can support out-center forces and moments. Each cell is supplied at a different pressure, in order to repel forces at varying operating conditions. Generally, a single pump is used with restrictors to supply each cell at the applicable pressure. The pressure in a cell is estimated as:

$$p_c = p_p \frac{R_c}{R_k + R_c} \qquad\qquad 2.3$$

Where p_p is the force pressure and R_k is the restrictor inflow resistance. The restrictors are erected in the form of capillaries where the resistance depends on the density of the canvas in the cells. The resistance to the canvas inflow of a capillary is estimated as:

$$R_k = \frac{8\eta L_k}{\pi r_k^4} \qquad\qquad 2.4$$

Where L_k and r_k are the length and compass of the capillary, independently. The short restrictors should have a veritably small periphery to give the necessary resistance. This periphery is limited by the size of suspended patches in the canvas which can block the capillary. The use of short capillaries is also limited because the design is veritably sensitive to their periphery; the resistance depends on the fourth power of the periphery. Capillaries with larger compasses and a longer length are also used, which tend to be in spiral form.

The restrictor automatically increases the cell pressure (Equation 2.3) and tends to drop the canvas gap and increase the inflow resistance (Equation 2.1) when external forces are applied. The canvas gap is kept nearly stable near the equilibrium gap height, h. The stiffness of the system is formed from the pump, restrictor and cell at the equilibrium state of the gap. In case of a cell supplied through a capillary by means of a pump which operates at a constant pressure, the performing stiffness is,

$$K = 3\frac{F_0}{h_0}\frac{R_k}{R_k + R_{cD}} \qquad\qquad 2.5$$

Where F_0 is the suspense force at equilibrium. As the stiffness depends on the cargo supported by the cell, hydrostatic attendants with 'retaining plate' are used in order to apply high preloads. In addition to gaining stiffness, the companion can absorb loads in both directions.

The hydrostatic attendants give much advanced damping values than the comber- grounded attendants against movements vertical to the cell. The source of the damping is the disunion force that the canvas lamella presents on sliding. In case of movements resemblant to the cell, the damping is reduced because there's no relegation of fluid. The temperature changes in the canvas affect the density, hence the damping and lifting force of the system.

The temperature can be estimated from the total work done by the pump and disunion forces. The work carried out by the pump is:

$$W_p = \frac{Q p_p}{\varepsilon} \qquad\qquad 2.6$$

where ε is the pump efficiency. The source of the friction work (W) is the translational motion of the table on the guides. The shear stress of the oil is:

$$\tau_r = \eta\frac{v}{h} \qquad\qquad 2.7$$

where v is the translational velocity. The work carried out to overcome the hydrodynamic friction can be determined as

$$W_r = \tau_r A_r v = A_r \eta\frac{v^2}{h} \qquad\qquad 2.8$$

Where A_r is the land area.

The dimensioning of a hydrostatic companion is fairly complex due to the strong reliance on the temperature rise of the canvas throughout the hydraulic circuit.

There are several challenges in optimal design of hydrostatic attendants. It's asked to have as small canvas gaps as possible. Still, the manufacturing delicacy, elastic distortions of the slides, the loss of canvas density due to temperature rise, and poor regulation of canvas inflow and pressure from a central pump may dwindle the gap between the sliding shells. Especially, the

canvas density and temperature rise are largely coupled; hence the fine modeling of the commerce between the two requirements to be iterative in order to prognosticate the equilibrium of their countries. A comprehensive fine model of the hydrostatic companion design, which predicts canvas circuit dynamics (i.e., pressure, inflow haste, normal and disunion force distribution), inflow energy, temperature, canvas density, elastic distortion of sliding shells, and vaticination of dynamic canvas cell gap and its stiffness could be largely useful tool for machine tool developer.
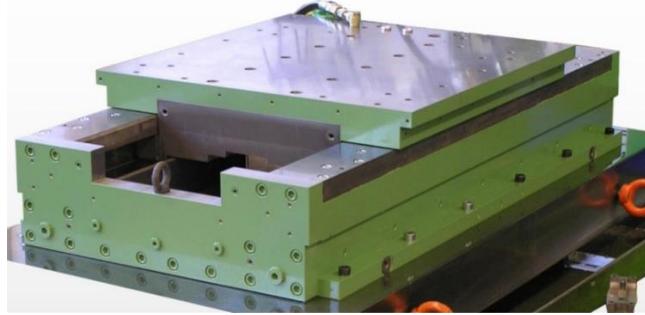
Figure 2.14 Hydrostatic guides

2.8   Sensors and Control Devices

Along with the motion controller, motion control systems make use of a variety of sensors and control components. Sensors are electronic devices that detect or measure physical quantities such as position, temperature, or pressure. A sensor's input is converted into a functionally related output. The physical quantity is the input, and the output is usually an electrical signal. There are many types of sensors used in motion control applications. Central to the operation of any motion control system is the need to measure the position and speed of the load in motion. The most common sensor for this purpose is an optical encoder.

Control devices are used in building user interfaces and to govern the power delivery to electrical loads such as motors. Devices such as push buttons, selector switches, and pilot lights are commonly used in building the user interface on the control panel of an automated system. Devices such as contactors and overload relays are used in control circuits to operate high voltage motors.

The absolute position of the feed drive must be measured in order to precisely position the tool on the workpiece. Optical encoders in machine tools are used to directly measure table position. Laser interferometer methods are used in precision machines that make optics, electronic circuits, or machines.

17

Optical encoders are based on the principle of transmitting or reflecting light by a glass or metal grating. A light source and photo detector array are used to sense the position of moving encoder disk or scale containing equally spaced reflective grating.

The absolute encoders have binary coded absolute positions. Incremental encoders have an additional reference mark which allows the controller to track the absolute position by counting the equally spaced marks relatively to the reference mark.

In case of a linear encoders with equivalent pitch spaces, the scanning unit moving in the direction of motion illuminates the scale at the measurement point. Motion between the scale and the scanning unit is evaluated through a scanning grid by means of photo elements.

The servo controller requires the velocity of the feed drive for tracking and damping of the table motion.
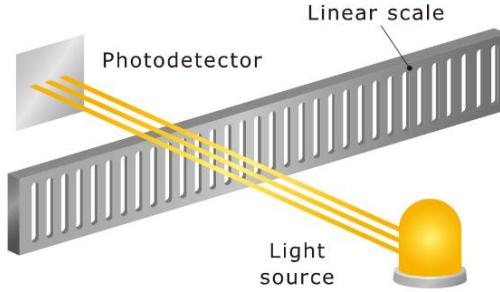
Figure 2.15 Optical Encoder

It is most common to estimate the velocity by digitally differentiating the position measurements obtained from the encoders. The accuracy of velocity estimation is determined by encoder resolution, quantization, speed, and harmonic error-frequencies. When the velocity change is too small in a very short digital integration interval, i.e., servo control interval, the velocity prediction becomes highly inaccurate and noisy. To smooth the velocity estimation, low pass and FIR filters are used, which adds unwanted phase delay to the drive controller.

In control laws, acceleration feedback is used to dampen structural dynamics and inspect the actual trajectory of the feed drive. The acceleration can be measured directly or indirectly by taking the second digital derivative of the position measurements.

The current of a servo motor is used in the first loop of a cascade controller, where the given value for the force or moment in form of the current is controlled via a PI controller. The current is also used to compensate friction and cutting force disturbances.

3 SELECTION OF AC SERVO MOTOR AND BALL-SCREW DRIVE

3.1 Considerations When Choosing an AC Servo Motor

There are many reasons why we choose AC Servo Motor for our motion control system. Servo motors are indispensable in industrial production. Since they have a closed-loop operating system, they are in constant communication with the system thanks to the feedback they provide. At the same time, servo motors with high speed and torque can be used in all kinds of applications.

We have also listed the reasons for choosing an AC Servo Motor in more detail below:

- AC servo motors can be used easily in projects that require high power and frequency.
- AC servo motors, which have a high-energy magnetic field, provide a great advantage in terms of efficiency in the areas where they are used.
- It allows for fast and rapid positioning in applications requiring motion control.
- AC servo motors, which are a fast type of motor, continue to work without disturbing their stability even under difficult working conditions.

The features of our AC Servo Motor that we have chosen for our project are shown in the (Figure 3.1)

750W
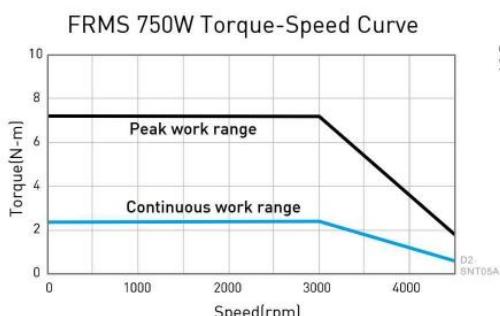
Parameter	Symbol	Unit	FRMS752☐☐08☐
Drive Input Voltage	V	V	AC220
Rated Power	W	W	750
Rated Torque	Tc	N.m	2.4
Rated Current	Ic	A(rms)	5.1
Peak Max. Torque	Tp	N.m	7.2
Peak Max. Current	Ip	A(rms)	15.3
Rated Speed	ωc	rpm	3000
Max. Speed	ωp	rpm	4500
Torque Constant	Kt	N.m / Arms	0.47
Back EMF Constant	Ke	Vrms / krpm	28.4
Resistance (line to line)	R	Ω	0.813
Inductance (line to line)	L	mH	3.4
Inertia of Rotating Parts (with brake)	J	kg-m²(×10⁻⁴)	1.4(1.46)
Weight (with brake)	M	kg	2.66(3.32)
Motor Insulation Grade			Class A (UL)
Motor protect			Total enclosed, self-cooled, IP65 (Except for shaft and connector)
Insulation resistance			10MΩ, DC500V
Insulation voltage resistance			AC1500V, 60 second
Brake specifications (Note 1)			
Static friction torque (Minimum)	Tb	N.m	2.4
Magnetizing current	Ab	A	0.358A
Brake input voltage	V	V	DC24±10%
Suction time (Maximum)	to	ms	45
Release time (Maximum)	tr	ms	10

Note1 Brakes are for maintaining object stop. Do not apply for deceleration, dynamic brakeing or emergency stop. Brake suction and release times vary with different circuitries, please note the actual operation delay time during application.

■ **Torque-Speed Curve**

■ **Dimensions**
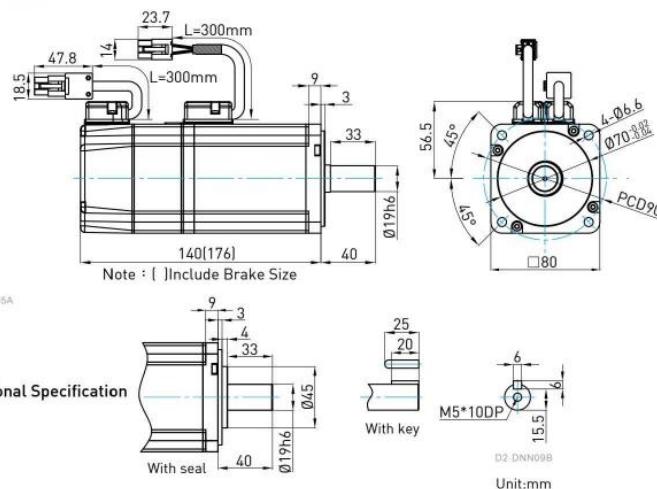
Figure 3.1 AC Servo Motor features

20

During selection of an AC Servo motor, we paid attention to details such as size, speed, torque, continuous torque, maximum torque, encoder resolution, ambient conditions.

We considered all the stages in detail before the implementation. We selected a level of product to match our calculations made during the design phase.

When the load is connected to the motor, the rotation speed of the motor will decrease at a certain rate. This angular shift in rotational speed is called "Torque". This is an event that will directly affect our system. Deciding torque on a system with a known maximum load weight requires mathematical calculation.

This mathematical operation is:

$$T_m = J_{\text{total}}\,\ddot{\theta}_m + T_{\text{load}\to\text{M}}$$
<div align="right">3.1</div>

Where T_m is the torque needed from the motor to achieve the motion; J_{total} is the inertia of all transmission elements + motor + load reflected to the motor shaft; $\ddot{\theta}_m$ is the motor shaft angular acceleration; and $T_{\text{load}\to\text{M}}$ is the torque demand from the motor due to all external loads reflected to the motor shaft.

The parameter we call Continuous Torque is the average of torque calculated over time during a full rotation. You can also see it as RMS Torque during motor selection.

The mathematical formula we used to calculate the RMS Torque is also given below:

$$T_{\text{RMS}} = \sqrt{\frac{T_{\text{acc}}^2 \cdot t_a + T_{\text{run}}^2 \cdot t_m + T_{\text{dec}}^2 \cdot t_d + T_{\text{dw}}^2 \cdot t_{\text{dw}}}{t_a + t_m + t_d + t_{\text{dw}}}}$$
<div align="right">3.2</div>

Where T_{acc}, T_{run}, T_{dec}, and T_{dw} are the torques required in the acceleration, running, deceleration, and dwell phases of the motion, respectively. Similarly, t_a, t_m, t_d and t_{dw} are the duration of each phase. The dwell torque may be zero if the axis is stopped and is not working against any external force during the dwell. It may also be nonzero such as in case of an axis with vertical load where torque must be applied to hold the load even though the axis is stopped.

Maximum Torque is the highest point of torque required at any point during the rotation of the motor under any load. After calculating our maximum torque, we made our motor choice according to the value we found.

The torque speed curve of a load is a plot showing its torque requirement on the y-axis versus

the speed on x-axis. It's the torque requirement of the load from zero speed to full speed.
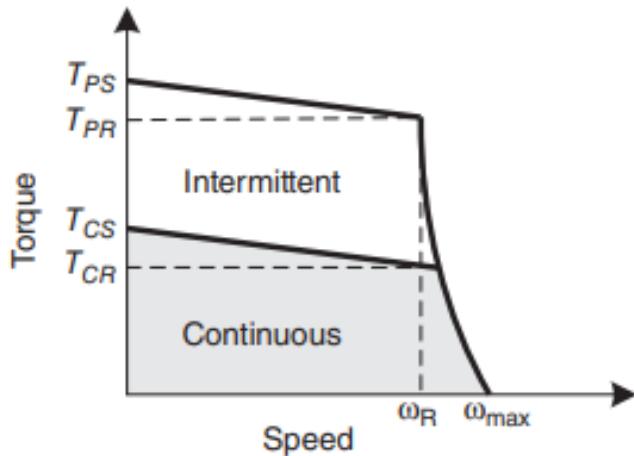
Figure 3.2 Torque–Speed Curves for AC Servomotors

The torque curve of the AC Servo Motor we have chosen shows two modes of operation: continuous and intermittent. The continuous operating zone contains all combinations of torque and speed that the engine can produce indefinitely. The intermittent zone shows much higher levels of torque the engine can deliver, but only for very short periods of time. Maximum speed ω_{max} is the speed of the motor at full supply voltage and no load. As can be seen in the graph of the engine we selected, this value is the maximum speed of our engine.
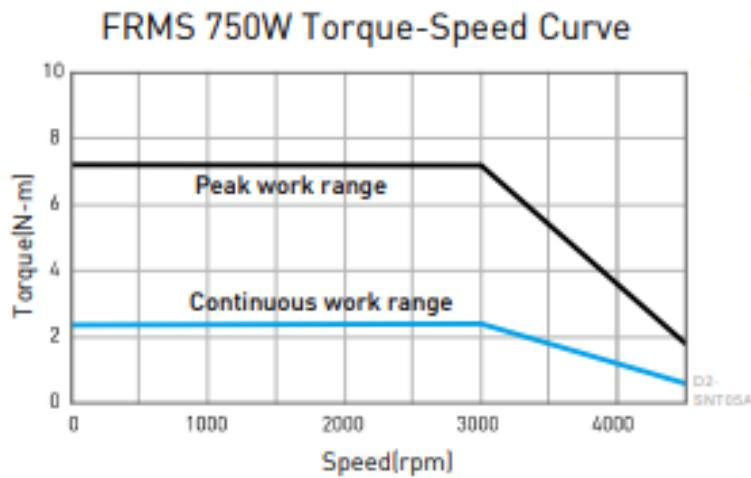
Figure 3.3 Torque-Speed Curves for our AC Servo Motor

3.1.1 NEMA Insulation Class

The purpose of NEMA motor insulation classes is to describe the ability of motor winding insulation to cope with heat. There are four electric motor insulation classes currently in use: A, B, F, and H (but there are also classes N, R, and S). Our motor's insulation class is A.

Class A Insulation:

- Maximum Temperature Rise: 60°C

- Hot-spot Over Temperature Allowance: 5°C

- Maximum Winding Temperature: 105°C

3.2 Considerations When Choosing a Machine Drive

One of the first elements to consider when choosing positioning stages is the drive mechanism. The drive mechanism we choose greatly affects the performance of our machine. Ball screw and lead screw are most commonly used as machine tool feed drives. There are many advantages to choosing a ball screw instead of a lead screw.
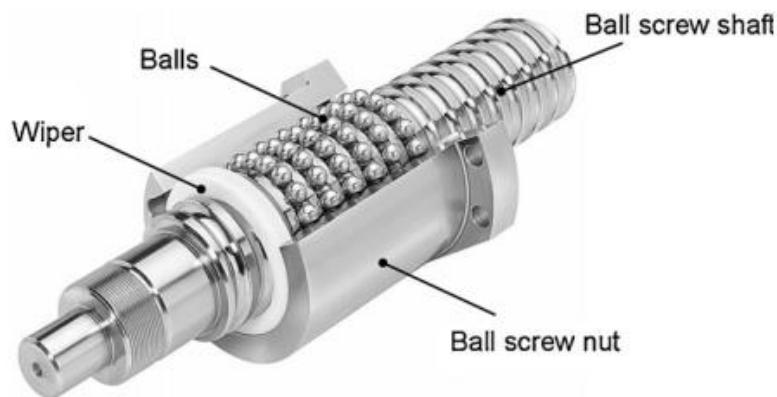
Figure 3.4 Structure of ball screw system

Ball screw drives have a high efficiency and thus low heating, wear, and service life without a stick–slip effect. The ball-screw drive consists of a screw with thrust bearings at both ends and a nut with recirculating balls. (Figure 3.4). The nut is connected to the table. One end of the ball-screw is either attached to a rotary motor directly or through gear/belt speed reduction mechanisms. The nuts are preloaded to avoid backlash by adjusting the spacer, creating offset

between the leads or using oversized balls as shown in (Figure 3.5). It is rather difficult to grind the pitch at uniform intervals, and the pitch errors are transmitted as position errors unless they are compensated. The design, calculation and acceptance terms of ball screw
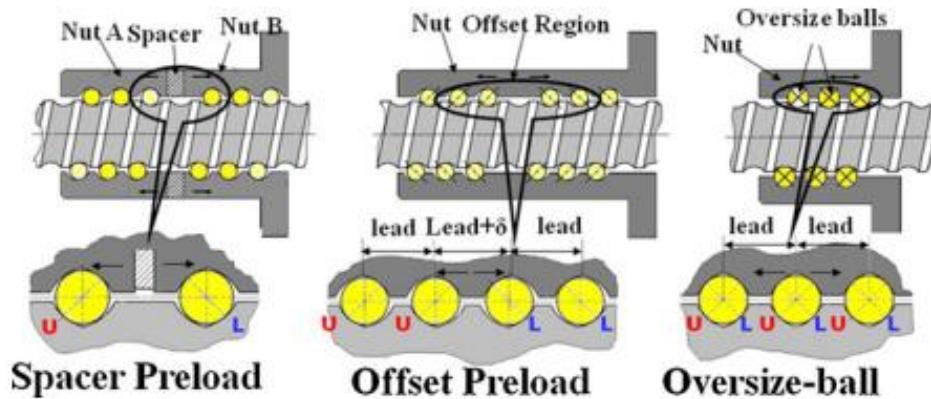
Figure 3.5 Ball-screw and nut mechanism

drives are described in DIN standards and. Ball screw drives with a length of approximately 12 m may be used in machine tools with long travel strokes. Depending on the application, the screw diameter and pitch may vary between 16 and 160 mm, and 5 and 40 mm, respectively. The current ball-screw drives can deliver up to 100 m/min travel speed with 2 g acceleration. Optimizations of the ball screw drive design, the coating of balls to reduce the friction and wear, deflection and nut preload control, led to significant increases in the speed and accuracy performance of ball screw drives. The balls roll between the guide slots of the screw and nut based on either the internal or external recirculation design principle shown in (Figure 3.6).
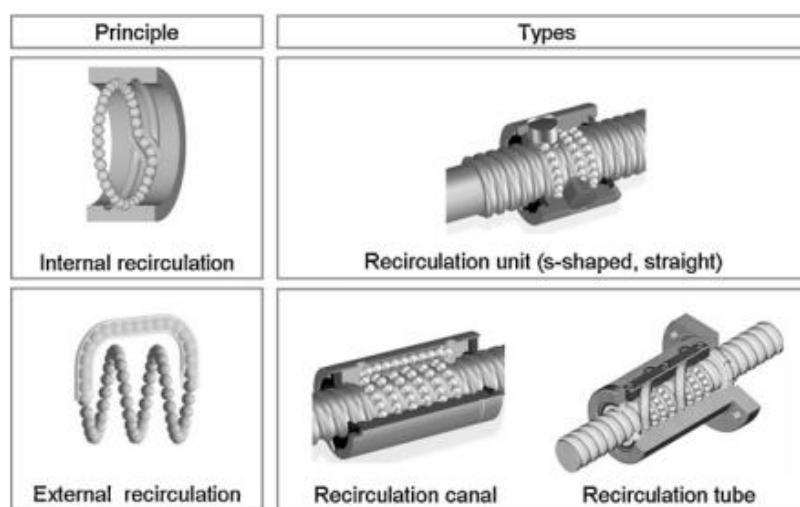
Figure 3.6 Ball recirculation systems

External ball recirculation is achieved by a recirculation tube or channel. An adapted design of the tube allows the balls to exit the bearing nut area and to enter it more tangentially, allowing amore even and smooth flow as well as higher speeds. Recirculation channel-based systems may be subdivided into various types, such as end cap recirculation or front-end recirculation. The internal recirculation system guides the balls via channels at the end of each thread. While this design has the advantage of requiring less space, the unfavorable ball entry and exit angles have adverse effects on even rolling and noise development. The design of feed motors and the mechanical components is initially carried out by considering only the rigid body dynamics and static stiffness of the system.

3.2.1   The HIWIN single-axis robot

The HIWIN KK single-axis robot is driven by a ball screw while a guideway slides on an optimized U-rail to achieve higher accuracy and greater stiffness.

Features of The HIWIN KK single-axis robot:

- An integrated system
- Easy installation and maintenance
- Compact and lightweight
- High accuracy
- Complete line of accessories
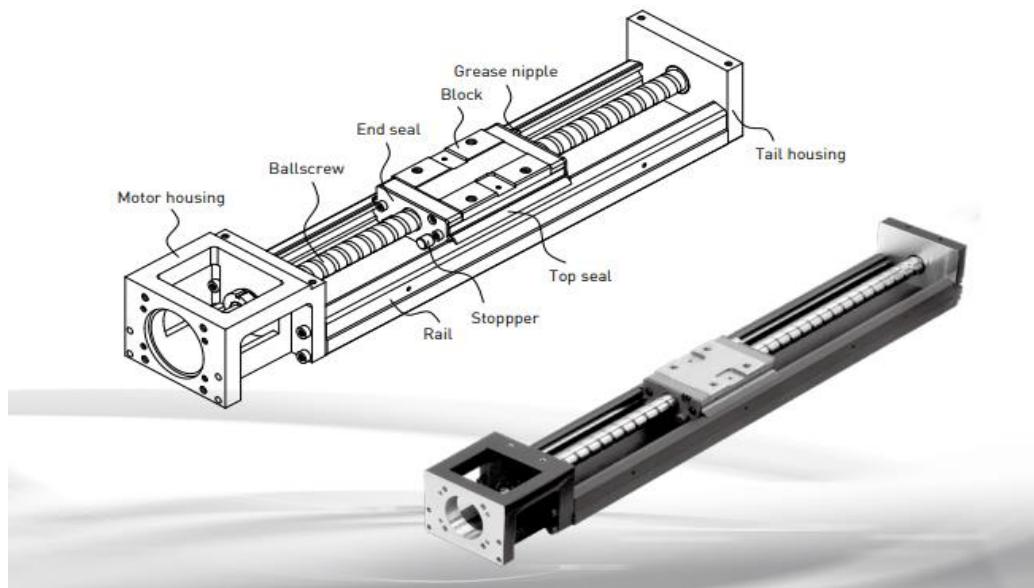
Figure 3.7 The HIWIN KK single-axis robot

The KK single axis robot integrating ball screw and guide rail creates a modular product. The modularized design can help customers save time, cost and system inspection. The KK single-axis robot enables installation efficiency and a space-saving design.
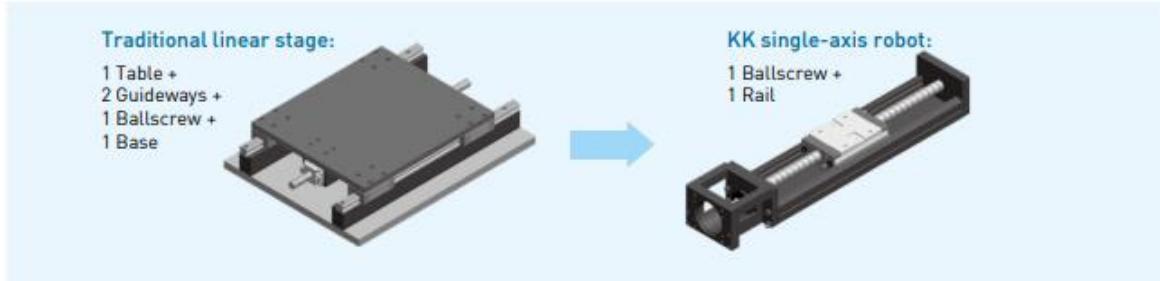
Figure 3.8 Traditional linear stage and KK single-axis robot

Using finite element analysis on the U-shaped cross section, the volume and rigidity can be balanced, resulting in a high rigidity rail, compact design, and light weight design all at the same time.

Accessories of KK single-axis robot is also supported for specific demands, such as an aluminum cover, bellows, motor adaptor flange and limit switches.

- Aluminum cover and bellow: contamination protection.
- Motor adaptor flange: Connection for different types of motors
- Limit switches: Starting point, positioning and other safety matters

The KK single axis robot integrating ball screw and guide rail creates a modular product. The modularized design can help customers save time, cost and system inspection. The KK single-axis robot enables installation efficiency and a space-saving design.

We chose the HIWIN KK single-axis robot because of the benefits we mentioned above.
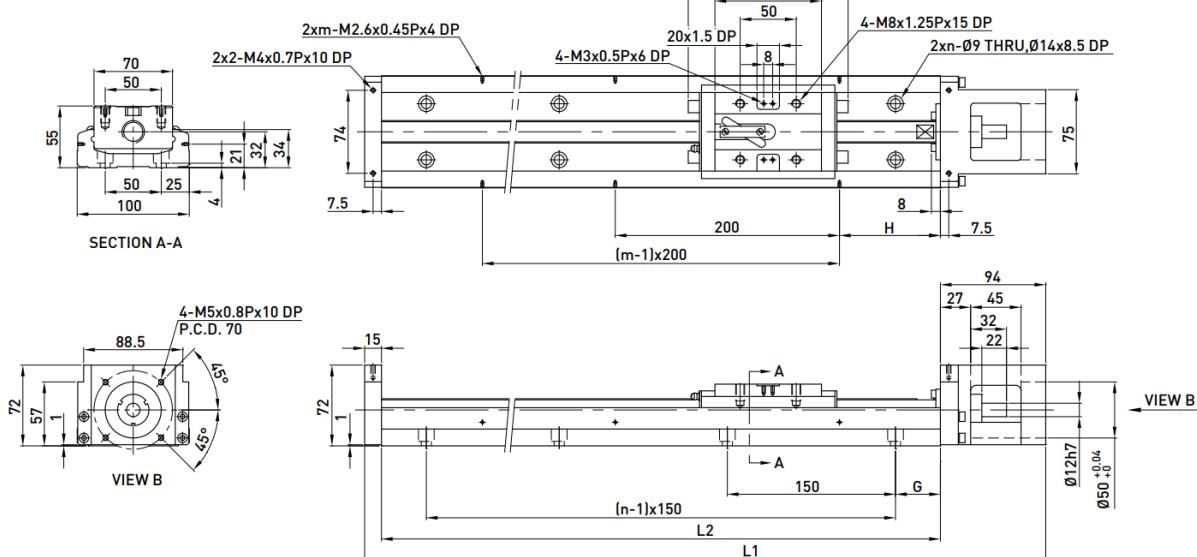
Figure 3.9 Single axis robot

Model No.		Ballscrew				Guideway																
		Nominal Diameter (mm)	Lead (mm)	Basic Dynamic Load (N)	Basic Static Load (N)	Basic Dynamic Load Rating (N)		Basic Static Load Rating (N)		Static Rated Moment												
										Allowable Static Moment M_P (N-m) (pitching)				Allowable Static Moment M_Y (N-m) (yawing)				Allowable Static Moment M_R (N-m) (rolling)			
						Block A	Block S	Block A	Block S	Block A1	Block A2	Block S1	Block S2	Block A1	Block A2	Block S1	Block S2	Block A1	Block A2	Block S1	Block S2
KK10020	Precision	20	20	7046	12544	39200	-	63406	-	960	4763	-	-	960	4763	-	-	2205	4410	-	-
	Normal			4782	9163																

Model	Ballscrew Lead (mm)	Rail Length L2 (mm)	Speed (mm/sec)	
			Precision	Normal
KK100	20	980	1120	800
		1080	980	800
		1180	750	750
		1280	630	630
		1380	530	530

KK100

| Rail Length L2 (mm)	Total Length L1 (mm)	Maximum Stroke (mm)		G (mm)	H (mm)	n	m	Mass (kg)	
		A1 Block	A2 Block					A1 Block	A2 Block
980	1089	828	700	40	90	7	5	18.6	20.3
1080	1189	928	800	15	40	8	6	20.3	22.0
1180	1289	1028	900	65	90	8	6	22.0	23.7
1280	1389	1128	1000	40	40	9	7	23.6	25.3
1380	1489	1228	1100	15	90	10	7	25.3	27.0

Figure 3.10 KK 100 Single-axis robot features

KK100 single axis precision robot lead value is 20mm seen on (Figure 3.10). Chosen rail length is 980mm, so maximum speed is 1120 (mm/s).

4 MODELLING of FEED DRIVE CONTROL SYSTEM

This section describes the components of feed drive model. With mathematical modelling in Simulink.

The traditional approach for modeling the dynamics of feed drive systems is to develop lumped rigid models based on equivalent inertia (J) and viscous damping coefficient (Be) reflected on motor shaft. In this model the drive train is considered as a rigid combination of links, joints, and couplings which transfer the motion from motor shaft to table position, and all structural flexibilities are neglected. But the model we made in this project has a single degree of freedom (SDOF) oscillator. Thanks to this addition into the rigid model, we will do vibration reduction.
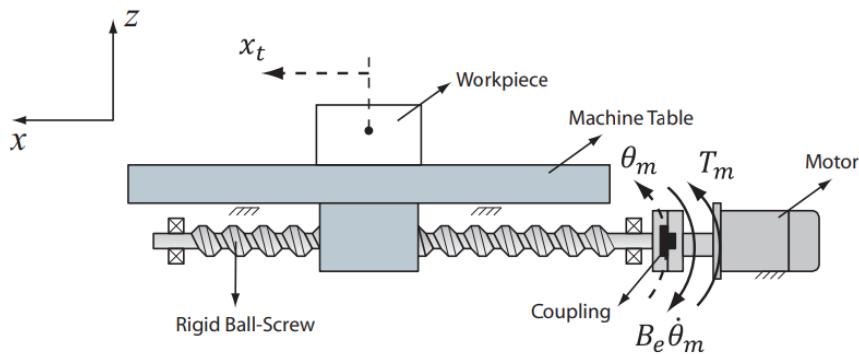
Figure 4.1 Rigid ball screw mounted on a rigid bed

In rigid body modeling, all the flexibilities in the column, bed, ball-screw, coupling, and bearings are neglected and they are assumed as rigid links and joints. Therefore, the angular displacement at motor shaft (θ_m) and nut (θ_n) are the same. The angular displacement is converted to linear motion at machine table (x_t) through the screw-nut interface. For each rotation of the screw, machine table moves for a screw pitch length (h_p). The machine table displacement (x_t) is a function of angular motion at ball-screw and nut interface (θ_m) as Equation 4.1.

$$x_t = r_g \theta_m; r_g = \frac{h_p}{2\pi} \qquad\qquad 4.1$$

Where the r_g is the nut transformation ratio.

Feed drive control system model has different pieces, and we will be starting to explain details of this model by AC servo motor model that placed in the inner loop subsystem (Figure 4.13)
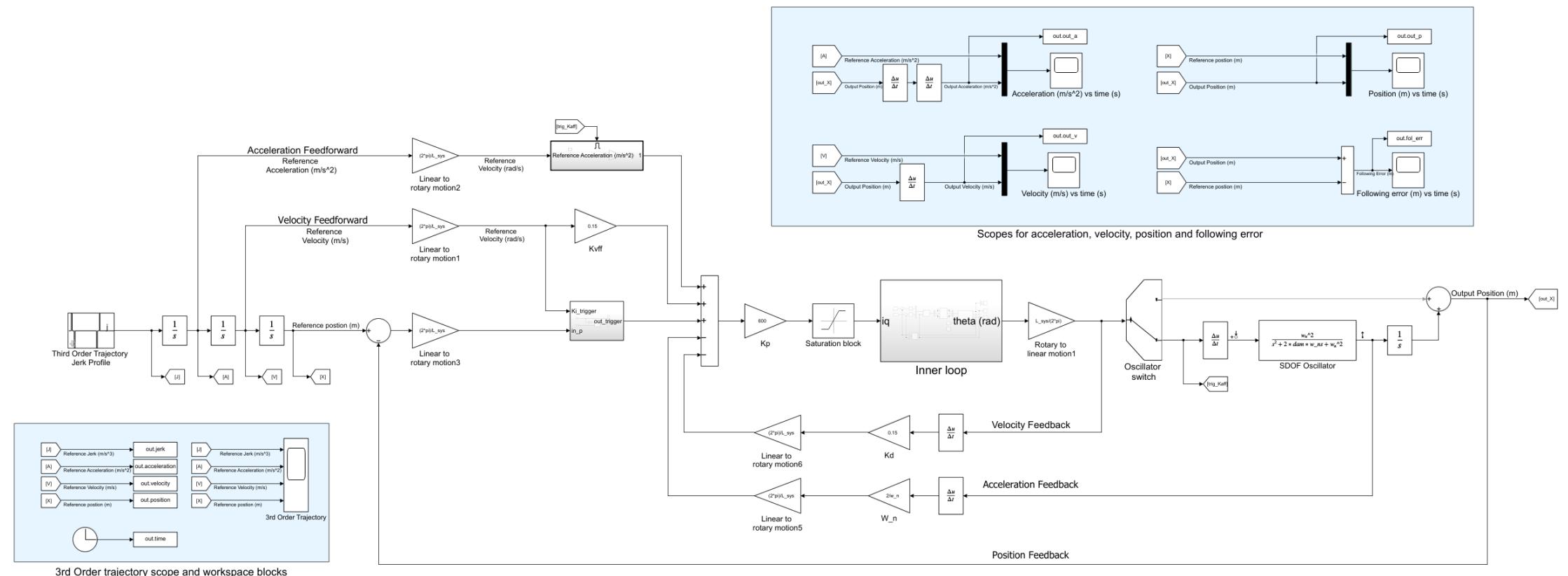
Figure 4.2 CNC Machine Tool Feed Drive Control System Model

4.1 Modelling of an AC Servo Motor

A servo motor is an electromechanical device that produces torque and velocity based on the supplied current and voltage. A servo motor works as part of a closed loop system providing torque and velocity as commanded from a servo controller utilizing a feedback device to close the loop. The feedback device supplies information such as current, velocity, or position to the servo controller, which adjusts the motor action depending on the commanded parameters.
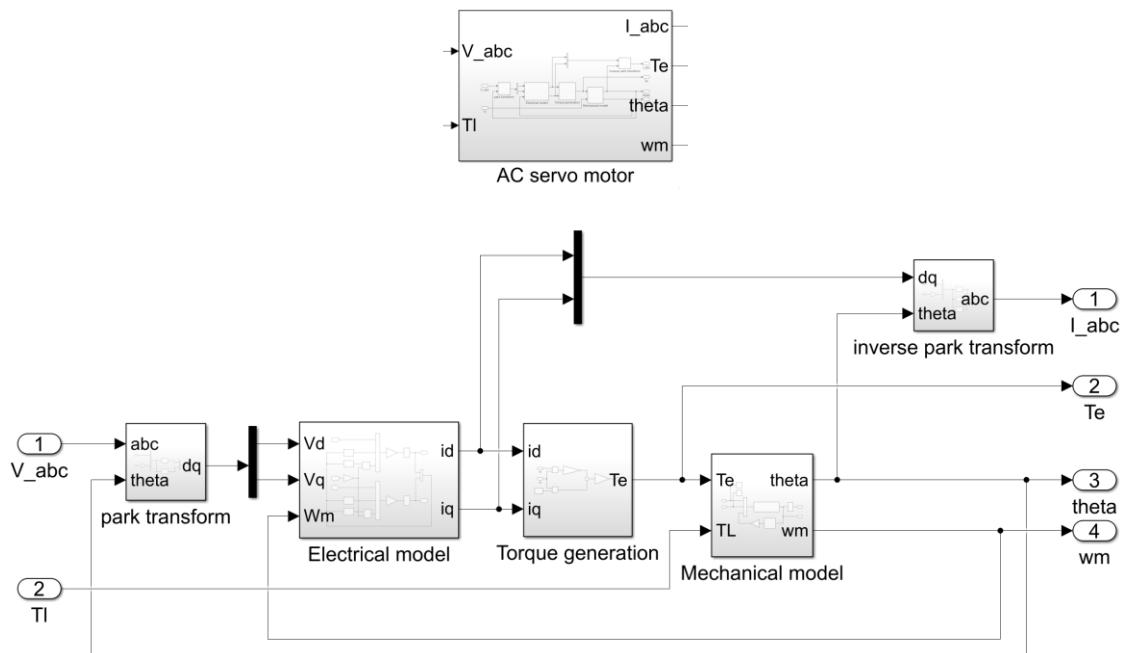
Figure 4.3 Simulink model of an AC servo motor

Motor converts electrical power into mechanical motion. Since the motor is an electromechanical device, its functionality can be described by the interaction of electrical and mechanical components as shown in (Figure 4.4).
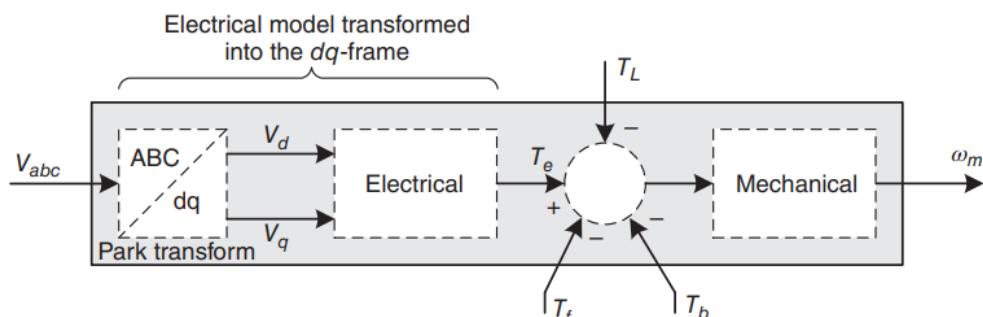
Figure 4.4 Motor model includes electrical and mechanical components

We can begin to model the electrical side of the motor by considering the per-phase electrical circuit. The motor's magnetic structure with windings generates a back emf voltage "e" at the terminals of the phase. This voltage opposes the input voltage and is given by Faraday's law.

$$V_0(t) = iR + \frac{d\lambda}{dt}$$

4.2

4.1.1 Mechanical Model

The mechanical side can be modeled, as shown in (Figure 4.5). Here, T_e is the torque generated by the electrical side of the motor. T_f models static friction, such as Coulomb friction, while Tb can account for any energy dissipation due to friction in the bearings of the motor.

The torque generated by the electrical side works against the torques due to the static friction, viscous friction, and the load. The difference of these torques is used to accelerate the rotor to a mechanical rotational speed of ω_m. The viscous friction torque is often modeled as $T_f = b\omega_m$ where b is the viscous friction coefficient with Nm s/rad units. Using Newton's second law, we can write Equation (4.5).

$$T_e - T_L - T_f = J\dot{\omega}_m + b\omega_m$$

4.3

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

4.4

$$\omega_m(s) = \frac{1}{J_s + b}(T_e - T_L - T_f)$$

4.5

The mechanical model in Equation (4.5) has implemented using Simulink as shown in (Figure 4.6). This is a top-level block. When this block is opened, its details are revealed.
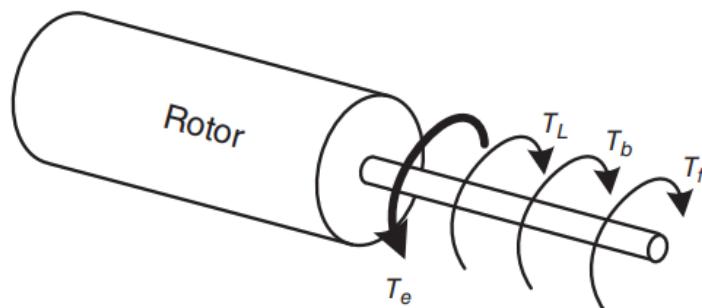
Figure 4.5 Motor's mechanical model

31
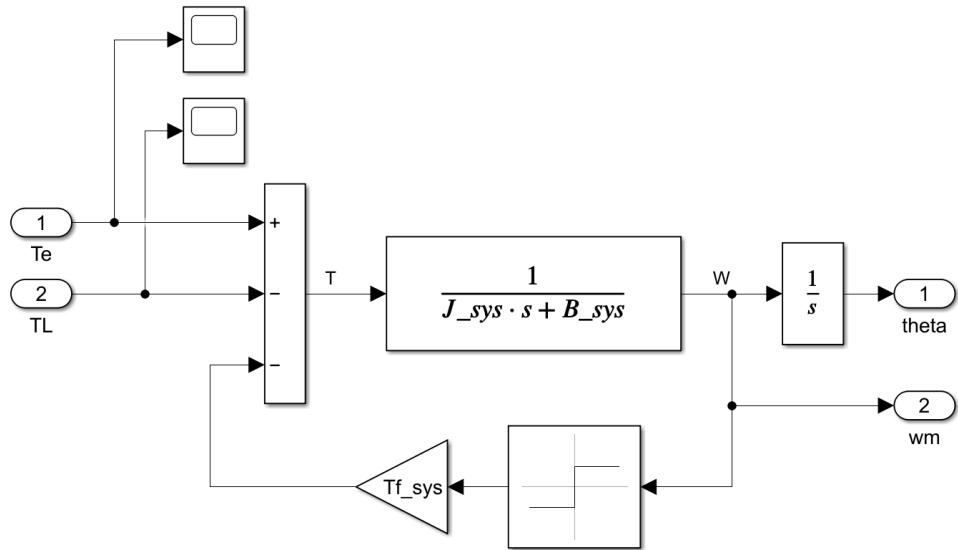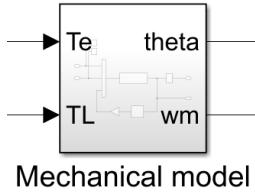
Figure 4.6 Mechanical model of a motor

The block takes inputs T_e and T_L as the torque generated by the electrical side and the load torque, respectively. It computes the angular speed wm and position theta. The signum function, Sign, was used to make sure that the static friction torque always opposes the input torque T_e. The J, B, and T_f values need to be specified for the simulated motor by entering them into MATLAB Appendix A1.

An AC servo motor receives phase voltages V_a, V_b, and V_c, which are then converted into torque by the stator and rotor field interactions Equation (4.6).

$$
\begin{aligned}
V_a &= i_a R + \frac{d\lambda_a}{dt} \\
V_b &= i_b R + \frac{d\lambda_b}{dt} \\
V_c &= i_c R + \frac{d\lambda_c}{dt}
\end{aligned}
\qquad 4.6
$$

$$\lambda = Li \qquad 4.7$$

where L is the inductance and i is the current.

4.1.2 Park Transform

Equation should be transformed into a new coordinate system known as "dq-frame" where the inductances become constant (Figure 4.9). This transformation converts the three-phase system into an equivalent

two-phase system. The dq-frame is attached to the rotor and rotates with it. The motor dynamic model turns into constant coefficient differential equations in the dq-frame.

Park transform is used to transform the three-phase stator quantities (such as phase voltages, currents, flux linkages) into the rotating dq-frame. This transformation can be described as a matrix as seen in the Equation (4.8). This matrix has been used to create park transform model seen in the (Figure 4.7)

$$
\begin{Bmatrix} S_d \\ S_q \\ S_0 \end{Bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ -\sin\theta & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3}) \\ 1 & 1 & 1 \end{bmatrix} \begin{Bmatrix} S_a \\ S_b \\ S_c \end{Bmatrix} \qquad 4.8
$$

Where S_a, S_b, and S_c are the stator quantities, and S_d and S_q are the corresponding dq-frame quantities. Angle θ is the electrical angle of the rotor. The S_0 term is included to yield a square matrix. In balanced three-phase systems, ones in the matrix are ignored when writing the functions. Fcn codes are on the Appendix A.
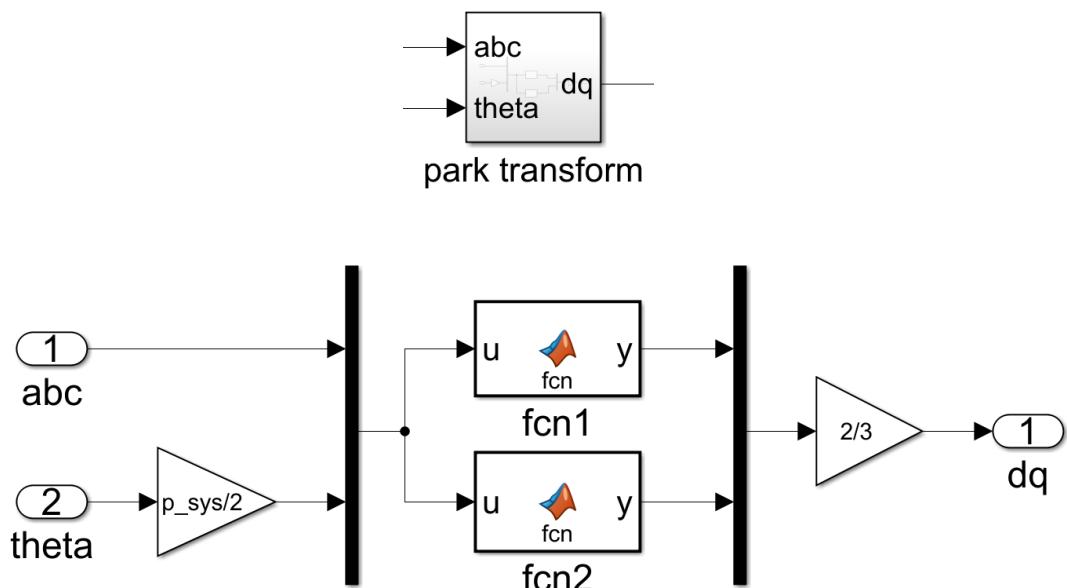
Figure 4.7 Park transform model

4.1.3 Inverse Park Transform

Inverse Park transforms given in Equation (4.9) can be used to transform the dq-frame quantities back to three-phase quantities. Matrix form Equation (4.9) has been used to create inverse park transform model seen in the (Figure 4.8). Fcn codes are on the Appendix B.

$$\begin{Bmatrix} S_a \\ S_b \\ S_c \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 1 \\ \cos(\theta - \dfrac{2\pi}{3}) & -\sin(\theta - \dfrac{2\pi}{3}) & 1 \\ \cos(\theta + \dfrac{2\pi}{3}) & -\sin(\theta + \dfrac{2\pi}{3}) & 1 \end{bmatrix} \begin{Bmatrix} S_d \\ S_q \\ S_0 \end{Bmatrix} \qquad 4.9$$
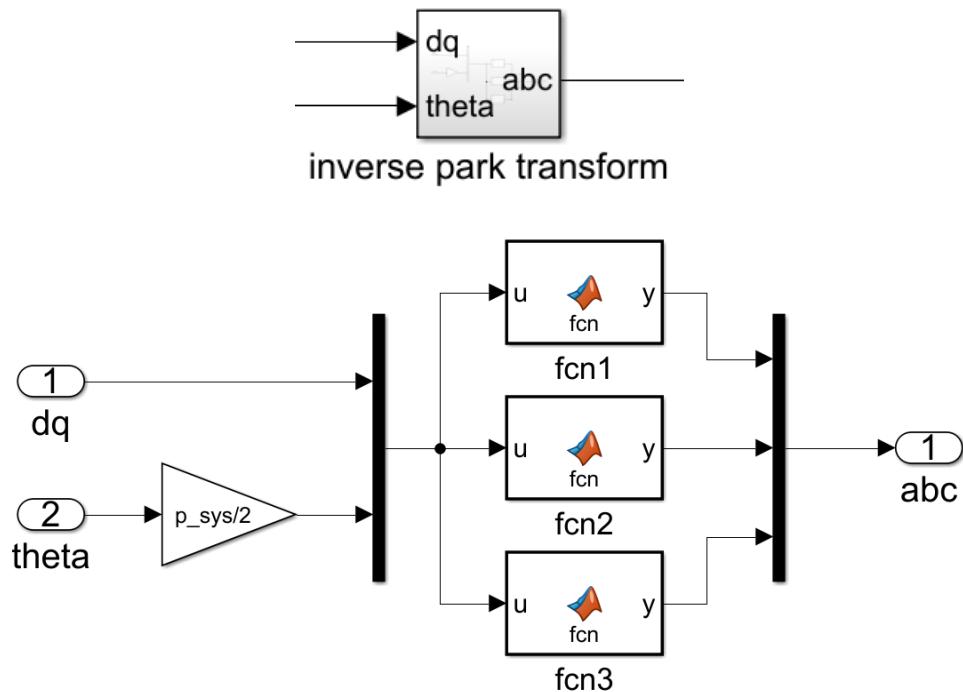
Figure 4.8 Inverse park transform model

By applying the Park transform to the stator currents $[i_a\ i_b\ i_c]^T$ and the flux linkages $[\lambda_a\ \lambda_b\ \lambda_c]^T$, Equation (4.6) can be transformed into the dq-frame, where ω_e is the rotor electrical speed.

$$V_d = i_d R + \frac{d\lambda_d}{dt} - \omega_e \lambda_q \qquad 4.10$$

$$V_q = i_q R + \frac{d\lambda_q}{dt} + \omega_e \lambda_d \qquad 4.11$$

4.1.4 Torque Generation Model

The Park transform converts the stationary three-phase ABC coil system into two equivalent coils, L_d and L_q (Figure 4.4). These hypothetical coils are attached to the dq-frame, which rotates with the rotor (Figure 4.9). Therefore, the position dependent inductance L becomes constant L_d and L_q. These are called direct-axis (L_d) and quadrature-axis (L_q) inductances. Similarly, currents i_d and i_q are called the d-axis and q-axis currents, respectively.
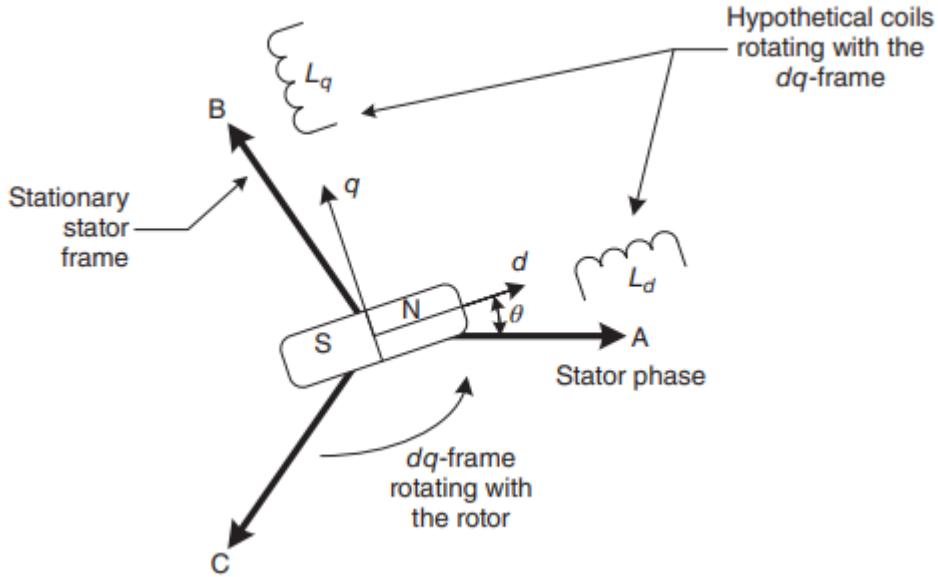
Figure 4.9 dq-frame and the hypothetical rotating direct-axis (L_d) and quadrature-axis (L_q) inductances

Along the d-axis, we have flux linkage due to the id current and the rotor permanent magnet (PM) as

$$\lambda_d = L_d i_d + \lambda_{PM}$$

4.12

Along the q-axis, we have flux linkage due to the i_q current as

$$\lambda_q = L_q i_q$$

4.13

The torque generated by the motor is:

$$T_e = \frac{3}{2}\left(\frac{p}{2}\right)\left[\lambda_d i_q - \lambda_q i_d\right]$$

4.14

By substituting Equations (4.16) and (4.17) into (4.22) we obtain

$$T_e = \frac{3}{2}\left(\frac{p}{2}\right)\left[(L_d i_d + \lambda_{PM})i_q - (L_q i_q)i_d\right] \tag{4.15}$$

Rearranging the equation gives the formula that we need for the torque generation model

$$T_e = \frac{3}{2}\left(\frac{p}{2}\right)\left[(L_d - L_q)i_d i_q + \lambda_{PM}i_q\right] \tag{4.16}$$
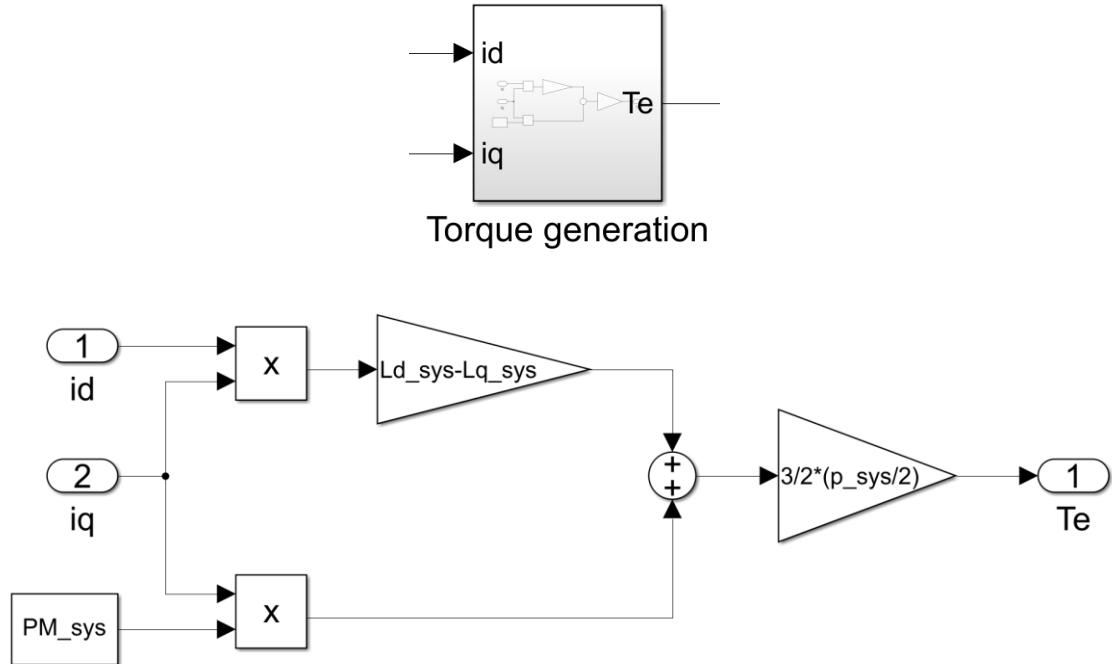
Torque generation

Figure 4.10 Torque generation model

4.1.5   Electrical Model

Substituting Equations (4.12) and (4.13) into (4.10) gives

$$V_d = i_d R + \frac{d}{dt}(L_d i_d + \lambda_{PM}) - \omega_e L_q i_q \& = i_d R + L_d \frac{di_d}{dt} - \omega_e L_q i_q \tag{4.17}$$

Simplified equation

$$\frac{di_d}{dt} = \frac{1}{L_d}\left[V_d - i_d R + \omega_e L_q i_q\right] \tag{4.18}$$

Taking Laplace transform of both sides with zero initial conditions gives

$$i_d(s) = \frac{1}{L_d s}\left[V_d - i_d R + \omega_e L_q i_q\right]$$

4.19
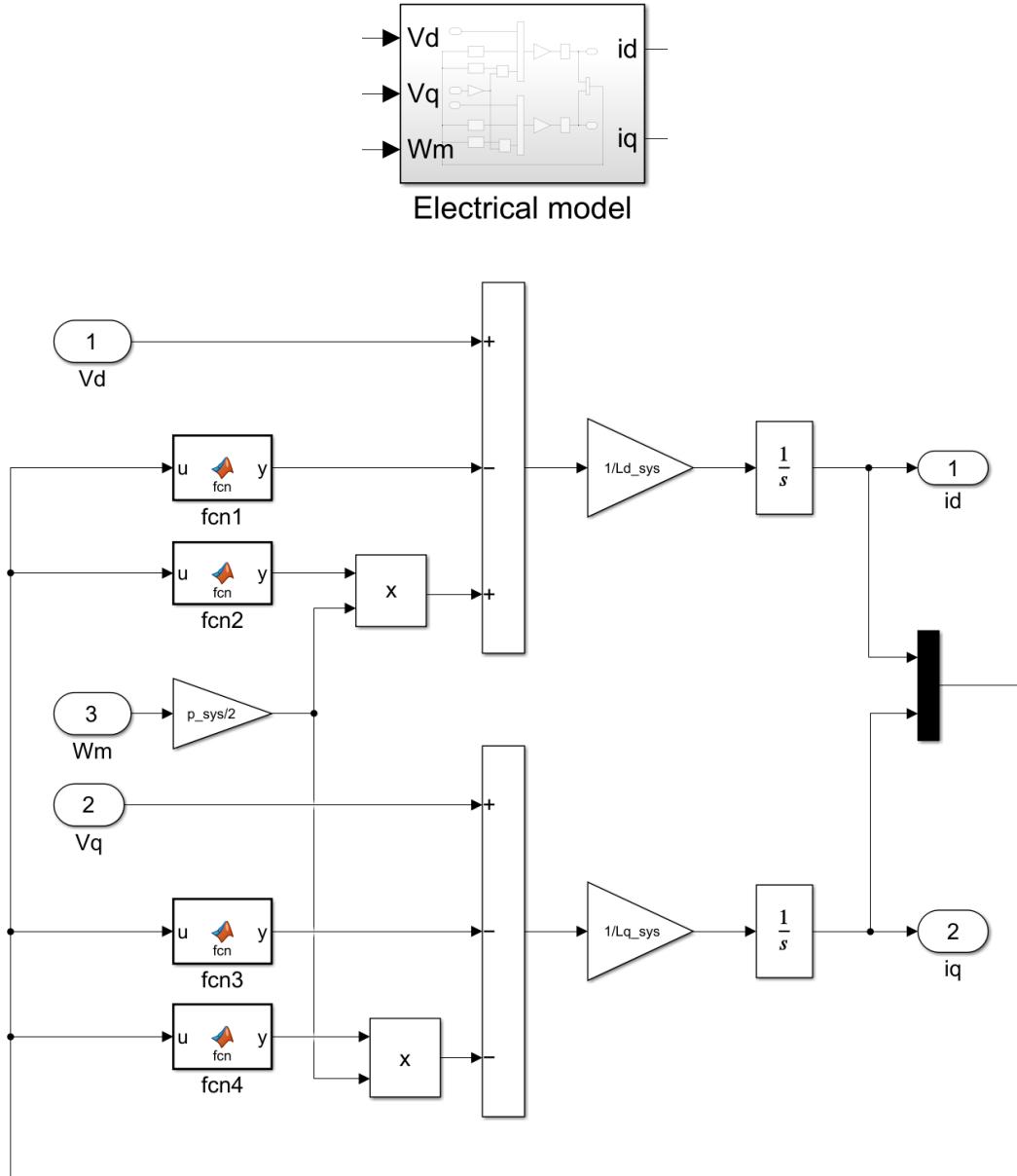
Electrical model

Figure 4.11 Electrical model

Similarly, substituting Equations (4.12) and (4.13) into (4.11) yields

$$V_q = i_q R + \frac{d}{dt}\left(L_q i_q\right) + \omega_e L_d i_d + \omega_e \lambda_{PM}$$

4.20

Simplified equation

$$\frac{di_q}{dt} = \frac{1}{L_q}\left[V_q - i_q R - \omega_e(L_d i_d + \lambda_{PM})\right]$$

4.21

Again, taking Laplace transform gives

$$i_q(s) = \frac{1}{L_q s}\left[V_q - i_q R - \omega_e(L_d i_d + \lambda_{PM})\right]$$

4.22

Equations (4.19) and (4.22) are the electrical model's (Figure 4.11) mathematical representation. MATLAB codes for the fcn blocks can be seen on the Appendix C.

4.2 Modelling of an AC Drive

AC drive amplifies small command signals generated by the controller to high-power voltage and current levels necessary to operate a motor. Therefore, the drive is also called an amplifier. In motion control systems, each axis operates under closed-loop control. Typically, there are three loops to close in each axis, namely, current, velocity, and position loops. The motor's velocity and position are measured and fed back to the controller, while the motor's currents are measured and fed back to the drive. In other words, the drive closes the current loop. Yet, in recent trends, the line between the functions of a controller and the drive continues to blur. Many functions of the controller, including closing the velocity and position loops, are now done by the drive.
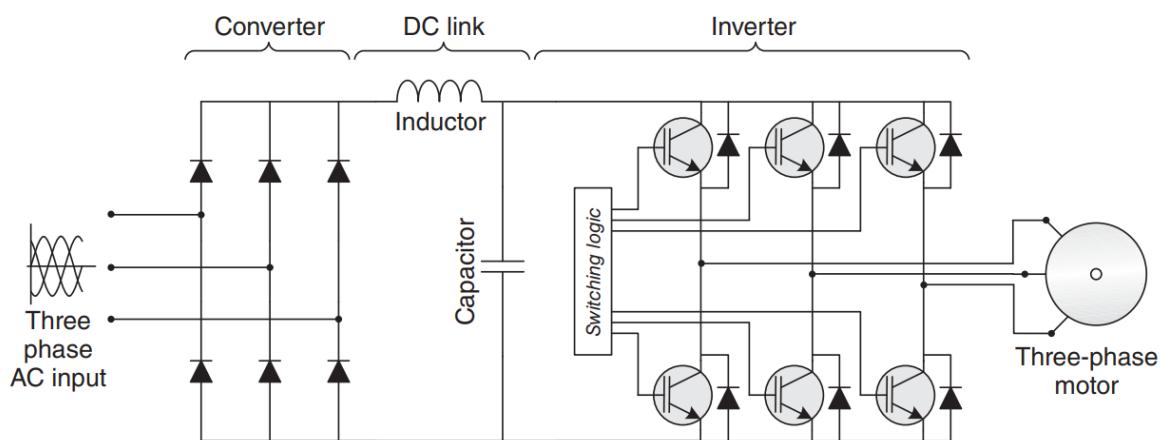
Figure 4.12 AC drive physical representation with pulse width modulation (PWM) inverter

The synchronous speed of the three-phase stator will be fixed if the stator is directly connected

to three-phase AC power lines. To vary the speed, we need to adjust the voltage level and frequency of the three-phase sinusoidal waveforms supplied to the motor. The power electronics equipment designed for this purpose is called an AC drive.

(Figure 4.12) shows the basic circuit blocks in a drive. AC drives convert AC power into DC and invert DC into variable voltage and frequency three-phase AC power. There are two types of drives: voltage source inverter (VSI)-based drives and current source inverter (CSI)-based drives. The PWM switching technique is commonly used to achieve the transition from DC to AC in the inverter section.

The PWM drives are energy efficient and provide high performance. The PWM is implemented in the switching logic circuit block of the drive. The VSI produces an adjustable three-phase PWM voltage waveform for the load (motor). The CSI outputs a PWM current waveform.
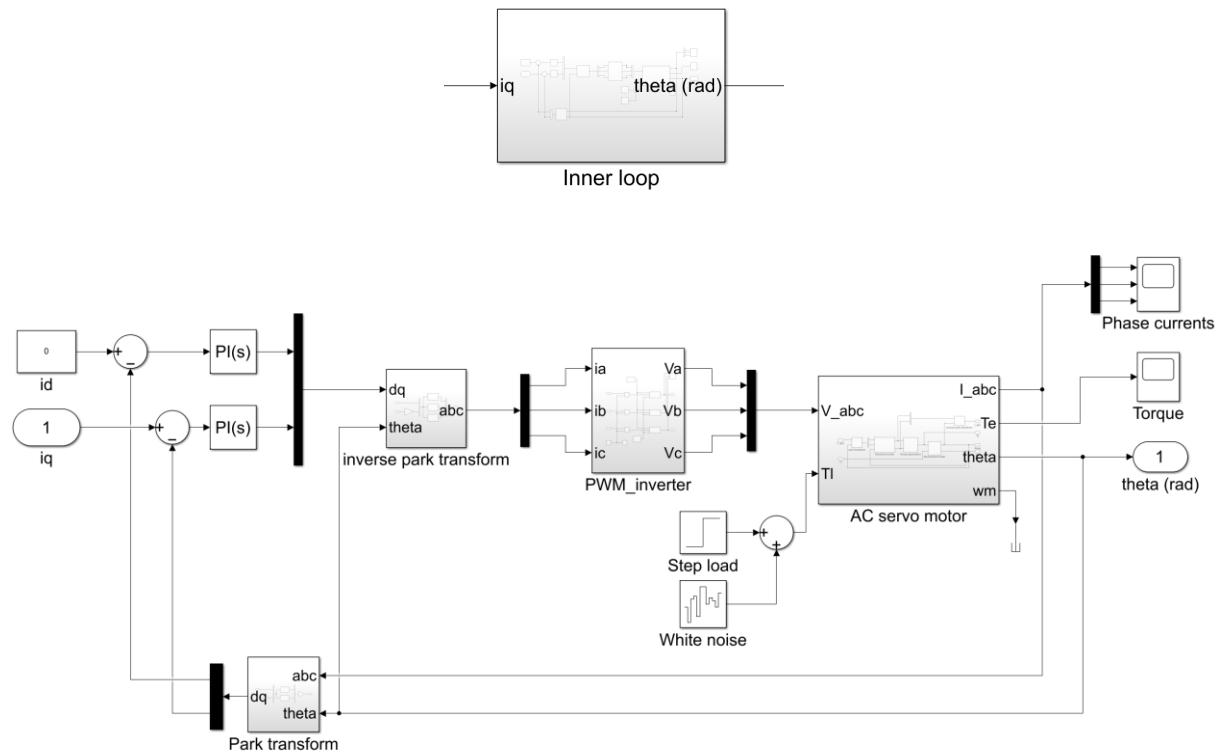
Figure 4.13 AC drive Simulink model

White noise and step load have been summed up to get most realistic load torque. White noise imitates stuttering problem that could be occur while cutting a work piece.

This chapter begins by presenting the building blocks of the drive electronics, namely, converter, DC link, and PWM inverter. Then AC drive Simulink model in (Figure 4.13) explained detailly. The popular pulse-width modulation (PWM) control technique is explained.

4.2.1 Pulse-width Modulation (PWM)

AC drives can vary the speed of the motor by adjusting the voltage and frequency of the three-phase sinusoidal waveforms supplied to the motor. When a fixed DC bus voltage is used with an uncontrolled rectifier bridge, we need a way to vary the amplitude of the voltage waveforms generated by the inverter.

Pulse Width Modulation or PWM technology is used in Inverters to give a steady output voltage of 230 or 110 V AC irrespective of the load. The Inverters based on the PWM technology are more superior to the conventional inverters. The use of MOSFETs in the output stage and the PWM technology makes these inverters ideal for all types of loads.

There are many PWM techniques including sinusoidal, space-vector, and hysteresis band current control. A variable effective voltage output can be obtained by varying how long a square wave stays on in a given period.
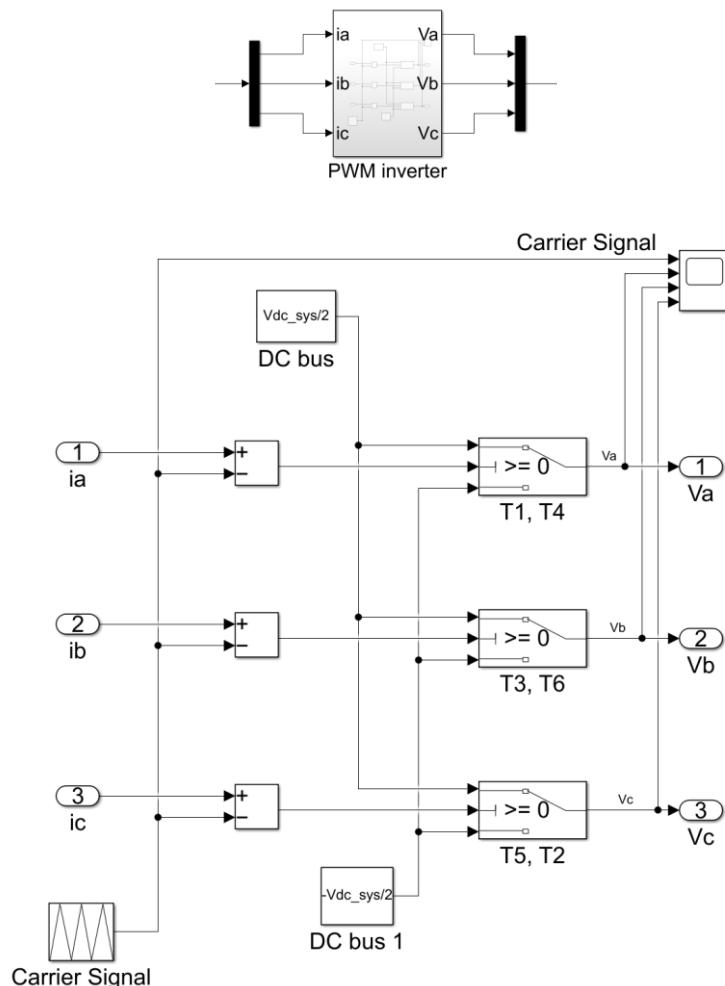
Figure 4.14 Simulation model for the PWM inverter.
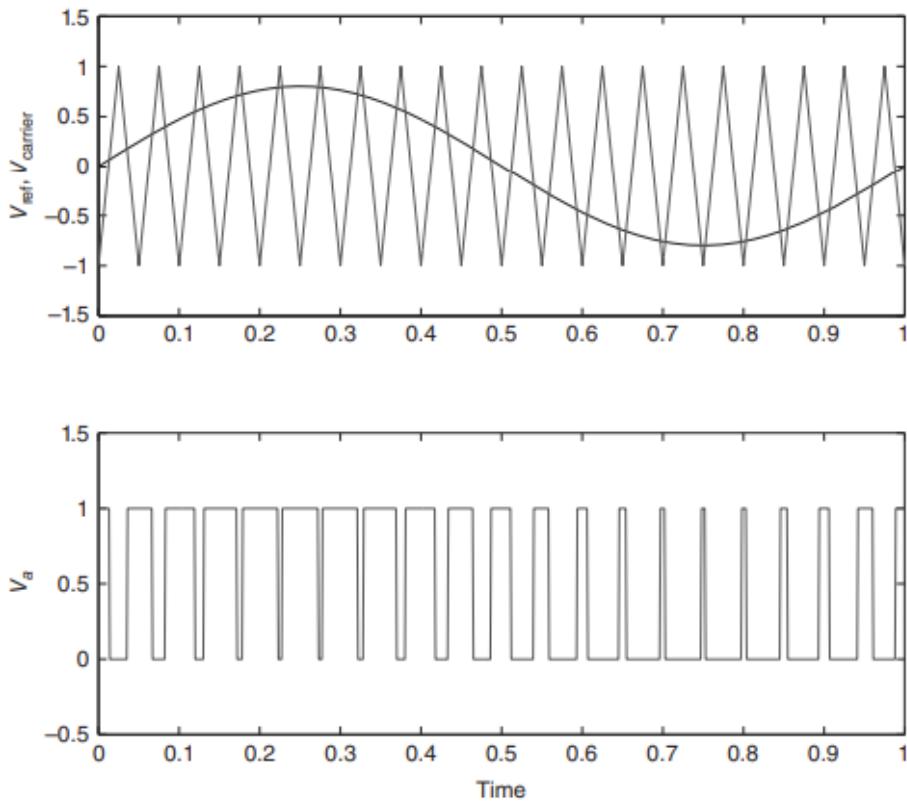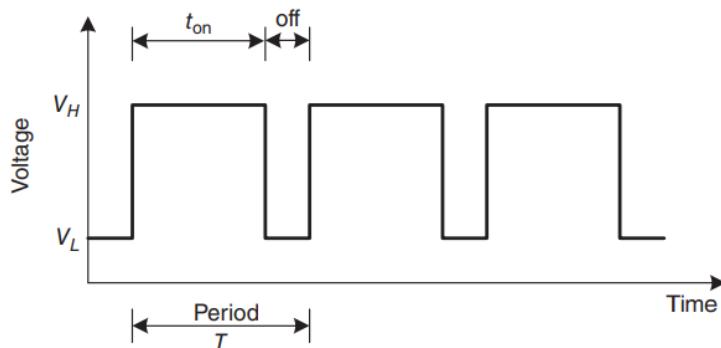
Figure 4.15 Three-phase reference signals shifted 120° and the corresponding PWM signals

To generate a PWM signal, a reference signal, v_{ref} , and a triangular carrier signal, $v_{carrier}$, are used as control signals as shown in (Figure 4.15). The voltage and frequency of the carrier signal are fixed, whereas the reference signal can have adjustable voltage and frequency. At any given instant, the magnitudes of the signals are compared to each other. If the magnitude of the reference signal is greater than the carrier, then the PWM signal is set to 1 (switch closed). Otherwise, it is set to zero (switch open).

PWM signals tend to be in the range of 2–30 kHz. 15 kHz carrier signal have been used for this project. The duty cycle of the PWM signal defines the average voltage applied to the stator,

41

which is proportional to the motor speed. If the duty cycle is increased, the motor speed will increase. In addition to the efficiency advantage, the PWM method also allows easy adjustment of the DC bus voltage. If the DC bus voltage of the controller is higher than the rated voltage of the motor, the magnitude of the PWM signal can easily be adjusted in software to match the rated voltage of the motor. As a result, the controller can be interfaced to motors with different rated voltages.

4.3 Third Order Trajectory Planning

From figure (Figure 4.2) entire control system can be seen. On the left part of the system there is a third order trajectory planning model. This model needs four different inputs to be able to work. These inputs are displacement, velocity, acceleration, jerk. MATLAB codes (Appendix E) take those inputs and makes the third order trajectory's jerk profile. Then this jerk profile changes into acceleration, velocity and position profiles with an help of 3 integrator blocks. Mathematical representation of this process (4.23).

$$j(t) = \frac{da(t)}{dt} = \frac{d^2v(t)}{dt^2} = \frac{d^3p(t)}{dt^3} \qquad 4.23$$
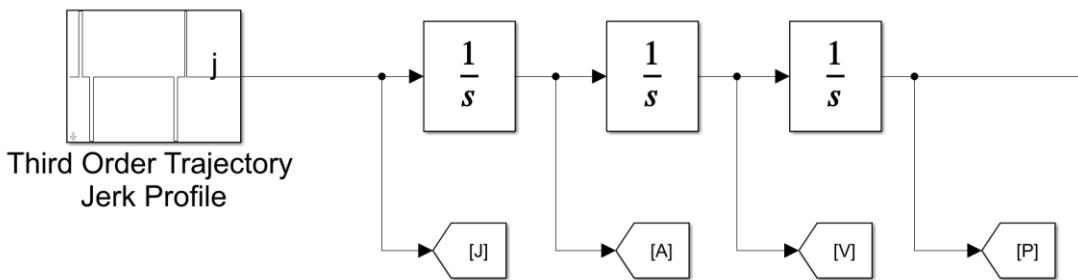
Figure 4.16 Thrid order trajectory planning model

The trajectory planning algorithm will be based on the MATLAB codes (Appendix E) [4] construction of an appropriate jerk profile. We will assume that a trajectory must be planned over a distance x, that bounds are given on velocity (v), acceleration (a) and jerk (j), and that the trajectory must be time optimal within these bounds. The bound on jerk can for instance be related to 'rise time behavior as commonly found in electromechanical actuation systems with a non-ideal power amplifier for generating the actuation force. This results in a bound on the maximal current change per second, which translates to a maximal actuation force change per second and a maximal acceleration change per second: hence a bound on jerk.

A symmetrical trajectory is completely determined by three time intervals: the constant jerk interval $t_{\bar{j}}$, the constant acceleration interval $t_{\bar{a}}$, and the constant velocity interval $t_{\bar{v}}$. The resulting profiles are given in (Figure 4.17). Including the starting time of the trajectory at to, there are eight time instances at which the jerk changes. The following relations are clear:

$$t_{\bar{j}} = t_1 - t_0 = t_3 - t_2 = t_5 - t_4 = t_7 - t_6 \qquad\qquad 4.24$$
$$t_{\bar{a}} = t_2 - t_1 = t_6 - t_5$$
$$t_{\bar{v}} = t_4 - t_3$$

we will discard the bounds on acceleration and velocity, and only consider the bound on jerk. This implies $t_{\bar{v}} = 0$ and $t_{\bar{a}} = 0$ and it is dear that this will provide us with a lower bound for the trajectory execution time $t_{\bar{x}} = 4 * t_{\bar{j}}$

To obtain $t_{\bar{j}}$ (step 2), we need the relation between $t_{\bar{j}}$ and $\bar{x}$ with given $\bar{J}$. For this we make use of the constant value of jerk of $+\bar{J}$ or $-\bar{J}$ during each interval. If we set the time instance of jerk change to 0, it is easily verified that for any time t during the constant jerk interval the third order profiles for acceleration, velocity and position can be expressed as follows:

$$a(t) = j_0 t + a_0 \qquad\qquad 4.25$$
$$v(t) = \frac{1}{2} j_0 t^2 + a_0 t + v_0$$
$$x(t) = \frac{1}{6} j_0 t^3 + \frac{1}{2} a_0 t^2 + v_0 t + x_0$$

Because we assume that the bounds on acceleration and velocity are not violated, we have $t_{\bar{a}} = 0$ and $t_{\bar{v}} = 0$, and consequently from figure 6: $t_2 = t_1 = t_{\bar{j}}$ and $t_4 = t_3$. Hence:

$$a(t_2) = a(t_1) = \bar{J} t_{\bar{j}} \qquad\qquad 4.26$$
$$v(t_2) = v(t_1) = \frac{1}{2} \bar{J} t_{\bar{j}}^2$$
$$x(t_2) = x(t_1) = \frac{1}{6} \bar{J} t_{\bar{j}}^3$$

and for the next period in which $J = -\bar{J}$:

$$a(t_4) = a(t_3) = -\bar{J} t_{\bar{j}} + a(t_2) \qquad\qquad 4.27$$
$$v(t_4) = v(t_3) = -\frac{1}{2} \bar{J} t_{\bar{j}}^2 + a(t_2) t_{\bar{j}} + v(t_2)$$
$$x(t_4) = x(t_3) = -\frac{1}{6} \bar{J} t_{\bar{j}}^3 + a(t_2) t_{\bar{j}}^2 + v(t_2) t_{\bar{j}} + x(t_2)$$
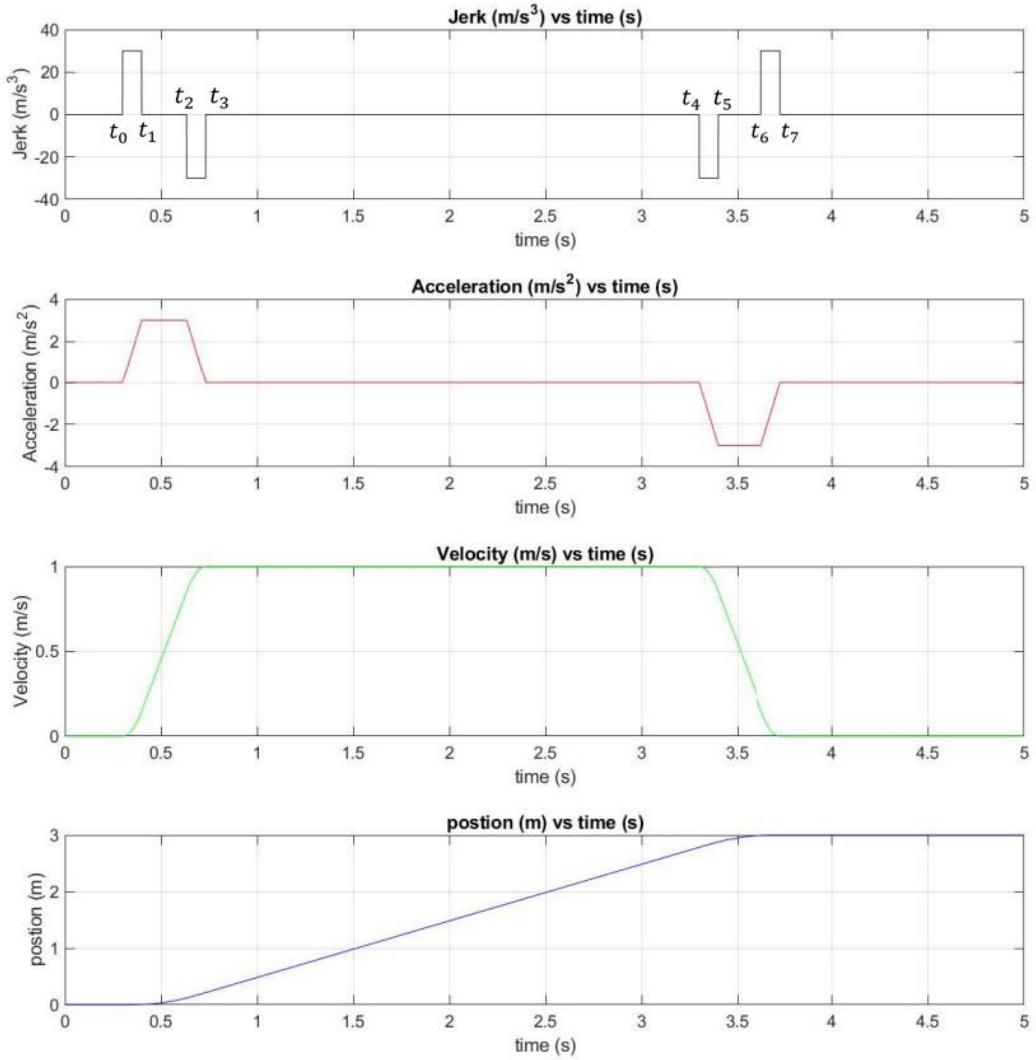
Figure 4.17 Third order trajectory planning

Displacement	$3\ (m)$
Bound on velocity	$1\ (m/s)$
Bound on acceleration	$3\ (m/s^2)$
Bound on jerk	$30\ (m/s^3)$

Table 4-1 3rd order trajectory values

Substitution gives:

$$a(t_4) = a(t_3) = 0 \qquad\qquad 4.28$$
$$v(t_4) = v(t_3) = \bar{J}t_j^2$$
$$x(t_4) = x(t_3) = \bar{J}t_j^3$$

and due to symmetry of the profile: $x(t_7) = 2x(t_3) = 2t_j^3$. From this we can calculate the minimal required time to execute the trajectory for a distance $\bar{x}$ given the bound on jerk $\bar{J}$ as:

44

$$t_{\bar{x}} = 4t_{\bar{j}} = 4 \cdot \sqrt[3]{\frac{\bar{x}}{2\bar{j}}}$$

As mentioned before, the result of Equation (4.29) does not take into account whether or not the bounds on acceleration and/or velocity are violated. However, if this would be the case, it is obvious that t, must increase, but also that $t_{\bar{j}}$ must decrease.

Step 3 of the algorithm follows from Equation (4.28); the maximum velocity occurs at t_3, such that:

$$\hat{v} = \bar{j}t_{\bar{j}}^2$$

If $\hat{v} > \bar{v}$ the bound is violated and we must recalculate $t_{\bar{j}}$ as follows:

$$t_{\bar{j}} = \sqrt{\frac{\bar{v}}{\bar{j}}}$$

Note that the resulting $t_{\bar{j}}$ will always be smaller than the result from Equation (4.29), and that consequently also the acceleration from Equation (4.27) will become smaller. This establishes step 4 of the algorithm.

Step 5 follows from Equation (4.27); the maximum acceleration occurs at t_1, such that:

$$\hat{a} = \bar{j}t_{\bar{j}}$$

If $\hat{a} > \bar{a}$ the bound is violated and we must recalculate $t_{\bar{j}}$ as follows:

$$t_{\bar{j}} = \frac{\bar{a}}{\bar{j}}$$

Again $t_{\bar{j}}$ can only be smaller than previously calculated, such that we now have the guarantee that $t_{\bar{j}}$ is the maximal value for which none of the bounds is violated. At the same time, we have that a maximal value of $t_{\bar{j}}$ will lead to a minimal value of $t_{\bar{x}}$, and will therefore be a part of the time optimal solution. This establishes step 6 of the algorithm and because a is the highest derivative before J, also step 7.

The next thing to do is to calculate $t_{\bar{a}}$: also, this value should be maximal for time-optimal performance, while at the same time it must be such that no bounds are violated. To do the

necessary calculations, assume that $t_{\bar{a}} > 0$, such that we now have $t_2 > t_1$ from (Figure 4.17). Equation (4.27) can then be extended with the help of Equation (4.26) and the fact that $J = 0$ during the period from t_1 to t_2:

$$a(t_2) = a(t_1) + 0t_{\bar{a}} = \bar{J}t_{\bar{J}} \qquad 4.34$$

$$v(t_2) = v(t_1) + a(t_1)t_{\bar{a}} = \frac{1}{2}\bar{J}t_{\bar{J}}^2 + \bar{J}t_{\bar{J}}t_{\bar{a}}$$

$$x(t_2) = x(t_1) + v(t_1)t_{\bar{a}} + \frac{1}{2}a(t_1)t_{\bar{a}}^2$$

$$= \frac{1}{6}\bar{J}t_{\bar{J}}^3 + \frac{1}{2}\bar{J}t_{\bar{J}}^2 t_{\bar{a}} + \frac{1}{2}\bar{J}t_{\bar{J}}t_{\bar{a}}^2$$

Again, using Equation (4.26) we can then calculate:

$$a(t_3) = a(t_2) - \bar{J}t_{\bar{J}} = 0 \qquad 4.35$$

$$v(t_3) = v(t_2) + a(t_2)t_{\bar{J}} - \frac{1}{2}\bar{J}t_{\bar{J}}^2$$

$$= \frac{1}{2}\bar{J}t_{\bar{J}}^2 + \bar{J}t_{\bar{J}}t_{\bar{a}} + \bar{J}t_{\bar{J}}^2 - \frac{1}{2}\bar{J}t_{\bar{J}}^2$$

$$= \bar{J}t_{\bar{J}}t_{\bar{a}} + \bar{J}t_{\bar{J}}^2$$

$$x(t_3) = x(t_2) + v(t_2)t_{\bar{J}} + \frac{1}{2}a(t_2)t_{\bar{J}}^2 - \frac{1}{6}\bar{J}t_{\bar{J}}^3$$

$$= \bar{J}t_{\bar{J}}^3 + \frac{3}{2}\bar{J}t_{\bar{J}}^2 t_{\bar{a}} + \frac{1}{2}\bar{J}t_{\bar{J}}t_{\bar{a}}^2$$

Now we assume that the velocity bound is not violated, such that $t_4 = t_3$. Then due to symmetry of the profile we have:

$$x(t_7) = 2x(t_3) = 2\bar{J}t_{\bar{J}}^3 + 3\bar{J}t_{\bar{J}}^2 t_{\bar{a}} + \bar{J}t_{\bar{J}}t_{\bar{a}}^2 \qquad 4.36$$

By setting $\bar{x} = x(t_7)$ we can then solve $t_{\bar{a}}$ from the following:

$$\left(\bar{J}t_{\bar{J}}\right) \cdot t_{\bar{a}}^2 + \left(3\bar{J}t_{\bar{J}}^2\right) \cdot t_{\bar{a}} + \left(2\bar{J}t_{\bar{J}}^3 - \bar{x}\right) = 0 \qquad 4.37$$

$$t_{\bar{a}}^2 + \left(3t_{\bar{J}}\right) \cdot t_{\bar{a}} + \left(2t_{\bar{J}}^2 - \frac{\bar{x}}{\bar{J}t_{\bar{J}}}\right) = 0$$

As $t_{\bar{a}}$, must be positive to make sense, the solution is:

$$t_{\bar{a}}\& = -1\frac{1}{2}t_{\bar{J}} + \frac{1}{2}\sqrt{9t_{\bar{J}}^2 - 8t_{\bar{J}}^2 + \frac{4\bar{x}}{\bar{J}t_{\bar{J}}}}\& = -1\frac{1}{2}t_{\bar{J}} + \frac{1}{2}\sqrt{t_{\bar{J}}^2 + \frac{4\bar{x}}{\bar{J}t_{\bar{J}}}} \qquad 4.38$$

Note that $t_{\bar{a}} \geq 0$ follows immediately from $\bar{x} \geq 2\bar{J}t_{\bar{J}}^3$. Hence, we now have determined $t_{\bar{a}}$, under the assumption that the velocity bound will not be violated: Equation (4.38) establishes steps 8 and 9 of the algorithm.

Step 10 introduces the velocity bound again; the maximal velocity is obtained at t_3 Equation (4.35):

$$\hat{v} = v(t_3) = \bar{J}t_{\bar{J}}^2 + \bar{J}t_{\bar{J}}t_{\bar{a}} \qquad 4.39$$

If $\hat{v} > \bar{v}$ the bound is violated and we can recalculate $t_{\bar{a}}$ as follows:

$$t_{\bar{a}} = \frac{\bar{v} - \bar{J}t_{\bar{J}}^2}{\bar{J}t_{\bar{J}}} = \frac{\bar{v}}{\bar{J}_{\bar{J}}} - t_{\bar{J}} = \frac{\bar{v}}{\bar{a}} - t_{\bar{J}} \qquad 4.40$$

Now we have determined $t_{\bar{J}}$ and $t_{\bar{a}}$ to be the time-optimal solution under the restriction of the given bounds (step 2).

As the next lower derivative is already v, we can skip step 12 and continue with step 13: the determination of the constant velocity time interval $t_{\bar{v}}$ such that the required total displacement $\bar{x}$ is obtained. From the previous calculations follows that $t_{\bar{v}} = 0$ if $t_{\bar{a}}$ is according to Equation (4.38). But if $t_{\bar{a}}$ is reduced according to Equation (4.40), we will have $x(t_7) < \bar{x}$ and we must add a constant velocity phase to the trajectory such that $x(t_4) - x(t_3) = \bar{x} - 2x(t_3)$. With Equation (4.35) this implies that we can calculate $t_{\bar{v}}$ as:

$$t_{\bar{v}} = \frac{\bar{x} - 2\bar{J}t_{\bar{J}}^3 - 3\bar{J}t_{\bar{J}}^2 t_{\bar{a}} - \bar{J}t_{\bar{J}}t_{\bar{a}}^2}{\bar{v}} \qquad 4.41$$

This completes the calculation of the characteristics of the time-optimal third order profile for a given distance $\bar{x}$ and given bounds $\bar{v}, \bar{a}$ and $\bar{J}$. The total displacement $\bar{x}$ can be expressed as a function of $\bar{J}$ and the times $t_{\bar{J}}, t_{\bar{a}}$ and $t_{\bar{v}}$:

$$\begin{aligned} \bar{x} &= \bar{J}\left(2t_{\bar{J}}^3 + 3t_{\bar{J}}^2 t_{\bar{a}} + t_{\bar{J}}t_{\bar{a}}^2\right) + v(t_3)t_{\bar{v}} \\ &= \bar{J}\left(2t_{\bar{J}}^3 + 3t_{\bar{J}}^2 t_{\bar{a}} + t_{\bar{J}}t_{\bar{a}}^2 + t_{\bar{J}}^2 t_{\bar{v}} + t_{\bar{J}}t_{\bar{a}}t_{\bar{v}}\right) \end{aligned} \qquad 4.42$$

4.4 Feedforward and Feedback Design (Cascade Control)

AC servo motor is controlled by a cascaded velocity/position/acceleration controller with feedforward gains as shown in (Figure 4.2). Controllers have been tuned to obtain fast response with minimal following error and no overshoot in a point-to-point move.

The controller gains of these cascaded loops are adjusted (tuned) starting from the inner most loop and working outward. After the inner loop and the velocity loop are tuned.

4.4.1 Velocity Feedforward

The velocity feedforward can eliminate the following error. However, if the K_{vff} gain is set too high, the velocity feedforward will cause overshoot because it will create large error spikes in the acceleration/deceleration regions of the velocity profile (Figure 4.19).

Setting the K_{vff} to the same value as the K_d was our beginning point, which lead us to acceptable results. User defined input values UDI are placed on the [Table 5-3] produces graphs on the [Figure 4.18].
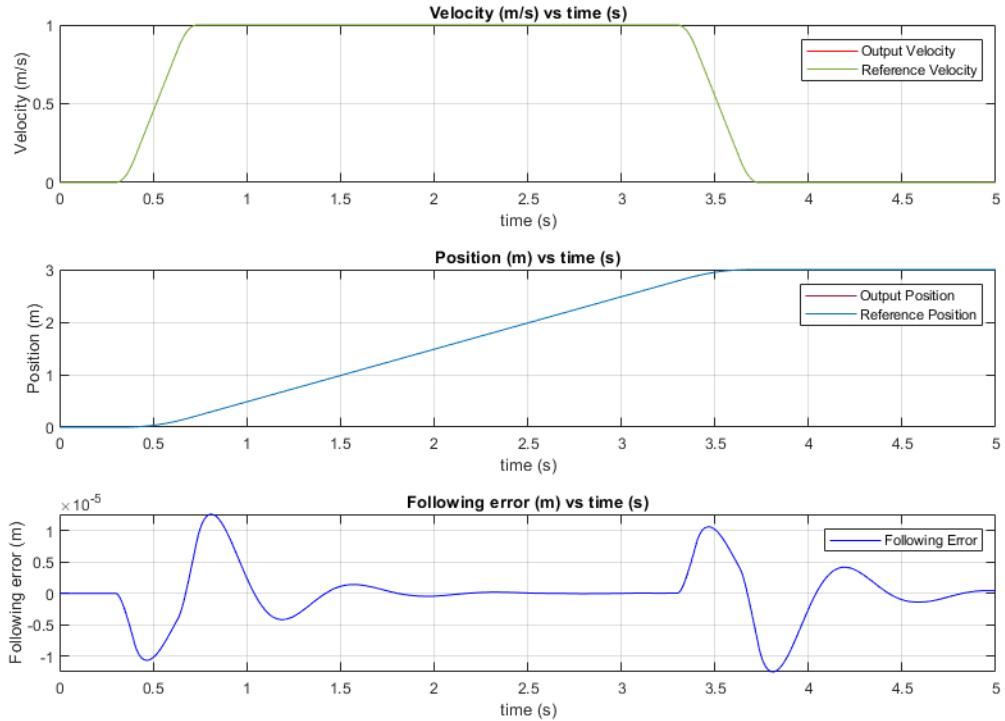
Figure 4.18 System response when velocity feedforward gain is 0.15 (Tuned)
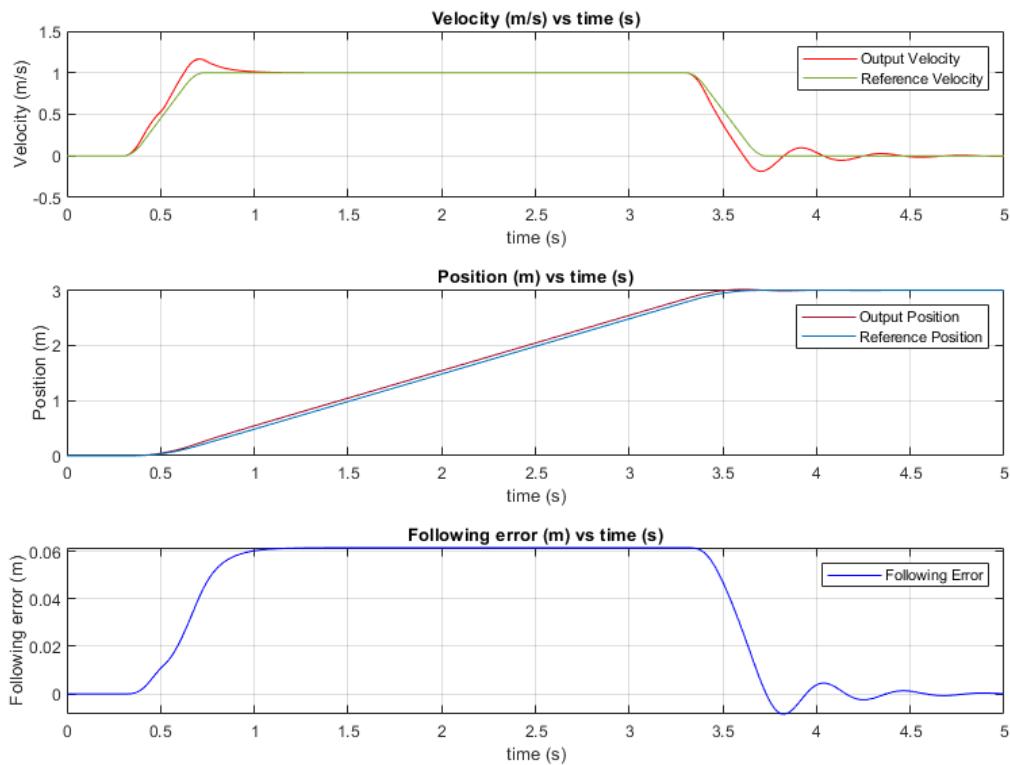
Figure 4.19 System response when velocity feedforward gain is 0.20 (untuned)

4.4.2 Acceleration Feedforward

The overshoot due to high-velocity feedforward gain can be addressed by adding acceleration feedforward. It eliminates the overshoot leading to an overall system with fast response and stiff disturbance rejection.
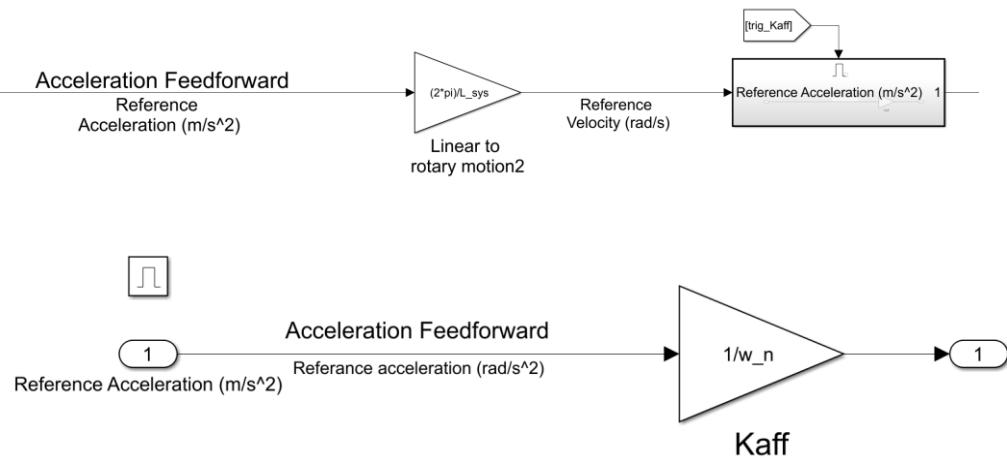

Figure 4.20 Acceleration feedforward enabled subsystem

To adjust K_{aff}, position input coming from a trapezoidal velocity profile have been applied as input while capturing the actual position and the following error. K_{vff} is increased until the following error is reduced as low as possible without making the system unstable.

Acceleration feedback gain determines the acc feedforward gain so that K_{aff} should be $\frac{1}{w_n}$. Natural frequency of the system is calculated by the following Equation (5.1)

$$w_n = \sqrt{\frac{k}{m}}$$

4.43

$$k = \frac{E * A}{l}$$

4.44

A is the area of the shaft, k is the axial stiffness value of the shaft, l is the length of the shaft. Damping value of the system has been set to "0.01". Natural frequency $w_n = 633\ Hz$.

It should be noted that the K_{aff} gain is implicitly proportional to the inertia (mass). Therefore, it would need to be adjusted every time the load inertia changed.

Not all industrial motion controllers have the acceleration feedforward feature. Our system uses the acceleration feedforward if and only if there is an oscillation on the system. When oscillation occur on the system. Acceleration feedback and feedforward can be activated with turning the switch seen on the (Figure 5.6). With a help of enabled subsystem on the acceleration feedforward (Figure 4.20), acceleration feedback and feedforward either work or doesn't work. When oscillator switch turned on (Figure 4.3). When oscillator switch is turned off (Figure 5.3). System error is highly acceptable without acceleration feedback and feedforward when there is not any oscillation.

4.4.3   Position Controller

Build of PID position controller involves K_p gain that placed before saturation block. K_d gain that placed on the velocity feedback and subsystem can be seen on the (Figure 4.21). This subsystem has made to be able to control integrator.

We set limits on the maximum amount of current the integrator can command (5.8, -5.8) This is called integrator clamping. These limits are gathered from spreadsheet [7] chosen AC servo motor (Hiwin FRMS D2-PPN06A 750W).
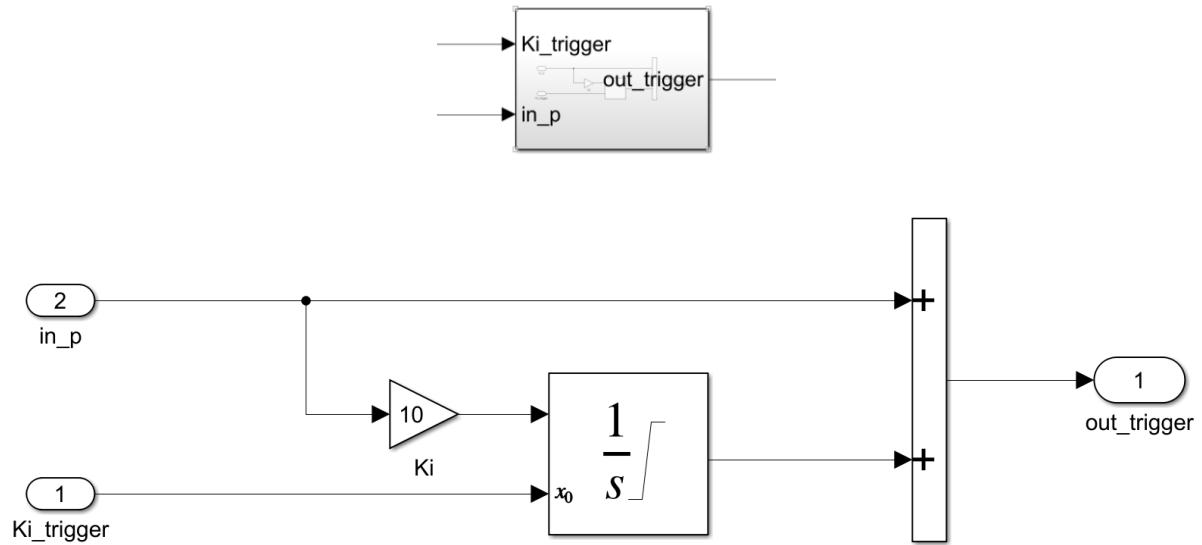
Figure 4.21 PI controller

Often the main role of the integrator is to overcome the friction. Hence, the integrator output can be clamped at a level just sufficient to overcome the friction torque in the system and not much more. This enables us to reduce the potential overshoot and oscillation effects of the integrator.
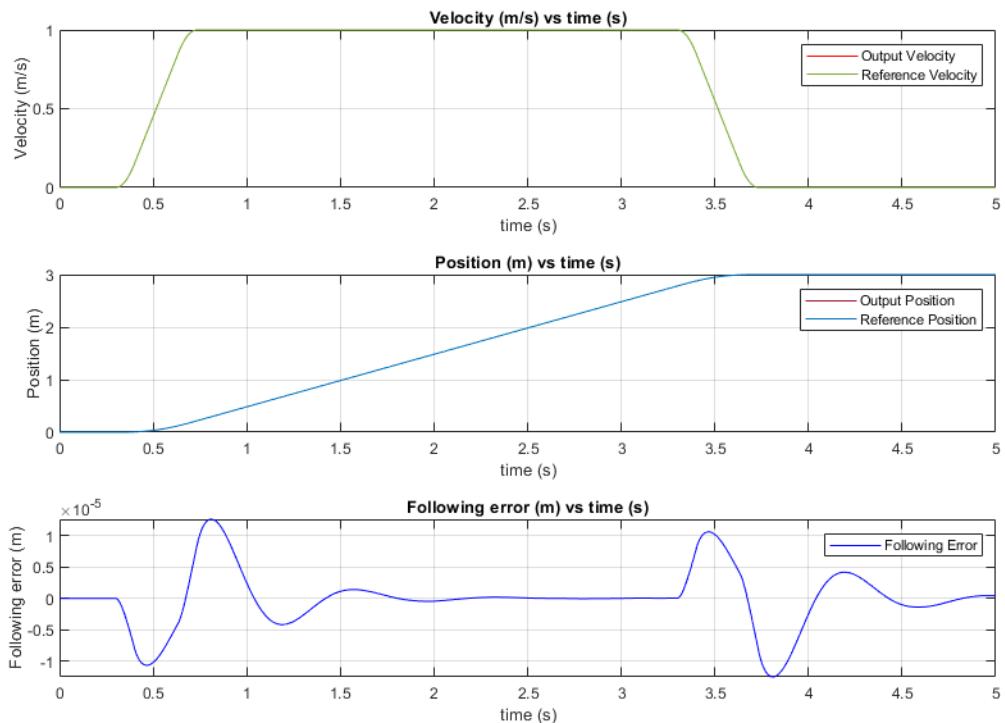
Figure 4.22 System response when integrator with clamping
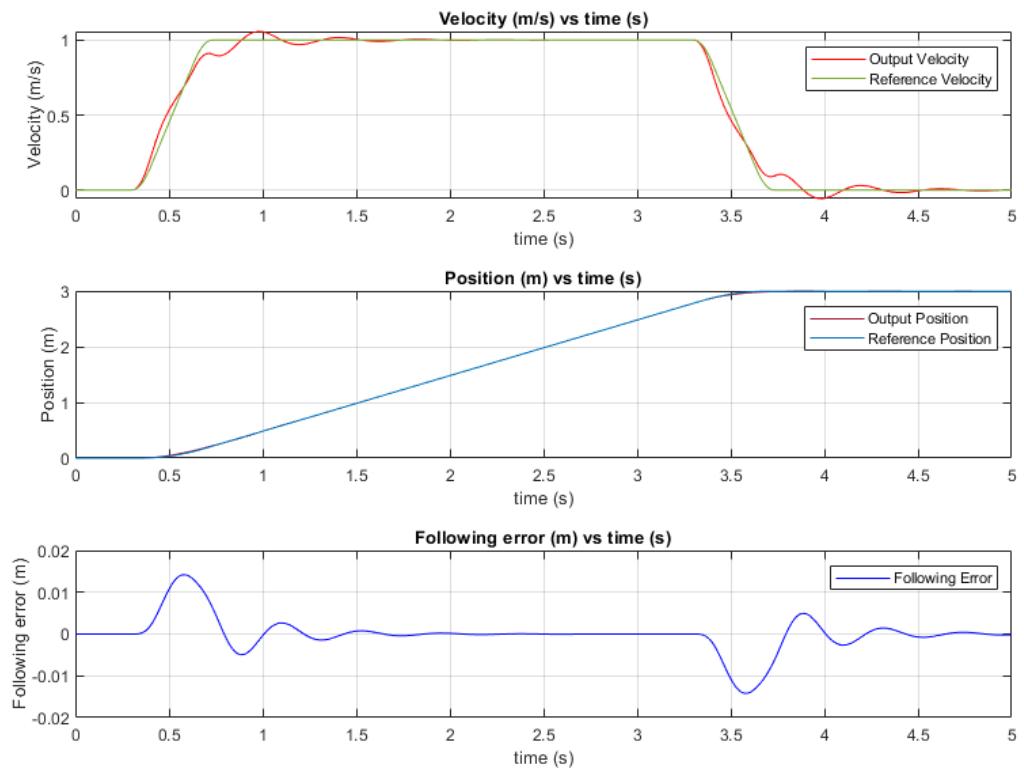
Figure 4.23 System response when integrator is continuous and without clamping

5 VIBRATION SUPPRESSION

5.1 Trajectory Planning

It has been detailly clarified that how we produced 3rd order trajectory on the (4.3). In this section, it will be shown how order of the trajectory changes the following error graph.
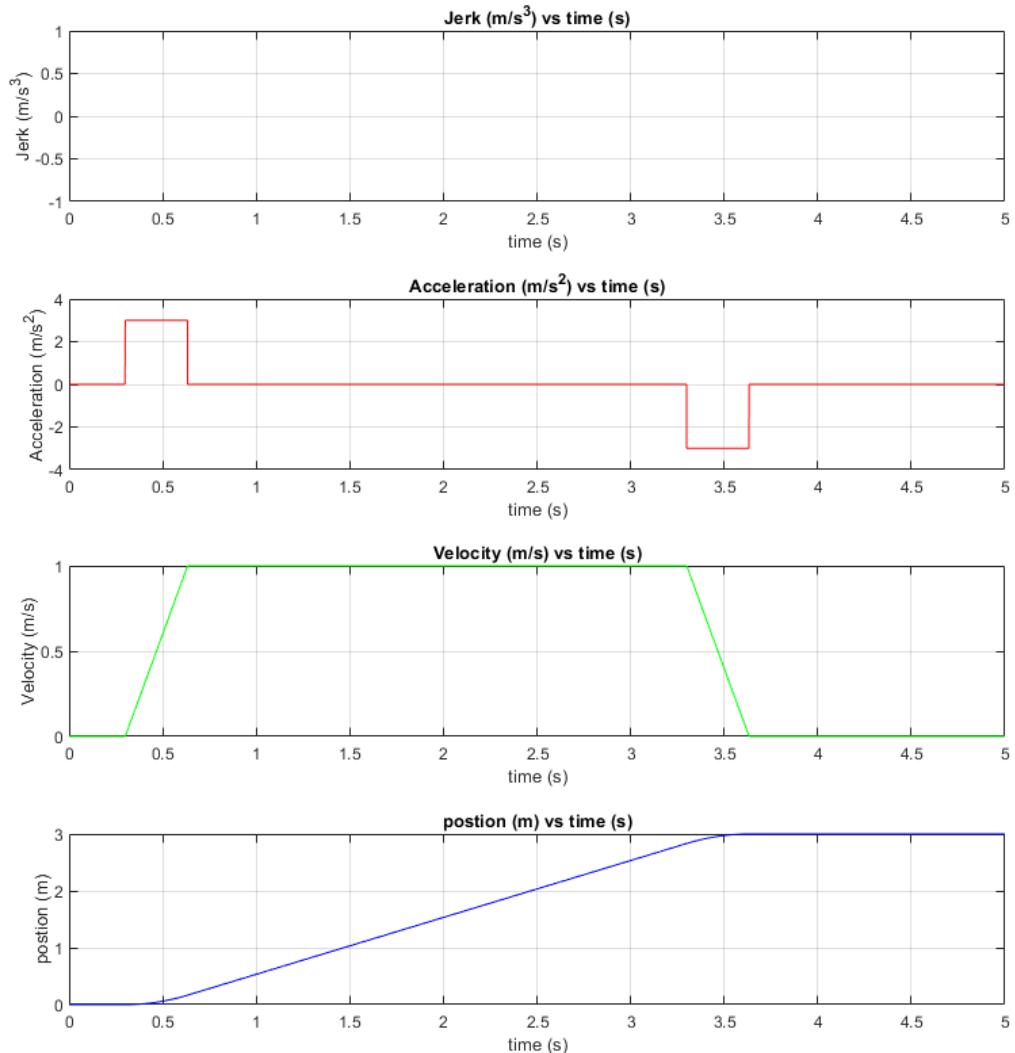
Figure 5.1 2nd order trajectory

Displacement	$3\ (m)$
Bound on velocity	$1\ (m/s)$
Bound on acceleration	$3\ (m/s^2)$
Bound on jerk	$\infty\ (m/s^3)$

Table 5-1 2nd order trajectory values
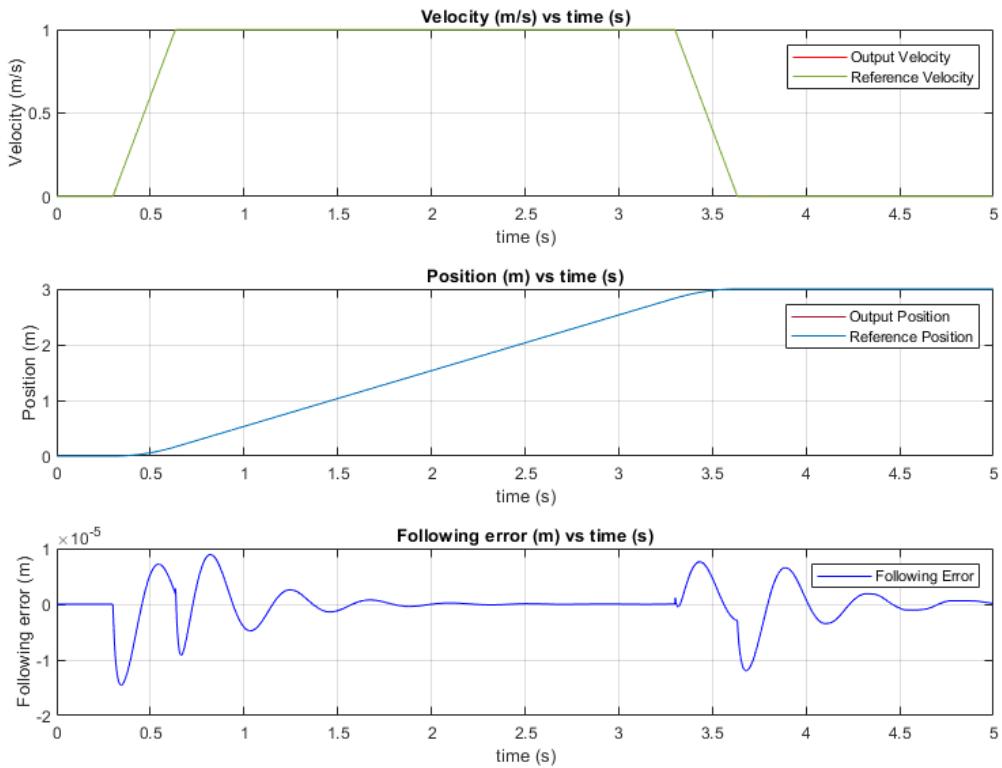
Figure 5.2 System response when trajectory is 2nd order instead of 3rd

It can be seen on the (Figure 5.2) order of the trajectory changing the following error graph. Following error is increased slightly. But most importantly there is a little spikes and breakup on the following error graph. That means 2nd order trajectory is not suitable for this motion control system.

5.2   Acceleration Feedback

This chapter contains working principle of acceleration feedback as a vibration reduction element. Oscillator has used to be able to simulate vibration that can occur on the table when machine tool is cutting the workpiece.

Integration of the oscillator and acceleration feedback to the system can be seen when oscillator switch turned on (Figure 4.2). When oscillator switch is turned off (Figure 5.3). These two different states of the model explain how oscillator switch works.
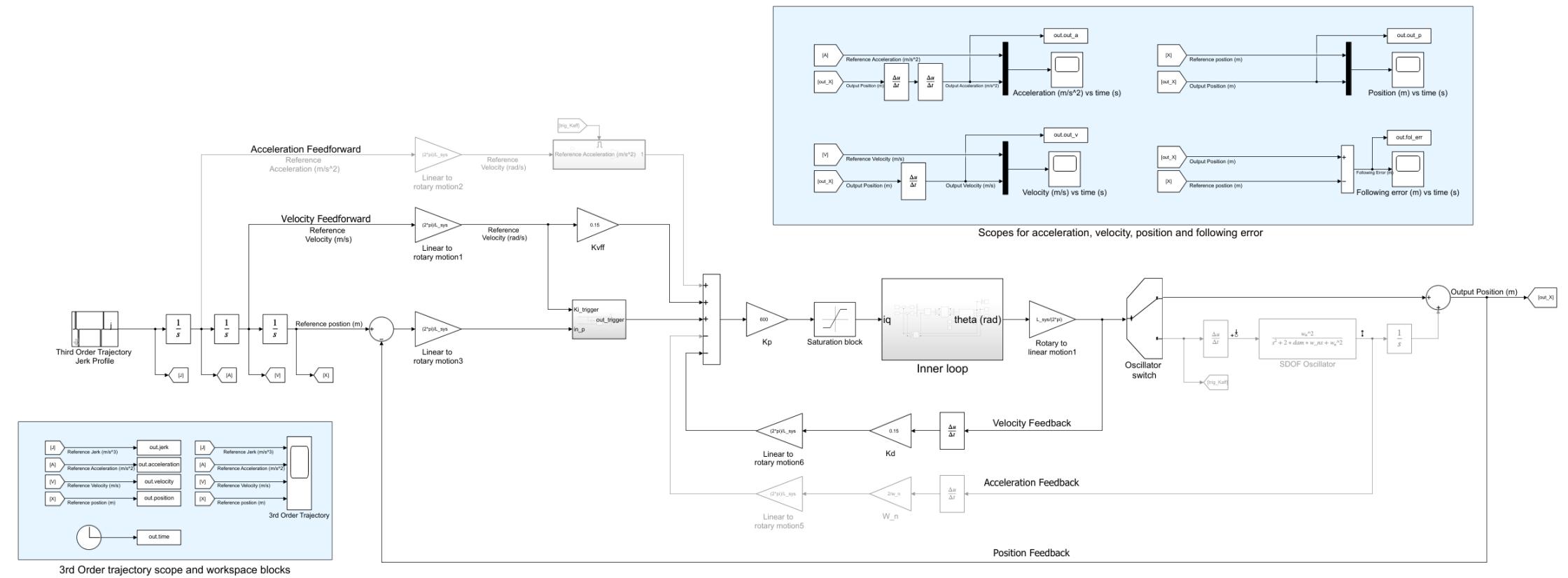
Figure 5.3 CNC machine tool feed drive control system model when oscillator switch turned off
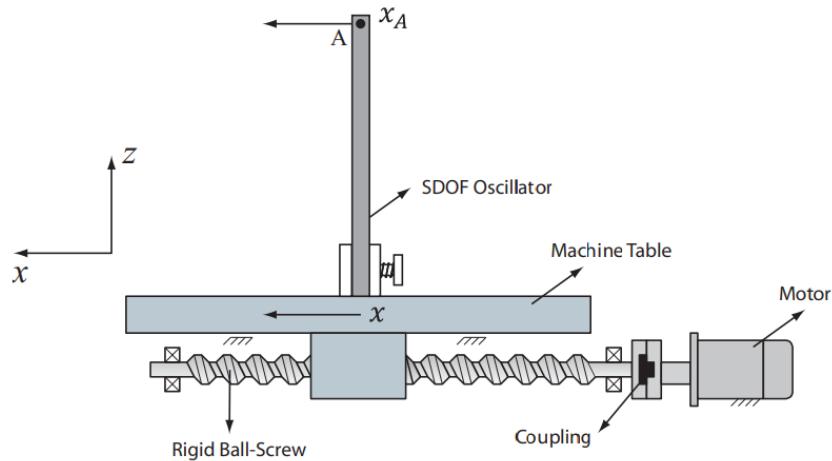
Figure 5.4 SDOF oscillator mounted on the ball screw table.

Drive based vibration reduction proposed by Dietmair and Verl [3] is an effective, and at the same time low-cost method of increasing damping ratio for a specific mode in production machines and robotic arms. This method increases the damping ratio by modifying the velocity loop when there is a cascade control structure where position and velocity loops can be disguised.

$$\frac{x_A}{x} \simeq \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad 5.1$$

The dynamics of this bar can be expressed using Equation (5.1). Where x_a represents the vibrations at the oscillator's tip point and x indicates the ball-screw's table displacement. The inertial forces generated by the rapid positioning of the machine table cause the flexible beam to vibrate. Drive based vibration reduction concerns the algorithms which enhance the damping ratio of a target mode and lead to faster vibration attenuation of the flexible beam.
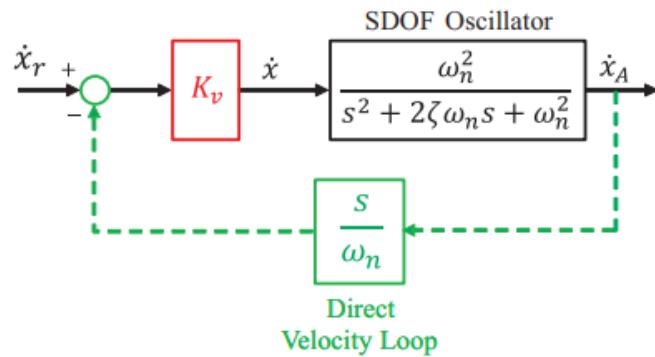
Figure 5.5 Velocity loop of the ball screw drives with single degree of freedom oscillator and active damping. The indirect velocity loop replaced by K_v

(Figure 4.2) presents the control loop block diagram of the ball-screw drive when a cascade control law is implemented. The velocity, position and acceleration loop all use the linear encoder feedback for accurate positioning and trajectory tracking.

An easy to implement approach of suppressing the oscillator's vibrations, is to include its dynamics in the control loop of the ball-screw drive; for this purpose, 90° phase shift with unity gain at the natural frequency (ω_0) is made in the direct velocity loop as shown in (Figure 5.5).

The blocks show: Oscillator switch → $\frac{\Delta u}{\Delta t}$ → SDOF Oscillator $\frac{w_n{}^2}{s^2 + 2*dam*w\_ns + w_n{}^2}$ → $\frac{1}{s}$, with [trig_Kaff] and summation.
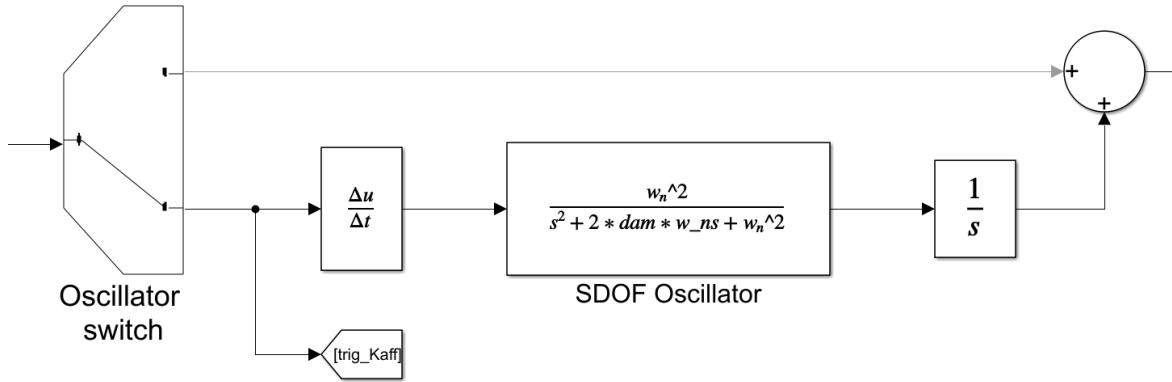
Figure 5.6 SDOF Oscillator and switch system
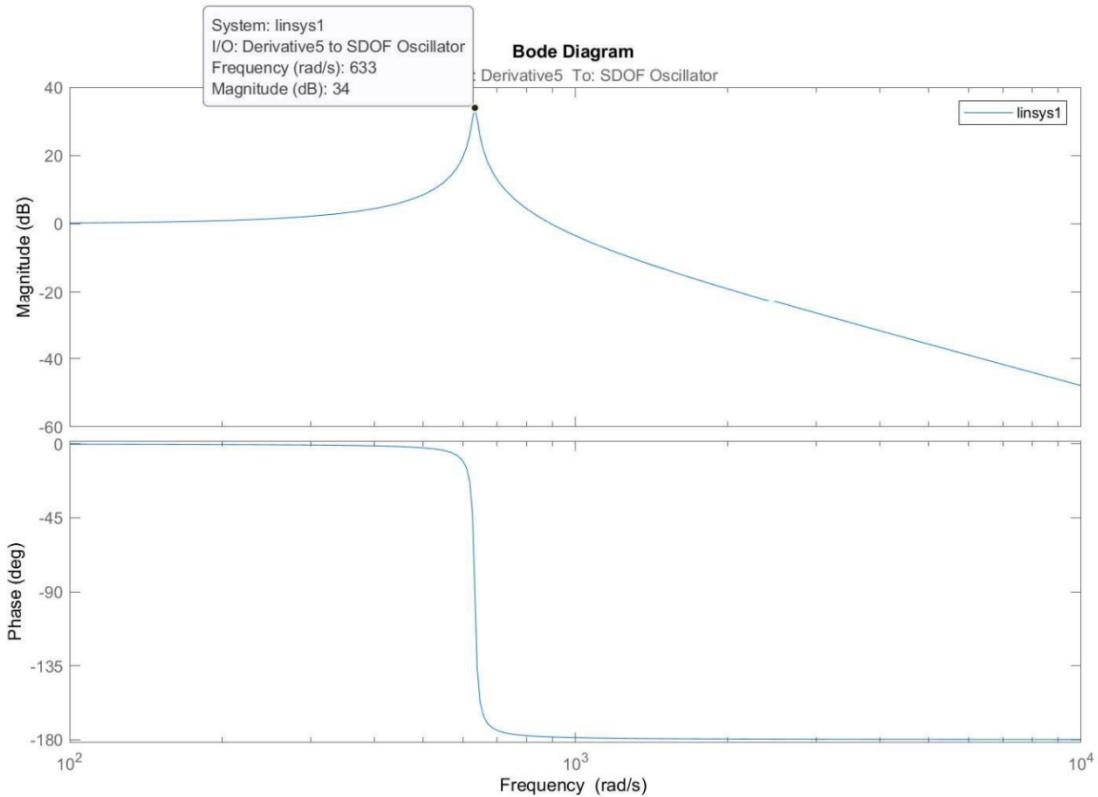
Figure 5.7 Bode plot of the SDOF Oscillator when natural frequency at 633Hz
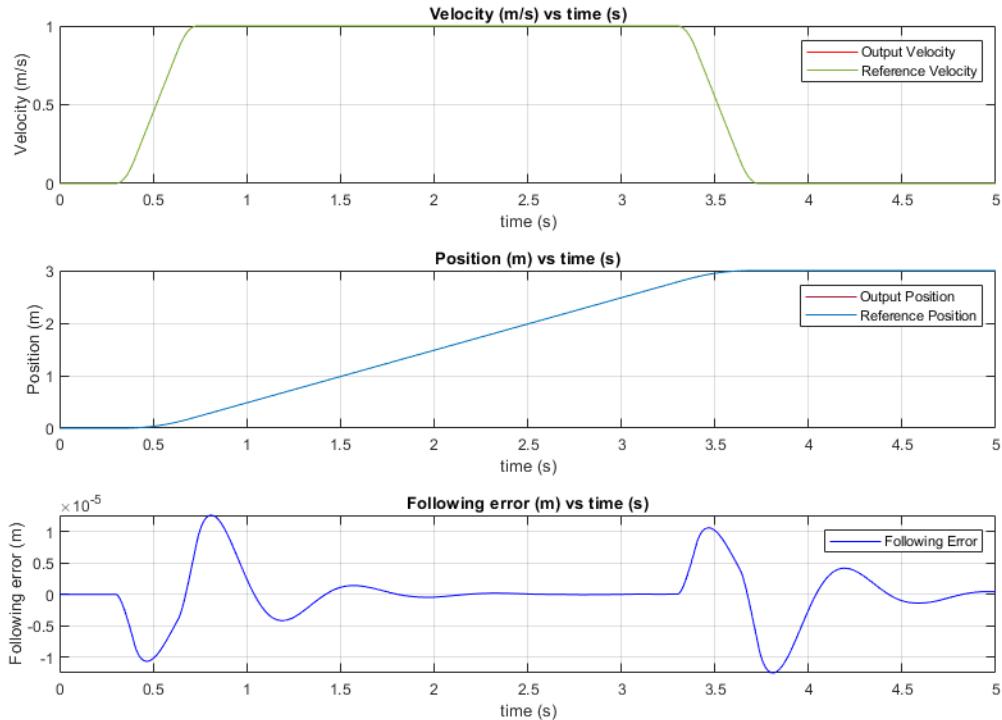
Figure 5.8 System response when oscillator and acc feedback activated
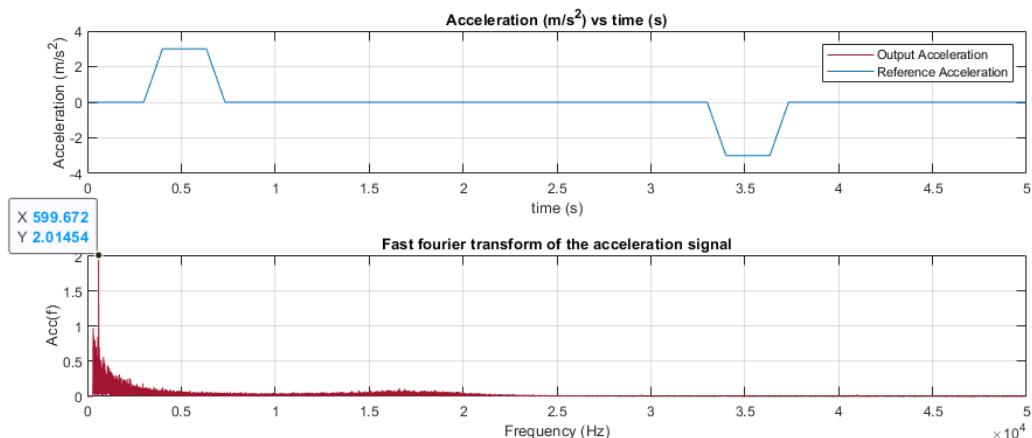
Figure 5.9 System Fast Fourier transformation when oscillator and acc feedback is activated

There we can see on the (Figure 5.8) and (Figure 5.9). When oscillator is activated acceleration feedback is doing great about reducing the error and keeping the acceleration output clean. Also following error is pretty reasonable as seen (Figure 5.8).

Fast Fourier transform of the acceleration signal can be seen on (Figure 5.9). This FFT graph

shows us which frequency is exited on the acceleration signal and it is 600Hz on this graph. This frequency shows systems natural frequency. See Appendix F for the MATLAB codes.
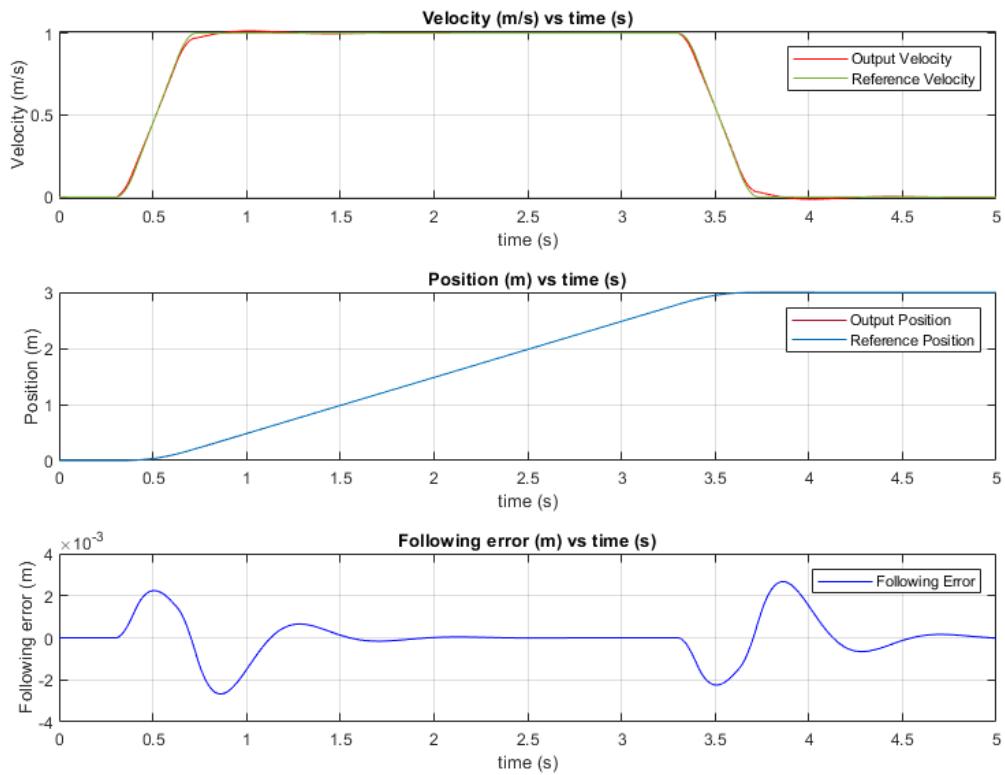
Figure 5.10 System response when oscillator is activated, acc feedback deactivated
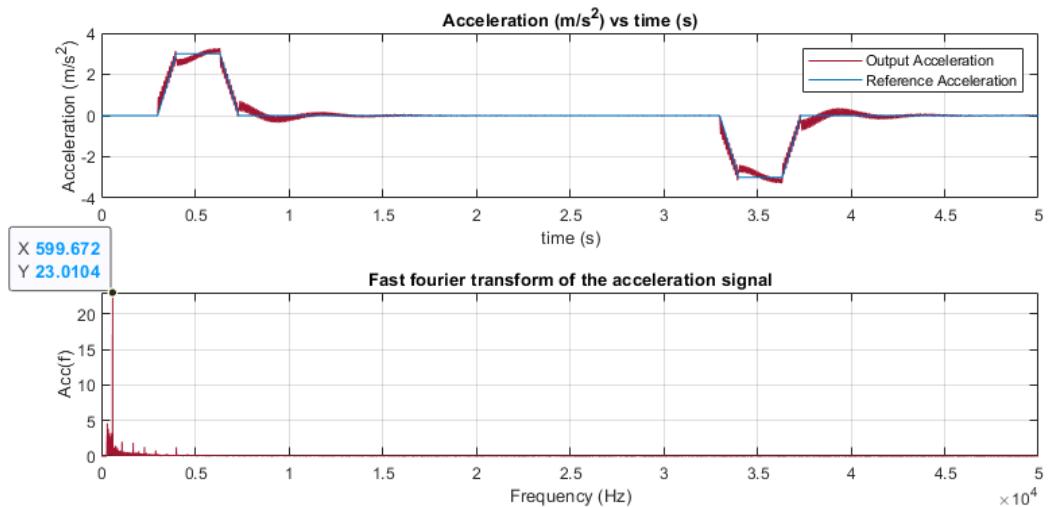
Figure 5.11 System Fast Fourier transformation when oscillator is activated, acc feedback deactivated
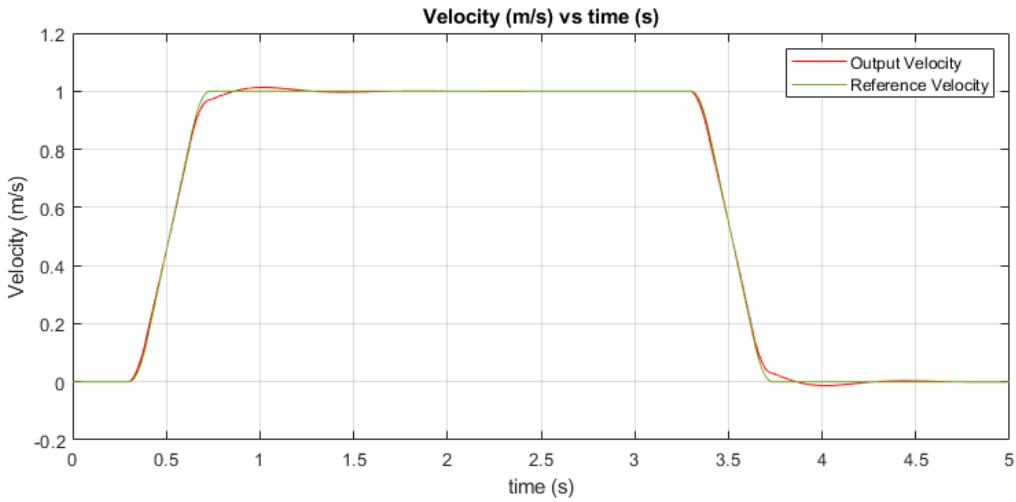
Figure 5.12 System velocity response zoomed in

When oscillator is active but acc feedback is deactivate system response has changed drastically as we can see on (Figure 5.10), (Figure 5.12) and (Figure 5.11)). Because there is not any acceleration feedback on the system, acceleration output graph is far away from the reference acceleration. In addition to that, this inconsistency on the acceleration side of the system reflected on the following error and increased it nearly two hundred times from before. And also, in the (Figure 5.11) Acc(f) amplitude is higher than before. So we can say that acceleration feedback is working as predicted and suppressing the vibration.
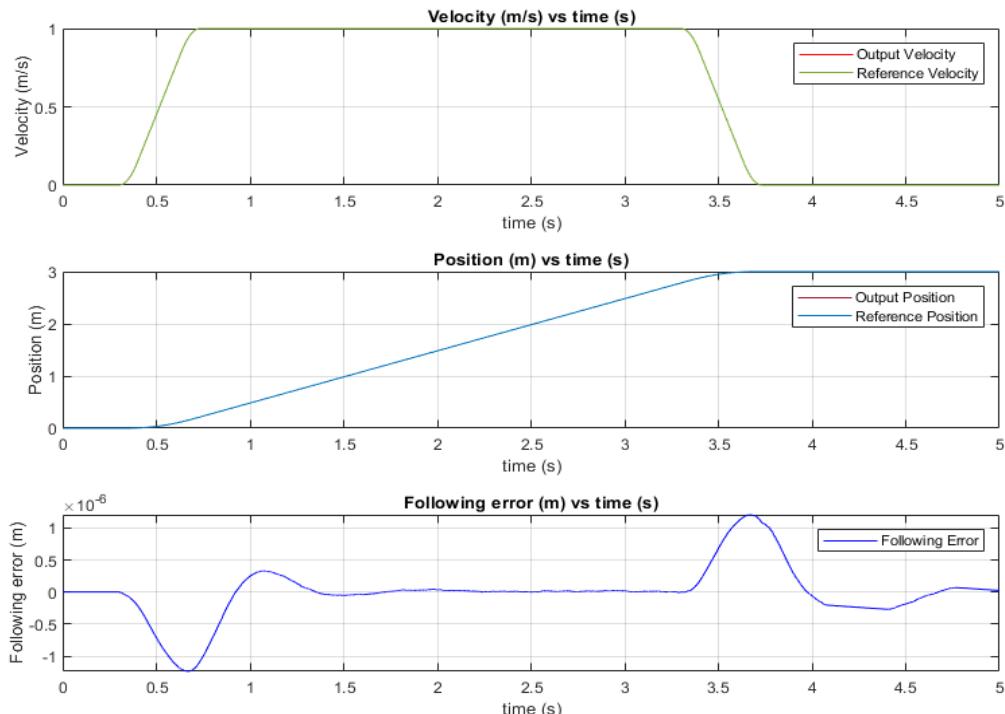
Figure 5.13 System response when oscillator and acc feedback deactivated

When oscillator switch is turned off (Figure 5.3) acceleration feedback, acceleration feedforward is not working, and system works without a disturbance of the oscillator. This changes on the system leads to a system response graphs as seen on the (Figure 5.13). If we analyze the system response. We can say that everything works smoothly as predicted before and following error is lower than any other state of the system.

In estimation theory, the root-mean-square deviation of an estimator is a measure of the imperfection of the fit of the estimator to the data. The RMS values of following error graphs can be seen on the (Table 5-2) and it can be seen that observations made before are consistent with RMS values.

Setup	Root-Mean-Square RMS values
Oscillator inactive	4.0421e-07
Oscillator and acc feedback active	4.4794e-06
Oscillator active acc feedback inactive	8.6328e-04

Table 5-2 Root-Mean-Square RMS values of tracking error

Simulation values for the motor parameters R, L_d, L_q, p, and PM (for λ_{PM}) etc. specified for the simulated motor by entering them into MATLAB (Appendix D).

Specified input values for: Motor: Hiwin FRMS D2-PPN06A 750W AC servo motor Ball-screw: Hiwin KK10020 980mm Ball-screw	
Inertia of rotating parts (J)	0.00014 [kg*m^2]
Viscous friction coefficient (B)	0.0003743 [Nm s/rad]
Direct-axis inductance (L_d)	0.001 [mH]
Quadrature-axis Inductance (L_q)	0.003 [mH]
Insulation Resistance (R)	10 [mΩ]
Pole number (p)	4
Rotor permanent magnet (λ_{PM})	0.1115
Viscous friction torque (T_f)	$7.8*10^{-3}$ [N.m]
DC bus voltage (V_{BUS})	297 [V]
Ball-screw lead	0.020 [m]
Saturation number	5.1
Natural frequency (w_n)	633 [Hz]
Damping ratio (ζ)	0.01

Table 5-3 Input values placed on UDI [Appendix D]

6 CONCLUSION AND FUTURE WORK

In this undergraduate thesis, design of the CNC machine tool feed drive control system model, and vibration suppression of the model has been made. Long-term research made from many sources sent by the instructor Associate Professor İbrahim Sina Kuseyri and other sources. Then CNC machine tool feed drive control system model designed on SIMULINK/MATLAB. It has been researched how to suppress vibration of the system.

It has been learned how to select AC servo motor and ball-screw drive for specific needs. Then we have chosen our motor and ball-screw drive form Hiwin catalog [7]. It has been learned how to do active damping of the structural vibration in machines.

Vibration suppression of the CNC machine tool feed drive control system has been made. Single degree of freedom oscillator has been used for this task and this is not an optimal way to simulate a vibration of this kind of a system. So, for the future work we would use lumped mass model discussed below (Figure 6.1).

The lumped mass model can reasonably reduce the number of degrees of freedom (DOF) of the simulation model while preserving the low-order modes of the system to simplify calculations. But in our case, we approached superficial about lumped system of the model. (Figure 6.1) defines the lumped mass model of a ball screw feed drive system better than our approach. This lumped mass model from [9] would be used for the future work.
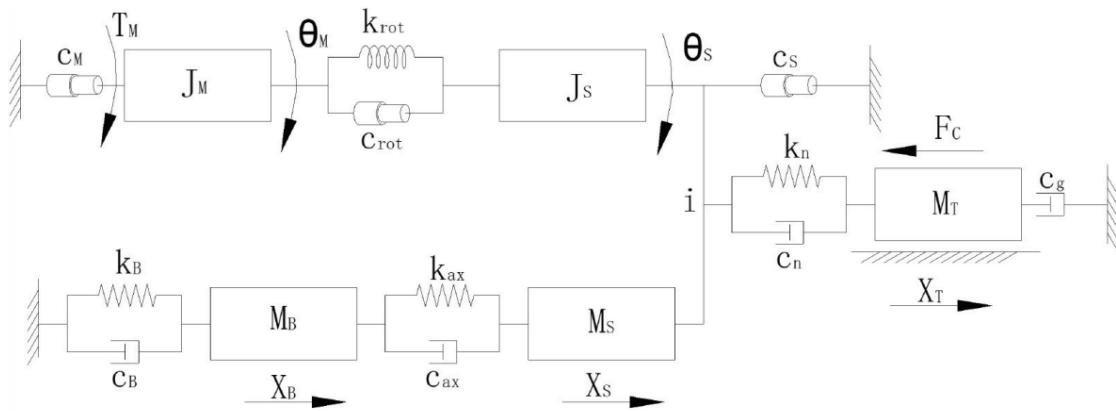
Figure 6.1 Lumped mass model of ball screw feed system.

The influence of the shaft on the rotational mode and axial mode of the drive system is explicitly included into the lumped mass model here. Therefore, the shaft is separated into two different branches, an axial branch and a rotational branch, while the coupling once more is realized

using constrained equations. Since all components are expressed by discrete springs and dampers, the rigidity values of shaft, coupling, and bearing are combined to an overall axial K_{ax} and rotational value K_{rot}.

In this model (Figure 6.1) the inertial component parameters are defined as the following: rotary inertia of servomotor J_M, screw shaft side equivalent rotary inertia J_S, mass of base M_B, screw shaft side equivalent mass M_S, and mass of the worktable M_T.

According to the information obtained in the light of a research and studies above, our CNC machine tool feed drive control system model design is finished. In addition to this, vibration suppression of this system has been carried out.

REFERENCES

[1] Hakan Gürocak, "Industrial Motion Control" 2016

[2] Amir Hossein Hadi Hosseinabadi "Modeling and Active Damping of Structural Vibrations in Machine Tools" *Master's thesis, The University of British Colombia, Vancouver,* November 2013.

[3] A. Dietmair and A. Verl, "Drive based vibration reduction for production machines" *Modern Machinery Science Journal, vol. 3, pp.* 129–133, 2009.

[4] Paul Lambrechts, "Trajectory planning and feedforward design for electromechanical motion systems" *Technische Universiteit Eindhoven.* 17th February 2003

[5] Haojin Yang, Zihao Wang, Tao Zhang, Fuxin Du "A review on vibration analysis and control of machine tool feed drive systems" *Springer-Verlag London Ltd., part of Springer Nature* 2020, 28 January 2020

[6] Y. Altintas, A. Verl, C. Brecher, L. Uriarte, G. Pritschow, "Machine tool feed drives" *Manufacturing Automation Laboratory, University of British Columbia, Vancouver, Canada,* 2011

[7] Single-Axis Robot Technical Information by HIWIN Technologies Corp. *16th Edition, April* 2019

[8] Danielle Collins "Servo motor torque curves: What you need to know" 2018

[9] Liang Luo and Weimin Zhang "Electromechanical Co-Simulation for Ball Screw Feed Drive System" 2018

Appendix A

MATLAB codes inside of the fcn1 block from park transform model:

```
function y = fcn(u)
y = u(1)*cos(u(4))+u(2)*cos(u(4)-2*pi/3)+u(3)*cos(u(4)+2*pi/3);
```

MATLAB codes inside of the fcn2 block from park transform model:

```
function y = fcn(u)
y = -u(1)*sin(u(4))-u(2)*sin(u(4)-2*pi/3)-u(3)*sin(u(4)+2*pi/3);
```

Appendix B

MATLAB codes inside of the fcn1 block from inverse park transform model:

```
function y = fcn(u)
y = u(1)*cos(u(3))-u(2)*sin(u(3));
```

MATLAB codes inside of the fcn2 block from inverse park transform model:

```
function y = fcn(u)
y = u(1)*cos(u(3)-2*pi/3)-u(2)*sin(u(3)-2*pi/3);
```

MATLAB codes inside of the fcn3 block from inverse park transform model:

```
function y = fcn(u)
y = u(1)*cos(u(3)+2*pi/3)-u(2)*sin(u(3)+2*pi/3);
```

Appendix C

MATLAB codes inside of the fcn1 block from electrical model:

```
function y = fcn(u)
R_sys = 10;
y = R_sys*u(1);
```

MATLAB codes inside of the fcn2 block from electrical model:

```
function y = fcn(u)
Lq_sys = 0.003;
y = Lq_sys*u(2);
```

MATLAB codes inside of the fcn3 block from electrical model:

```
function y = fcn(u)
R_sys = 10;
y = R_sys*u(2);
```

MATLAB codes inside of the fcn4 block from electrical model:

```
function y = fcn(u)
Ld_sys = 0.001;
PM_sys = 0.1115;
y = Ld_sys*u(1)+PM_sys;
```

Appendix D

```matlab
%% UDI USER DEFINED INPUTS
clear
clc

%% 3rd Order Profile Generating

    x_trj    = 3;           % Displacement [m]
    vmax_trj = 1;           % Bound on velocity [m/s]
    amax_trj = 3;           % Bound on acceleration [m/s^2]
    jmax_trj = 30;          % Bound on jerk [m/s^3]

%% Specified input values for:
    % Motor:         Hiwin FRMS D2-PPN06A 750W AC servo motor
    % Ball-screw:    Hiwin KK10020 980mm Ball-screw

    J_sys = 0.00014;        % Inertia of rotating parts [kg*m^2]
    B_sys = 0.0003743;      % Viscous friction coefficient [Nm s/rad]

    Ld_sys = 0.001;         % Direct-axis inductance [mH]
    Lq_sys = 0.003;         % Quadrature-axis Inductance [mH]
    R_sys = 10;             % Insulation Resistance [mΩ]
    p_sys = 4;              % Pole number
    PM_sys = 0.1115;        % Rotor permanent magnet
    Tf_sys = 7.8*10^-3;     % Viscous friction torque
    Vdc_sys = 297;          % DC bus voltage for 220V ac motor 220*1.35 = 297 [V]
    L_sys = 0.020;          % Ball-screw lead 0.020m
    stur = 5.1;             % Saturation number

    w_n = 633;              % Natural frequency
    dam = 0.01;             % Damping ratio

%% Running simulation
    disp('.Running simulation...')

    sim("sim_sys.mdl")

%% Plotting of scope's graphs
    %% Velocity (m/s) vs time (s)
figure(1)
    tiledlayout(3,1)
    ax1 = nexttile;

    plot(ax1,ans.time,ans.out_v, '-r')
    hold on
    grid on
    plot(ax1,ans.time,ans.velocity, 'color','#77AC30')
    hold off

        legend('Output Velocity', 'Reference Velocity')
        title('Velocity (m/s) vs time (s)')
        xlabel('time (s)')
        ylabel('Velocity (m/s)')

    %% Position (m) vs time (s)
    ax2 = nexttile;
```

```matlab
plot(ax2,ans.time,ans.out_p, 'color','#A2142F')
hold on
grid on
plot(ans.time,ans.position, 'color','#0072BD')
hold off

    legend('Output Position', 'Reference Position')
    title('Position (m) vs time (s)')
    xlabel('time (s)')
    ylabel('Position (m)')

%% Following error (m) vs time (s)
ax3 = nexttile;

plot(ax3,ans.time,ans.fol_err, '-b')
grid on

    legend('Following Error')
    title('Following error (m) vs time (s)')
    xlabel('time (s)')
    ylabel('Following error (m)')

%% Plotting of generated 3rd order trajectory profile
figure(2)
    tiledlayout(4,1)
    tilax1 = nexttile;
    plot(tilax1,ans.time,ans.jerk, '-k');
    grid on

        title('Jerk (m/s^3) vs time (s)')
        xlabel('time (s)')
        ylabel('Jerk (m/s^3)')

    tilax2 = nexttile;
    plot(tilax2,ans.time,ans.acceleration, '-r');
    grid on

        title('Acceleration (m/s^2) vs time (s)')
        xlabel('time (s)')
        ylabel('Acceleration (m/s^2)')

    tilax3 = nexttile;
    plot(tilax3,ans.time,ans.velocity, '-g');
    grid on

        title('Velocity (m/s) vs time (s)')
        xlabel('time (s)')
        ylabel('Velocity (m/s)')

    tilax4 = nexttile;
    plot(tilax4,ans.time,ans.position, '-b');
    grid on

        title('postion (m) vs time (s)')
        xlabel('time (s)')
        ylabel('postion (m)')
```

```matlab
%% Acceleration (m/s^2) vs time (s)
figure

    plot(ans.time,ans.out_a, 'color','#A2142F')
    hold on
    grid on
    plot(ans.time,ans.acceleration, 'color','#0072BD')
    hold off

        legend('Output Acceleration', 'Reference Acceleration')
        title('Acceleration (m/s^2) vs time (s)')
        xlabel('time (s)')
        ylabel('Acceleration (m/s^2)')

%% Velocity (m/s) vs time (s)
figure

    plot(ans.time,ans.out_v, '-r')
    hold on
    grid on
    plot(ans.time,ans.velocity, 'color','#77AC30')
    hold off

        legend('Output Velocity', 'Reference Velocity')
        title('Velocity (m/s) vs time (s)')
        xlabel('time (s)')
        ylabel('Velocity (m/s)')
%%
    disp('...finished')
```

Appendix E

```matlab
% 3rd ORDER TRAJECTORY PLANNING [4]
function [t,jd]=make3(varargin)

% [t,jd] = make3(p,v,a,j,Ts)
%
% Calculate timing for symmetrical third order profiles.
%
% inputs:
%       p    = desired path (specify positive)              [m]
%       v    = velocity bound (specify positive)            [m/s]
%       a    = acceleration bound (specify positive)        [m/s2]
%       j    = jerk bound (specify positive)                [m/s3]
%       Ts   = sampling time (optional, if not specified or 0: continuous time)
%
% outputs:
%       t(1) = constant jerk phase duration
%       t(2) = constant acceleration phase duration (default 0)
%       t(3) = constant velocity phase duration (default 0)
%
%        t1                       t1
%    j   .-.                      .-.
%        | |                      | |
%        | | t2        t3       t2 | |
%       -'-'----.-.-------.-.----' '--
%               | |       | |
%               | |       | |
%    -j         '-'       '-'
%              t1         t1
%
% In case of discrete time, jerk bound j is reduced to jd
% Paul Lambrechts, TUE fac. WTB, last modified: March 4, 2003.

if nargin < 4 | nargin > 5
   help make3
   return
else
   p=abs(varargin{1});
   v=abs(varargin{2});
   a=abs(varargin{3});
   j=abs(varargin{4});
   if nargin == 4
       Ts=0;
   else
       Ts=abs(varargin{5});
   end
end

if length(p)==0 | length(v)==0 | length(a)==0 | length(j)==0 | length(Ts)==0
   disp('ERROR: insufficient input for trajectory calculation')
   return
end

tol = eps;                 % tolerance required for continuous time calculations
jd = j;                    % required for discrete time calculations
```

```matlab
% Calculation t1
t1 = (p/(2*j))^(1/3) ;      % largest t1 with bound on jerk
if Ts>0
    t1 = ceil(t1/Ts)*Ts;
    jd = 1/2*p/(t1^3);
end
% velocity test
if v < jd*t1^2              % v bound violated ?
    t1 = (v/j)^(1/2) ;      % t1 with bound on velocity not violated
    if Ts>0
        t1 = ceil(t1/Ts)*Ts;
        jd = v/(t1^2);
    end
end
% acceleration test
if a < jd*t1                % a bound violated ?
    t1 = a/j ;              % t1 with bound on acceleration not violated
    if Ts>0
        t1 = ceil(t1/Ts)*Ts;
        jd = a/t1;
    end
end
j = jd;                     % as t1 is now fixed, jd is the new bound on jerk

% Calculation t2
t2 = (t1^2/4+p/j/t1)^(1/2) - 3/2*t1 ;  % largest t2 with bound on acceleration
if Ts>0
    t2 = ceil(t2/Ts)*Ts;
    jd = p/( 2*t1^3 + 3*t1^2*t2 + t1*t2^2 );
end
if abs(t2)<tol t2=0; end                % for continuous time case
% velocity test
if v < (jd*t1^2 + jd*t1*t2)             % v bound violated ?
    t2 = v/(j*t1) - t1 ;                % t2 with bound on velocity not violated
    if Ts>0
        t2 = ceil(t2/Ts)*Ts;
        jd = v/( t1^2 + t1*t2 );
    end
end
if abs(t2)<tol t2=0; end        % for continuous time case
j = jd;                         % as t2 is now fixed, jd is the new bound on jerk

% Calculation t3
t3 = (p - 2*j*t1^3 - 3*j*t1^2*t2 - j*t1*t2^2)/v ; % t3 with bound on velocity
if Ts>0
    t3 = ceil(t3/Ts)*Ts;
    jd = p/( 2*t1^3 + 3*t1^2*t2 + t1*t2^2 + t1^2*t3 + t1*t2*t3 );
end
if abs(t3)<tol t3=0; end        % for continuous time case

% All time intervals are now calculated
t=[ t1 t2 t3 ] ;
if min(t)<0
    disp('ERROR: negative values found')
end
% Finished.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Appendix F

```matlab
%% Fast Fourier Transformation

    freq = length(out.time);

    fft_v = out.out_a;
    fft_v = rot90(fft_v);
    ref_v = out.acceleration;
    ref_v = rot90(ref_v);
    fft_v2 = fft_v-ref_v;

    nfft = length(fft_v2)/5;
    nfft2 = 2^nextpow2(nfft);
    ff = fft(fft_v2,nfft2);
    fff = ff(1:nfft2/2);
    xfft = nfft*(0:nfft2/2-1)/nfft2;
    fff(1,1:400) = 0;

%% Plotting graphs
    %% Acceleration (m/s^2) vs time (s)
figure(1)
    tiledlayout(3,1)
    ax1 = nexttile;

    plot(ax1,out.time,out.out_a, 'color','#A2142F')
    hold on
    grid on
    plot(ax1,out.time,out.acceleration, 'color','#0072BD')
    hold off

        legend('Output Acceleration', 'Reference Acceleration')
        title('Acceleration (m/s^2) vs time (s)')
        xlabel('time (s)')
        ylabel('Acceleration (m/s^2)')

    %% Acc(f) vs Freq (Hz)
    ax2 = nexttile;

    plot(ax2,xfft,abs(fff), 'color','#A2142F')
    grid on

        title('Fast fourier transform of the acceleration signal')
        xlabel('Frequency (Hz)')
        ylabel('Acc(f)')

    %% Tracking error (m) vs time (s)
    ax3 = nexttile;

    plot(ax3,out.time,out.fol_err, '-b')
    grid on

        title('Tracking error')
        xlabel('time (s)')
        ylabel('Tracking error (m)')
```