

**HIGH FIDELITY CONTROL AND SIMULATION OF A THREE  
DEGREES-OF-FREEDOM WAFER HANDLING ROBOT**

A Thesis  
Presented to  
The Academic Faculty

by

Elaina Babayan

In Partial Fulfillment  
of the Requirements for the Degree  
Master of Mechanical Engineering in the  
Woodruff School of Mechanical Engineering

Georgia Institute of Technology  
December 2014

**COPYRIGHT 2014 BY ELAINA BABAYAN**

**HIGH FIDELITY CONTROL AND SIMULATION OF A THREE  
DEGREES-OF-FREEDOM WAFER HANDLING ROBOT**

Approved by:

Dr. Harvey Lipkin, Advisor  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Dr. Chris Paredis  
School of Mechanical Engineering  
*Georgia Institute of Technology*

Mr. Robert Mitchell  
Varian Semiconductor Equipment  
*Applied Materials*

Date Approved: November 6<sup>th</sup>, 2014

For my grandfather, Joseph S. Griffo, in loving memory.

## ACKNOWLEDGEMENTS

I am grateful to the many people who have supported this work over the past four years. I wish to thank my thesis advisor, Dr. Harvey Lipkin, for his patience, support and detailed feedback, Dr. Chris Paredis for introducing me to Dymola and providing advice and guidance on model construction and troubleshooting, and Robert Mitchell for his technical guidance and mentorship. Many people at Applied Materials helped make this research possible, and I their appreciate the time and support, especially Robert Mitchell, Jim Buonodono, Scott Peitzsch and Gregory Thronson who have been strong advocates for the value of this work. Finally, I wish to thank my parents, my family, Slavik, and Sasha for their love, understanding, and support.

# TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS	iv
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF SYMBOLS	xiv
LIST OF ABBREVIATIONS	xix
SUMMARY	xx
<u>CHAPTER</u>	
1 Introduction	1
Background	1
Semiconductor Devices	1
Semiconductor Manufacturing Process	1
Wafer Fabrication	2
Front-End Manufacturing	3
Back-End Operation	4
Automation in Semiconductor Manufacturing	5
Three Degrees-of-Freedom Test Mechanism	11
Mechanical System	11
Motion Control Architecture	15
Dynamic Modeling of Robotic Mechanisms	16
Conclusions	18
2 Closed-Loop Model Generation with a Rigid-Body Plant	20
Rigid-Body Plant Model Generation	20

Mathematical Derivation	20
Denavit-Hartenberg Coordinate System Definition	20
Configuration Constraints	24
Y-Linear Axis Counterweight	24
Y-Rotate Axis Belt Drive	25
Kinematics	27
Dynamics	28
Conclusions	30
3 Software Modeling	31
Mechanism Dynamic Model Generation	31
Matlab Implementation	31
Dymola Implementation	31
Open-Loop Simulation	36
Controller Model Generation	38
PMAC Motion Controller	38
Copley Motor Controller	43
Motor Model Generation	48
Three-Phase Delta Wound Linear Motor	48
Three-Phase Wye Wound Rotary Motor	49
Conclusions	53
4 Simulated Rigid Connection Model and Experimental Results	54
Closed-Loop Dynamic Simulation	54
Closed-Loop Model Development	54
Mathematical Verification	57
Experimental Validation	58

Conclusions	65
5 Plant Model Generation with Flexible Connections	67
Modeling of Flexible Connections	67
Y-Linear Axis	67
Cables	67
Damping	70
Damping Models	70
Dymola Implementation	71
Structural Damping in Cables	71
Linear Bearings	74
X-Rotate Axis	80
Ferrofluid Seal	80
Drive Shaft	80
Cross-Roller Bearing	82
Viscous Damping of Ferrofluid	83
Y-Rotate Axis	85
Two-Stage Planetary Gearbox	85
Belt Drive	87
Lip Seal	90
Bearing Friction	91
Conclusions	92
6 Closed-Loop Flexible Connection Model and Experimental Results	94
Experimental Validation	94
Y-Linear Axis	94
X-Rotate Axis	100

Y-Rotate Axis	105
Simplified Flexible Connection Model	109
Conclusions	113
7 Conclusions	114
Model Limitations	114
Future Work	114
Y-Linear Axis	114
X-Rotate Axis	116
Y-Rotate Axis	116
Experimental Approach	116
Conclusions	117
APPENDIX A: Transistors	118
APPENDIX B: Symbolic Generation of Dynamic Equations in Matlab	120
APPENDIX C: Modelica Code for Cable in Tension Model	125
REFERENCES	126



## LIST OF TABLES

	Page
Table 2.1: Denavit-Hartenberg Parameters for Test Mechanism	23
Table 3.1: Relevant Modelica Connectors	35
Table 3.2: Relevant PMAC Motor Parameters	41
Table 4.1: Comparison of Rigid Connection Simulation and Measured Y-Linear Step Response	59
Table 4.2: Comparison of Rigid Connection Simulation and Measured X-Rotate Step Response	62
Table 4.3: Comparison of Rigid Connection Simulation and Measured Y-Rotate Step Response	64
Table 6.1: Comparison of Flexible Connection Simulation and Measured Y-Linear Step Response	97
Table 6.2: Performance Improvement from Flexible Connections in Y-Linear Step Response	99
Table 6.3: Comparison of Flexible Connection Simulation and Measured Y-Linear Step Response	102
Table 6.4: Performance Improvement from Flexible Connections in Y-Linear Step Response	104
Table 6.5: Comparison of Flexible Connection Simulation and Measured Y-Linear Step Response	107
Table 6.6: Performance Improvement from Flexible Connections in Y-Linear Step Response	108
Table 6.7: First Non-Rigid Frequency of Each Flexible Element	110
Table 6.8: Normalized Simulation Time for Rigid, Flexible, and Simplified Flexible Connection Models	111

## LIST OF FIGURES

	Page
Figure 1.1: Czochralski (CZ) crystal growing apparatus	2
Figure 1.2: Typical Front-End Wafer Processing Flowchart	3
Figure 1.3: FOUP used for Wafer Transport	6
Figure 1.4: OHT moves FOUPS between process steps	6
Figure 1.5: Two examples of Atmospheric Robot Architecture	7
Figure 1.6: EFEM Standardized Tool Interface	8
Figure 1.7: Two Examples of Vacuum Robot Architecture	9
Figure 1.8: Elevator Mechanism	10
Figure 1.9: Architecture of Three Degrees-of-Freedom Test Mechanism	12
Figure 1.10: Structure of the Y-Linear Axis	13
Figure 1.11: Structure of X-Rotate Axis	14
Figure 1.12: Structure of Y-Rotate Axis	15
Figure 1.13: Overview of Mechanism Control Architecture	16
Figure 2.1: Denavit-Hartenberg Parameters for a Generic Two-Link Configuration	21
Figure 2.2: Link Structure of Test Mechanism	22
Figure 2.3: Denavit-Hartenberg Coordinate Definitions for Test Mechanism	23
Figure 2.4: Simplified Y-Linear Drive Diagram	24
Figure 2.5: Simplified Y-Rotate Drive Diagram	26
Figure 3.1: Open-Loop Plant Dynamic Model	32
Figure 3.2: Modelica Standard Library Structure and Utilized Components	33
Figure 3.3: Across and Through Variables Defined for a Simple Rotational Dymola Model	34

Figure 3.4: Rigid-Body Model of Test Mechanism in Dymola	36
Figure 3.5: Rigid-Body Payload Sub-Model in Dymola	37
Figure 3.6: Dymola Open-Loop Plant Dynamic Model	38
Figure 3.7: Simplified Structure of PMAC Control Loop	40
Figure 3.8: Simulink Model of the PMAC Motion Controller	42
Figure 3.9: Simulink Model of the Integration Mode Selector	43
Figure 3.10: Simulink Model of Copley Motor Controller	45
Figure 3.11: Simulink Model of the Copley Phase Current Control Loop	46
Figure 3.12: Simulink Model of Copley Current Limiter	47
Figure 3.13: Circuit Diagram of Three-Phase Delta Wound Brushless DC Motor	48
Figure 3.14: Three-Phase Delta Wound Brushless Linear Motor Model in Dymola	50
Figure 3.15: Circuit Diagram for Three-Phase Wye Wound Brushless DC Motor	51
Figure 3.16: Three-Phase Wye Wound Brushless Rotary Motor Model in Dymola	52
Figure 4.1: Closed-Loop Dynamic Simulink Model with Dymola Plant	55
Figure 4.2: Input Block Generates Position Command in Simulink	56
Figure 4.3: Closed-Loop Verification of Rigid Connection Plant	57
Figure 4.4: Simulated and Measured Y-Linear Step Response	58
Figure 4.5: Initial Peak on Measured Y-Linear Response Crosses Command	60
Figure 4.6: Simulated and Measured Step Response Current on Y-Linear Axis	61
Figure 4.7: Simulated and Measured X-Rotate Step Response	62
Figure 4.8: Simulated and Measured Current for X-Rotate Step Response	63
Figure 4.9: Simulated and Measured Y-Rotate Step Response	63
Figure 4.10: Simulated and Measured Current for Y-Rotate Step Response	65

Figure 5.1: Cross-Section of a 7x19 Stranded Cable	69
Figure 5.2: Diagram of Simple System for Calculation of Cable Natural Frequency	72
Figure 5.3: Sample Axial Force-Deflection Diagram for a Rotary Bearing	74
Figure 5.4: Static Bearing Loading on Y-Linear Axis	76
Figure 5.5: Simplified Bearing Y-Linear Stiffness Model	77
Figure 5.6: Dymola Model of Y-Linear Axis with Flexible Connections and Damping	79
Figure 5.7: Typical Ferrofluid Seal Design	80
Figure 5.8: Static Loading on X-Rotate Cross-Roller Bearing	82
Figure 5.9: Dymola Model of X-Rotate Axis with Flexible Connections and Damping	84
Figure 5.10: Two-Stage Planetary Gearbox	86
Figure 5.11: Simplified Belt Drive Model for Natural Frequency Calculation	89
Figure 5.12: Typical Lip Seal Cross-Section	90
Figure 5.13: Y-Rotate Bearing Static Loading	91
Figure 5.14: Dymola Model of Y-Rotate Axis with Flexible Connections and Damping	93
Figure 6.1: Simulated and Measured Y-Linear Step Response with Calculated Damping Parameters	95
Figure 6.2: Simulated and Measured Y-Linear Step Current Response with Calculated Damping Parameters	95
Figure 6.3: Simulated and Measured Y-Linear Step Response with Improved Damping Parameters	97
Figure 6.4: Initial Peak in the Simulated Y-Linear Step Response Exceeds the Commanded Position	98
Figure 6.5: Error Between Rigid and Flexible Connection Y-Linear Models and the Measured Response	99
Figure 6.6: Simulated and Measured Y-Linear Step Response Current with Improved Damping Parameters	100

Figure 6.7: Simulated and Measured X-Rotate Step Response with Calculated Damping Parameters	101
Figure 6.8: Simulated and Measured X-Rotate Step Response Current with Calculated Damping Parameters	101
Figure 6.9: Simulated and Measured X-Rotate Step Response with Improved Damping Parameters	102
Figure 6.10: Simulated and Measured X-Rotate Step Response Current with Improved Damping Parameters	103
Figure 6.11: Error Between Rigid and Flexible Connection X-Rotate Models and the Measured Response	104
Figure 6.12: Simulated and Measured Y-Rotate Step Response with Calculated Damping Parameters	105
Figure 6.13: Simulated and Measured Y-Rotate Step Response Current with Calculated Damping Parameters	106
Figure 6.14: Simulated and Measured Y-Rotate Step Response with Improved Damping Parameters	106
Figure 6.15: Simulated and Measured X-Rotate Step Response Current with Improved Damping Parameters	107
Figure 6.16: Error Between Rigid and Flexible Connection Y-Rotate Models and the Measured Response	109
Figure 6.17: Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured Y-Linear Step Response	111
Figure 6.18: Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured X-Rotate Step Response	112
Figure 6.19: Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured Y-Rotate Step Response	112
Figure A.1: Simplified Cross-Section of BJT	118
Figure A.2: Simplified Cross-Section of MOSFET	119
Figure C.1: Modelica Code for CableInTension Component	125

## LIST OF SYMBOLS

$i$	Axis index
$a_i$	Denavit-Hartenberg parameter specifying the distance between consecutive $z$ -axes
$d_i$	Denavit-Hartenberg parameter specifying the distance between consecutive $x$ -axes
$\alpha_i$	Denavit-Hartenberg parameter specifying the angle between consecutive $z$ -axes
$\theta_i$	Denavit-Hartenberg parameter specifying the angle between consecutive $x$ -axes
$d_{cw}$	Y-Linear counterweight vertical position relative to the base frame
$c_3$	Vertical distance between the Y-Linear pulley center and base frame
$c_1$	Vertical distance between frame 1 and the Y-Linear payload carriage cable mount
$c_2$	Vertical distance between the Y-Linear counterweight CG and the counterweight cable mount
$L_{cable}$	Total Y-Linear cable length
$r_p$	Y-Linear pulley radius
$v_{cw}$	Velocity of Y-Linear counterweight
$\theta_M$	Position of Y-Rotate motor output
$\theta_G$	Position of Y-Rotate gearbox output
$R_G$	Y-Rotate gear ratio
$r_1$	Y-Rotate drive pulley radius
$r_2$	Y-Rotate driven pulley radius
$\omega_i$	Angular velocity of frame $i$
$v_i$	Linear velocity of frame $i$
$v_{Gi}$	Linear velocity of the CG of link $i$

$r_{Gi/i}$	Position of the CG of link $i$ relative to frame $i$
$T_i$	Kinetic energy of link $i$
$V_i$	Potential energy of link $i$
$m_i$	Mass of link $i$
$I_{i/Gi}$	Inertia of link $i$ about the CG of link $i$
$g$	Gravity vector
$\mathcal{L}$	Lagrangian
$q_j$	Generalized system coordinates
$Q_j$	Generalized forces
$\delta W$	Virtual work
$m$	Degree-of-freedom index
$p$	Rotational degree-of-freedom index
$k$	Translational degree-of-freedom index
$F_{1M}$	Y-Linear motor force
$\tau_{2M}$	X-Rotate motor torque
$\tau_{3M}$	Y-Rotate motor torque
$\tau_{3P}$	Y-Rotate motor torque applied at the driven pulley
$\zeta$	Motor phase number
$n$	Number of poles in rotary motor
$\rho$	Linear motor electrical cycle length
$i_{peak}$	Peak current limit
$i_{cont}$	Continuous current limit
$t_{I^2T}$	I <sup>2</sup> T time limit
$i_{actual}$	Actual motor current
$\Delta t$	Motor controller time step

$R_\zeta$	Motor phase $\zeta$ resistance
$L_\zeta$	Motor phase $\zeta$ inductance
$k_{emf,\zeta}$	Motor phase $\zeta$ back-emf
$V_\zeta$	Motor phase $\zeta$ voltage
$i_\zeta$	Motor phase $\zeta$ current
$e_t$	Error between simulated and measured response at time t
$n_s$	Total number of time samples
$k_{eff}$	Effective axial stiffness of Y-Linear cable
$E$	Young's Modulus of Y-Linear Cable
$A$	Effective cross-sectional of Y-Linear cable
$b$	Viscous damping coefficient
$\eta$	Loss factor
$b_s$	Equivalent viscous damping coefficient for structural damping model
$\mu$	Coefficient of friction
$N$	Normal force
$\omega_n$	$n^{\text{th}}$ natural frequency
$\varphi_n$	Mode shape corresponding to $n^{\text{th}}$ natural frequency
$F_r$	Y-linear static bearing loading in the radial direction
$F_t$	Y-linear static bearing loading in the transverse direction
$p_x$	Distance between Y-Linear payload CG and bearing center in $x$ -direction
$p_z$	Distance between Y-Linear payload CG and bearing center in $z$ -direction
$b_y$	Vertical distance between Y-Linear bearing blocks
$k_{rad}$	Y-Linear bearing stiffness in radial loading direction
$k_{rev}$	Y-Linear bearing stiffness in reverse loading direction
$k_t$	Y-Linear bearing stiffness in transverse loading direction



$\kappa_{Mx}$	Y-Linear equivalent bearing rotational stiffness about the $x$ -axis
$\kappa_{My}$	Y-Linear equivalent bearing rotational stiffness about the $y$ -axis
$\kappa_{Mz}$	Y-Linear equivalent bearing rotational stiffness about the $z$ -axis
$\kappa_s$	Torsional stiffness of X-Rotate drive shaft
$G$	Modulus of rigidity of X-Rotate drive shaft
$J$	Second moment of area of X-Rotate drive shaft
$d_o$	Outer diameter of X-Rotate drive shaft
$d_i$	Inner diameter of X-Rotate drive shaft
$k_{sb}$	Bending stiffness of X-Rotate drive shaft
$\omega_{torsion}$	First torsional frequency of X-Rotate drive shaft
$\omega_{bending}$	First bending frequency of X-Rotate drive shaft
$c_{sp}$	Specific stiffness of Y-Rotate belt
$w$	Width of Y-Rotate belt
$k_a$	Axial stiffness of Y-Rotate belt
$k_m$	Mesh stiffness of Y-Rotate belt
$z_m$	Number of teeth in mesh
$z_{mv}$	Virtual mesh stiffness
$k_{tooth}$	Tooth stiffness of Y-Rotate belt
$J_1$	Inertia of the Y-Rotate motor and gearbox about the $y$ -axis
$J_2$	Inertia of the Y-Rotate payload about the $y$ -axis
$r_s$	Radius of Y-Rotate bearing shaft
$R_{A,X}$	Static radial loading on Y-Rotate duplex bearing pair in $x$ -direction
$R_{A,Z}$	Static radial loading on Y-Rotate duplex bearing pair in $z$ -direction
$R_{B,X}$	Static radial loading on Y-Rotate needle bearing in $x$ -direction
$R_{B,Z}$	Static radial loading on Y-Rotate needle bearing in $z$ -direction

$T_{belt}$	Y-Rotate belt pretension
$\beta$	Angle of Y-Rotate belt tension force relative to the pulley tangent vector
$x_{cg}$	Offset of Y-Rotate CG from bearing axis in the $x$ -direction
$z_{cg}$	Offset of Y-Rotate CG from bearing axis in the $z$ -direction
$L_T$	Distance from the center of the Y-Rotate needle bearing to the center of the belt
$L_A$	Distance from the center of the Y-Rotate duplex bearing pair to the payload CG
$L_B$	Distance from the center of the Y-Rotate needle bearing to the payload CG
$\Delta r$	Percent difference between simulation and measurement for rigid connection model
$\Delta f$	Percent difference between simulation and measurement for flexible connection model

## LIST OF ABBREVIATIONS

IC	Integrated Circuit
CZ	Czochralski
EPI	Epitaxy
CVD	Chemical Vapor Deposition
PVD	Physical Vapor Deposition
ALD	Atomic Layer Deposition
CMP	Chemical Mechanical Polishing
FOUP	Front-Opening Universal Pod
OHT	Overhead Hoist Transfer
OHS	Overhead Shuttle
EFEM	Equipment Front End Module
SCARA	Selective Compliance Arm for Robot Manufacturing
PMAC	Programmable Multi-Axis Controller
FMI	Functional Mock-Up Interface
FMU	Functional Mock-Up Unit
PID	Proportional-Integral-Derivative
PD	Proportional-Derivative
PI	Proportional-Integral
DAC	Digital to Analog Converter
PWM	Pulse Width Modulation
EMF	Electromotive Force
DOF	Degree-of-Freedom

BLDC	Brushless DC
RMS	Root Mean Square
BJT	Bipolar Junction Transistor
FET	Field Effect Transistor
MOSFET	Metal-Oxide-Semiconductor FET

## SUMMARY

Wafer handling robotics are critical in semiconductor manufacturing to enable tight control of temperature, humidity, and particle contamination during processing. Closed-loop dynamic modeling during the robot design process ensures designs meet throughput and stability specifications prior to prototype hardware purchase. Dynamic models are also used in model-based control to improve performance. This thesis describes the generation and mathematical verification of a dynamic model for a three degrees-of-freedom wafer handling mechanism with one linear and two rotary axes. The dynamic plant model is integrated with motion and motor controller models, and the closed-loop performance is compared with experimental data. Models with rigid and flexible connections are compared, and the flexible connection models are shown to overall agree better with a measured step response. The simulation time increase from the addition of flexible connections can be minimized by modeling only the component stiffnesses that impact the closed-loop mechanism response. A method for selecting which elements to include based on controller bandwidth is presented and shown to significantly improve simulation times with minimal impact on model predictive performance.

# CHAPTER 1

## INTRODUCTION

This chapter provides an overview of semiconductor devices and the semiconductor manufacturing process, substrate handling, and a variety of substrate handling mechanisms. Robot dynamic modeling in the literature is reviewed, and the three degrees-of-freedom test mechanism modeled in this study is introduced.

### Background

#### Semiconductor Devices

Semiconductor devices are the basis of modern electronics and range in complexity from discrete diodes and transistors to complex arrays making up integrated circuits (ICs). Microprocessor complexity is typically characterized by the number of transistors as this is strongly linked to processing speed and memory capacity. Following Moore's Law the number of transistors approximately doubles every 24 months. Decreases in component size are characterized by the minimum line width, typically defined as the smallest lateral feature size printed on the wafer during fabrication <sup>[27]</sup>. Currently, leading microprocessors have line widths down to 20nm and contain over one billion transistors. An overview of two common transistor types is provided in Appendix A.

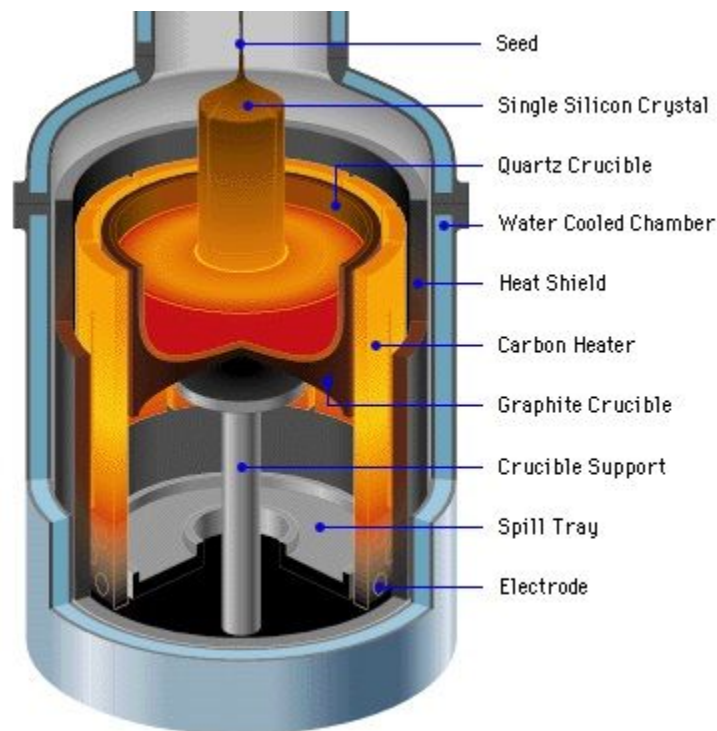
#### Semiconductor Manufacturing Process

The semiconductor manufacturing process is comprised of three stages. The silicon substrate is created during wafer fabrication. Then the IC circuitry is fabricated in

the front-end manufacturing phase. Finally, the circuits are tested and separated as individual chips during back-end manufacturing.

### Wafer Fabrication

The silicon substrate is the foundation of most semiconductor devices and must be free of impurities or flaws that are detrimental to device performance. Wafer production starts with a purified silicon ‘seed’ crystal that is dipped into a pool of molten silicon and slowly pulled upwards. The surface tension draws a small amount of molten silicon up with the seed, and upon cooling forms a single-crystal silicon ingot which is approximately 300mm in diameter, the wafer size for current state-of-the-art processes (Figure 1.1).



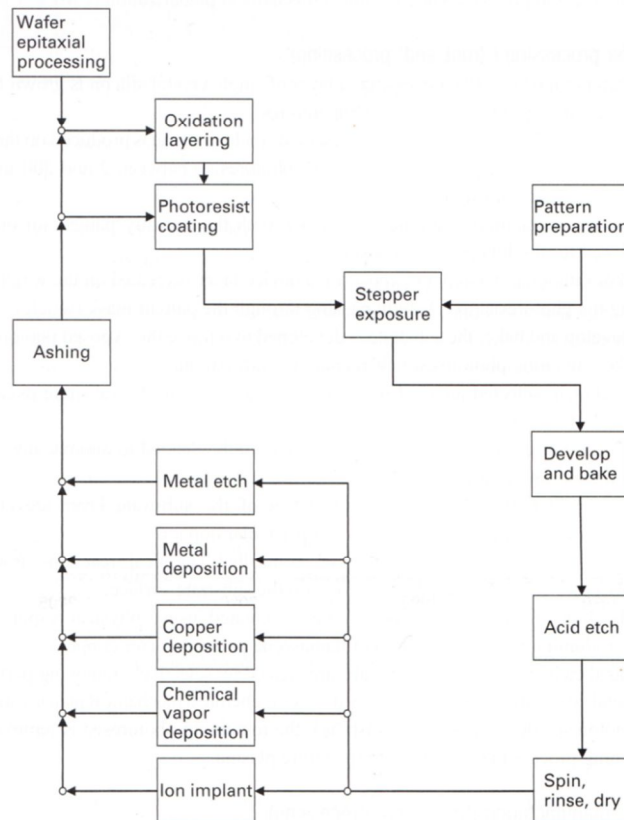
**Figure 1.1: Czochralski (CZ) crystal growing apparatus. Silicon is heated in the quartz crucible and then a seed is slowly pulled upward creating a single-crystal silicon ingot. Image from “Two growth techniques for mono-crystalline silicon: Czochralski vs. Float Zone”<sup>[22]</sup>**

The cylindrical ingot is then sliced into wafers which are ground smooth and chemically polished to a mirror-like finish.

### Front-End Manufacturing

The front-end manufacturing stage is the most complex of the three phases. It includes all the wafer processing steps required to fabricate the IC circuitry. The total number and order of process steps in device fabrication is dependent on the particular technology and device manufacturer; however, the basic process steps are the same.

Figure 1.2 shows a flowchart of the typical process steps <sup>[21]</sup>.



**Figure 1.2: Typical front-end wafer processing flowchart. Specific number and order of process steps vary depending on device. Process steps are repeated to generate multiple layers in the IC. Image from “Robotics for Electronics Manufacturing Principles and Applications in Cleanroom Automation” <sup>[21]</sup>.**



The first step is typically Epitaxy (EPI), in which a layer of single crystal silicon is grown on the surface of the wafer. Then an insulating silicon dioxide layer is grown or deposited through thermal oxidation or chemical vapor deposition (CVD).

Next a light-sensitive photoresist is applied across the surface of the wafer. A pattern mask is then used to cover specified areas. Photolithography is used to transfer the pattern to the wafer by exposing the uncovered areas to ultraviolet light. For a negative photoresist, the light hardens the exposed areas. The material below the unexposed photoresist is etched away. The wafer is then repeatedly cleaned to remove any surface contaminants. Ion implantation or diffusion may then be used to modify the electrical conductivity of the exposed silicon substrate through the controlled addition of impurities. Chemical vapor deposition (CVD) can also be used to create device layers on the wafer surface. A metal layer can be deposited through physical vapor deposition (PVD) and selectively etched away using a patterned photoresist to create conductive circuit paths within the IC. Alternatively, damascene patterning may be used to create Copper connections. If necessary, the wafer is planarized using chemical mechanical polishing (CMP). The remaining photoresist is then removed using a plasma ashing process. The above steps are repeated until the device is complete <sup>[2]</sup>.

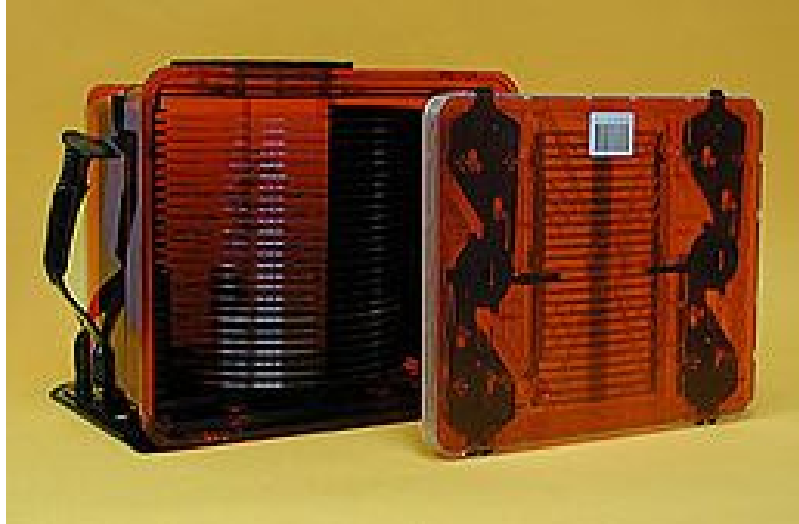
### Back-End Operation

Testing is performed to ensure that the ICs function as expected. Then the wafers are cut into individual ICs in a step known as die preparation. The die is then bonded to a lead die frame. Wire bonding is performed to connect electrical leads on the die frame to the input/output terminals of the chip. The entire package is encapsulated in plastic to provide physical and chemical protection <sup>[2]</sup>.

## **Automation in Semiconductor Manufacturing**

A clean environment is required for the manufacturing of semiconductor devices. Temperature, humidity and particle contamination must be tightly controlled. As device size decreases, these requirements become more stringent because smaller defects can degrade device performance. Humans naturally generate heat, moisture and particles, so maintaining an ultra-clean environment with human operators is difficult. Automated materials handling has been a key improvement in environmental control since the transition to the 300mm wafer size in the mid-1990's. The increase in wafer size also drove an increase in carrier mass, so ergonomics became a concern for operators transporting wafers manually between process steps<sup>[21]</sup>.

Automation in semiconductor fabrication facilities can be characterized into three levels: interbay automation (between bays of process tools), intrabay automation (within a single process bay) and tool-level automation (within a single process tool)<sup>[21]</sup>. Wafers are transported within and between process bays in specialized plastic enclosures called Front Opening Universal Pods (FOUPs) (Figure 1.3). The FOUP provides a controlled environment during wafer transport. A FOUP can hold up to 25 wafers which rest on plastic fins spaced to accommodate a robot end effector during loading and unloading at each process step. FOUPs are moved both between and within process bays using overhead transport vehicles which move rapidly along an overhead track.



**Figure 1.3: FOUP used for wafer transport. Front of FOUP is open to show 25 wafers on plastic fins inside. Photo Courtesy of Entegris, Inc.**

The overhead hoist transfer (OHT) rides below the track and uses a belt-driven hoist mechanism to raise and lower the FOUP (Figure 1.4).



**Figure 1.4: OHT moves FOUPS between process steps. *Left:* Multiple OHT vehicles move along an overhead track between process steps. *Center:* A FOUP is loaded onto an OHT vehicle. *Right:* A FOUP is lowered from an OHT vehicle using a belt hoist system. Photos Courtesy of Daifuku Co, Ltd.**

The OHT is the most common method of wafer transport in modern 300mm fabrication facilities. Within a process bay and to and from storage facilities such as stockers, the FOUP may also be moved using conveyor systems or an overhead shuttle system (OHS). In an OHS the FOUP rides on a carriage above the track.

At the tool-level, robots are used to transport wafers from the FOUP into the process environment and back. Atmospheric robots remove the wafers from the FOUP and move them into the tool. In tools that operate at atmospheric pressure, such as CMP, inspection and metrology tools, the atmospheric robot can pass the wafer directly into the process environment. However, deposition (ALD, PVD, CVD, EPI), etch, and ion implant tools require a high vacuum processing environment. In these cases, the atmospheric robot places the wafers in a load-lock where it is transferred through gate valves from the atmosphere into the vacuum environment. Once in the vacuum environment wafers are handled by the vacuum robot.

Two examples of atmospheric robot architecture are shown in Figure 1.5.

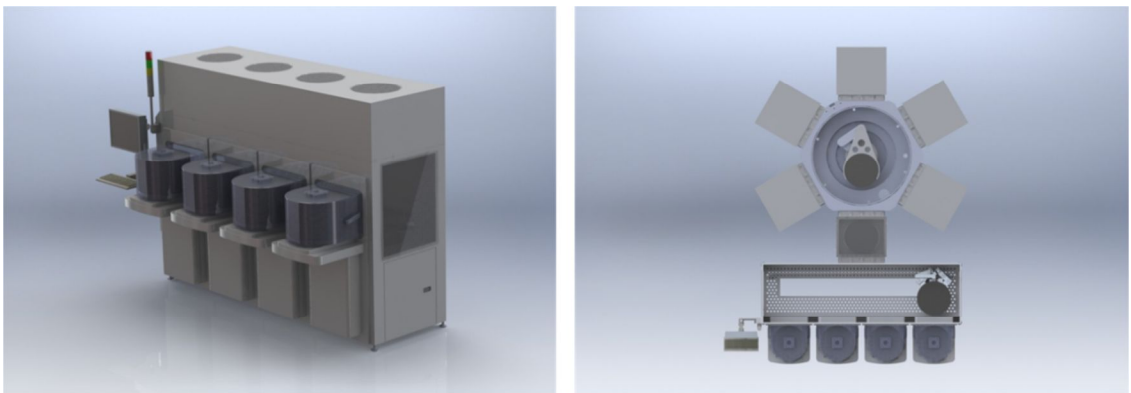


**Figure 1.5: Two examples of atmospheric robot architecture. *Left:* Single end-effector atmospheric robot comprised of 2 rotary axes (a rotation at the base and a belt-driven extension) and a vertical linear axis requires a horizontal track to feed multiple load ports. *Right:* Dual end-effector atmospheric robot with 8 motion axes can feed four load ports without a track. Robot has a vertical linear axis at the base, a rotary axis at the base, a belt-driven extend axis, a rotary axis to allow rotation of the third link, and rotary and linear axes on each end effector. Images Courtesy of Genmark Automation, Inc.**

Atmospheric robots range in complexity, but typically have five degrees-of-freedom. A vertical linear axis enables the robot to lift wafers from the FOUP. Three vertical rotary axes allow the robot end effector to transport wafers in a horizontal plane from a single

FOUP. Most robots have an additional fourth rotary axis or ride on a horizontal track to allow them to access two to four FOUPs. The end effector of an atmospheric robot is designed to fit into the 10mm vertical pitch between wafers in the FOUP. Atmospheric end effectors may include specially designed wafer edge-gripping pads or vacuum suction mechanisms to prevent wafer slip during transport.

An atmospheric robot may be part of an equipment front end module (EFEM), a standardized interface used by some tools to minimize contamination when moving the wafer from the FOUP into the process environment. Many EFEMs have two load ports, but additional load ports may be used on certain processing tools where higher throughputs are required. Figure 1.6 shows an EFEM with 4 load ports used to feed wafers to a vacuum cluster system.



**Figure 1.6: EFEM standardized tool interface. *Left:* 4 load port EFEM used to minimize contamination when moving the wafers into the process environment. *Right:* Top view of 4 load port EFEM showing atmospheric robot on horizontal track, load lock, and vacuum robot used to feed 5 process stations. Images Courtesy of Genmark Automation, Inc.**

Vacuum robots are designed to withstand high-vacuum pressures (between  $10^{-6}$  and  $10^{-8}$ Torr) and harsh environments, including aggressive chemicals and plasmas.

Figure 1.7 shows two vacuum robot architecture configurations. Vacuum robots vary in complexity, but the most common architecture differs slightly from the standard SCARA

(“selective compliance arm for robot assembly”) robot arm because the vertical linear axis is at the base of the arm rather than the tip <sup>[21]</sup>. Having the vertical axis at the base allows the end effector to fit through small openings (such as through gate valves into a load-lock), and it also improves cleanliness since the particles generated by the motion of the linear axis are farther from the wafer. Vacuum robots transport the wafer from the load-lock into the process environment.



**Figure 1.7: Two examples of vacuum robot architecture. *Left:* Vacuum robot with base rotate axis and belt-driven extend axis. *Right:* Dual end-effector robot with both arms retracted. Robot consists of 6 motion axes: rotary and linear axes at the base, a belt-driven rotary extension axis for each arm set, and a linear axis for each end effector. Photos courtesy of Genmark Automation, Inc.**

Depending on the number of wafers processed during a load-lock cycle, the load-lock may include an elevator mechanism to drive wafers vertically, reducing the required stroke or entirely eliminating the requirement for a linear axis on the vacuum robot (Figure 1.8). Once in the process environment the wafer is handed off to a processing station or a process-specific mechanism.



**Figure 1.8: Elevator mechanism.** Elevator mechanism used in load-lock to drive wafers vertically, reducing or eliminating the requirement for vertical motion in the vacuum robot. Photo courtesy of Genmark Automation, Inc.

### **Three Degrees-of-Freedom Test Mechanism**

Aggressive design schedules make hardware iteration in the design of semiconductor robotics impractical. Dynamic modeling in the design process enables the prediction of robot performance and minimizes hardware iteration. The diverse architectures and complexities of wafer handling mechanisms necessitate a flexible modeling platform where validated components and subsystems can be re-used to minimize modeling time. This thesis presents a methodology for modeling semiconductor mechanisms and develops a model of an existing process mechanism to demonstrate model performance.

For ion implant, machine throughput is critical. Depending on the specific recipe, implanters can process up to 500 wafers per hour. This requires aggressive motion

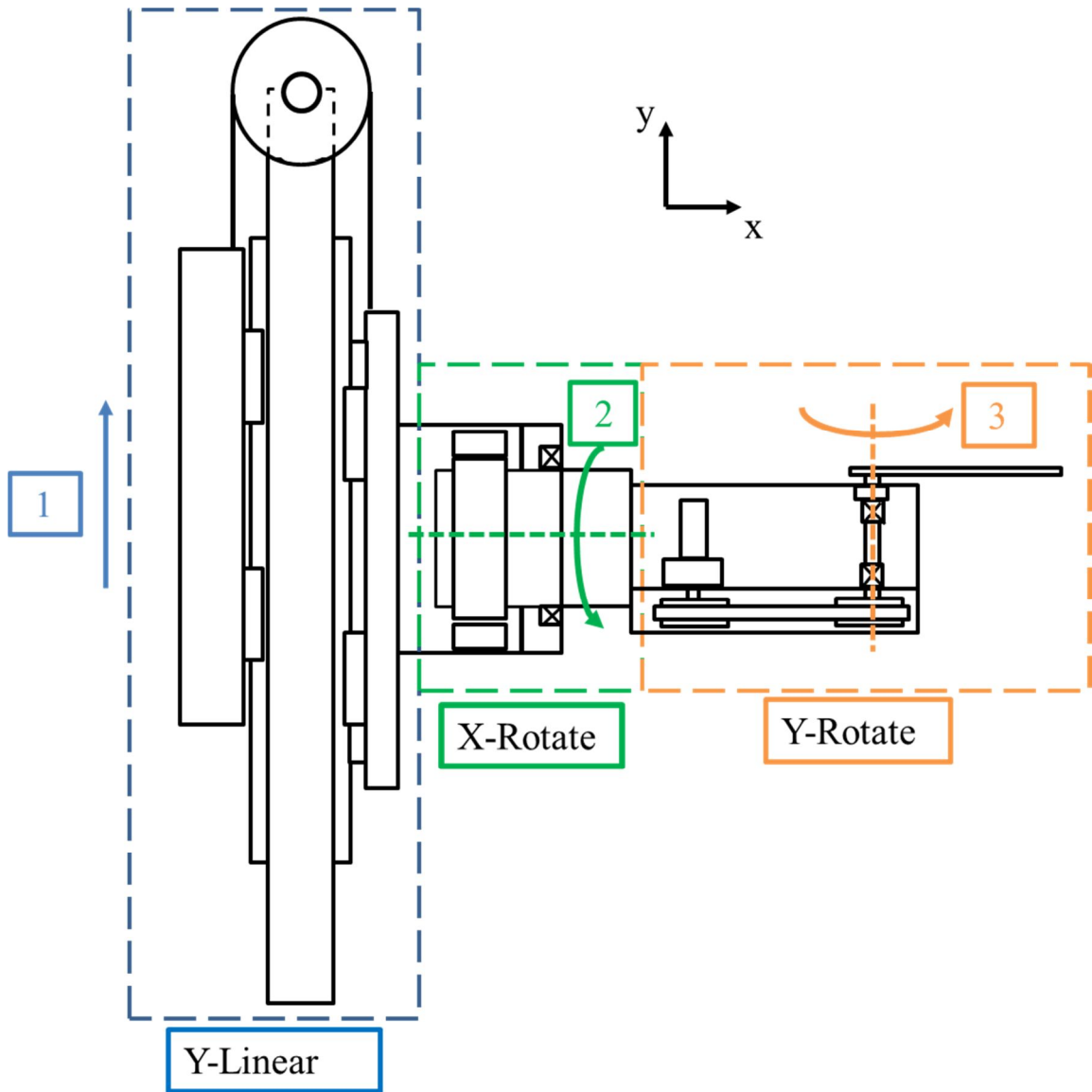
profiles and minimal settling time between moves. The mechanism studied receives wafers from the vacuum robots and passes them through the process environment (in vacuum). The goals of modeling are to predict positional accuracy (following error), closed-loop system stability and settling time.

## **Mechanical System**

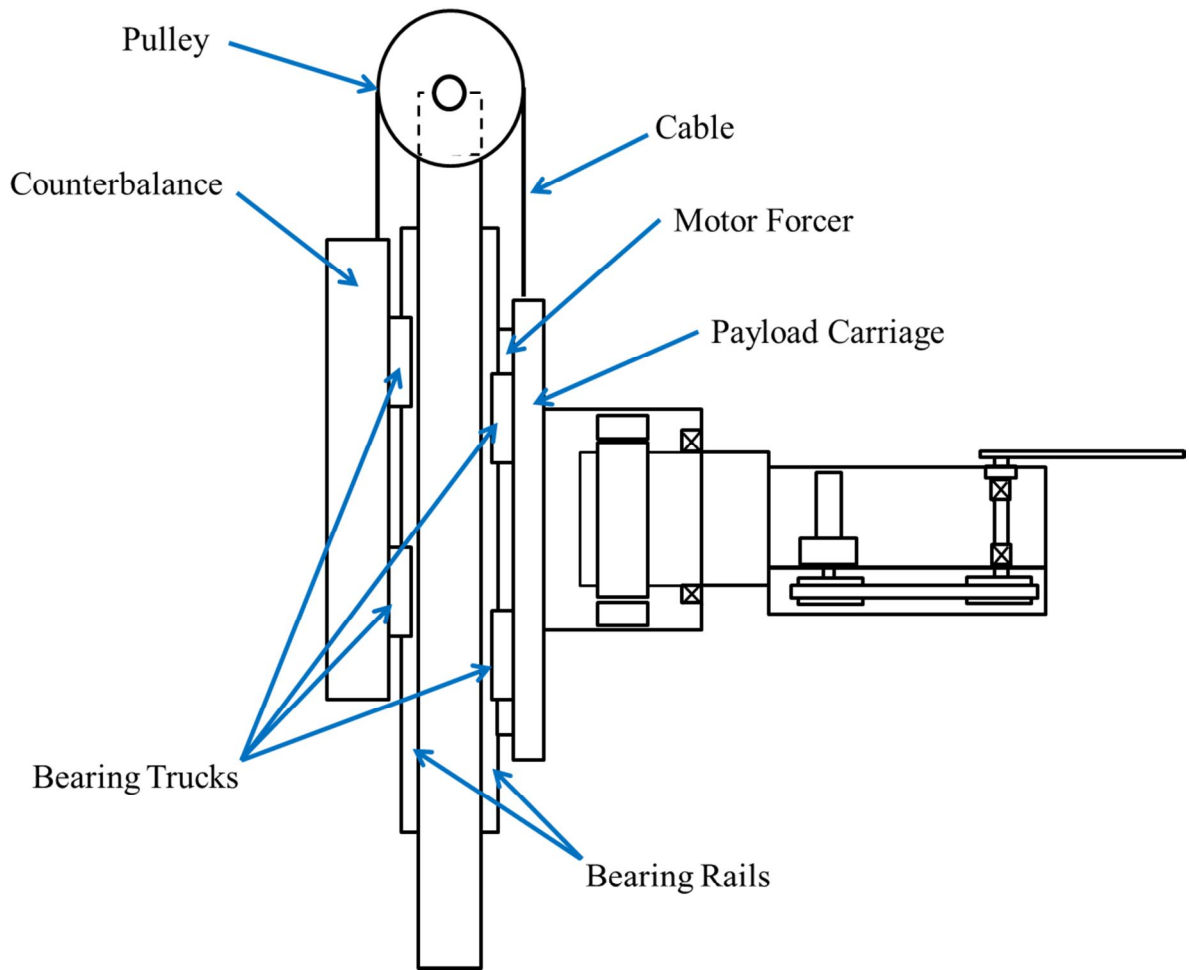
The test mechanism described in this study is depicted in Figure 1.9. The test mechanism moves wafers vertically, relative to gravity, in the wafer process environment. To facilitate the exchange of wafers from the vacuum robot, the mechanism end effector is oriented horizontally. Once the exchange is complete, the end effector with the wafer is rotated 90 degrees about the  $x$ -axis into the vertical orientation, and the wafer is passed through the process environment.

The three degrees-of-freedom test mechanism consists of one linear and two rotary axes. The vertically-oriented linear axis (Linear Y) is driven by a linear motor mounted directly to the payload carriage. In Figure 1.10, the payload is counterbalanced with two cables over a pair of pulleys which prevents payload damage in the case of a single cable failure. Both the payload and counterbalance ride on a pair of linear bearing rails, each with two bearing blocks per rail which provides increased moment stiffness. Feedback is provided by a high-resolution linear encoder mounted to the payload carriage.



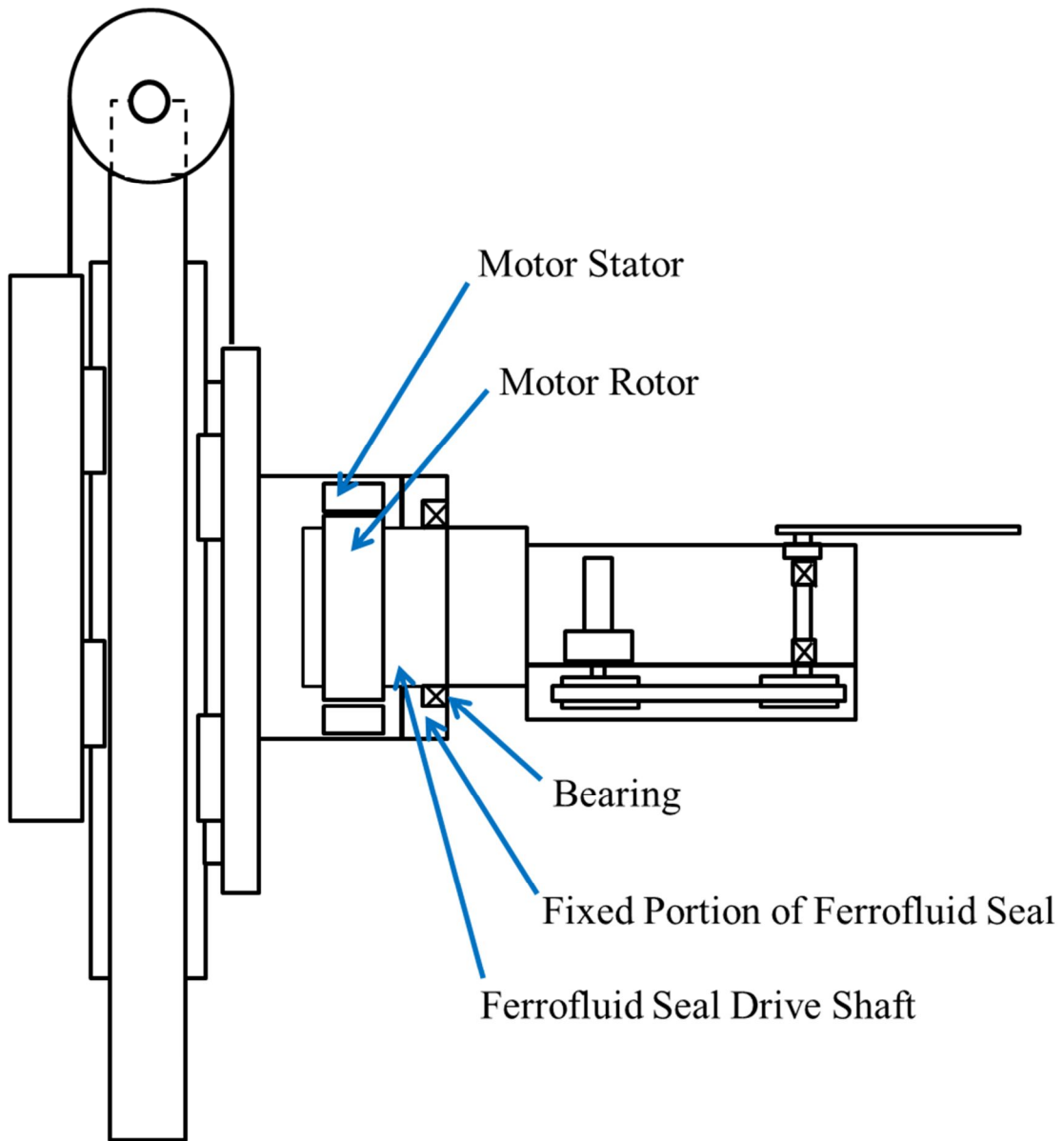


**Figure 1.9: Architecture of Three Degrees-of-Freedom Test Mechanism.** The test mechanism is comprised of a vertical linear axis (Y-Linear), a rotary axis about  $x$  (X-Rotate), and a rotary axis about  $y$  (Y-Rotate).



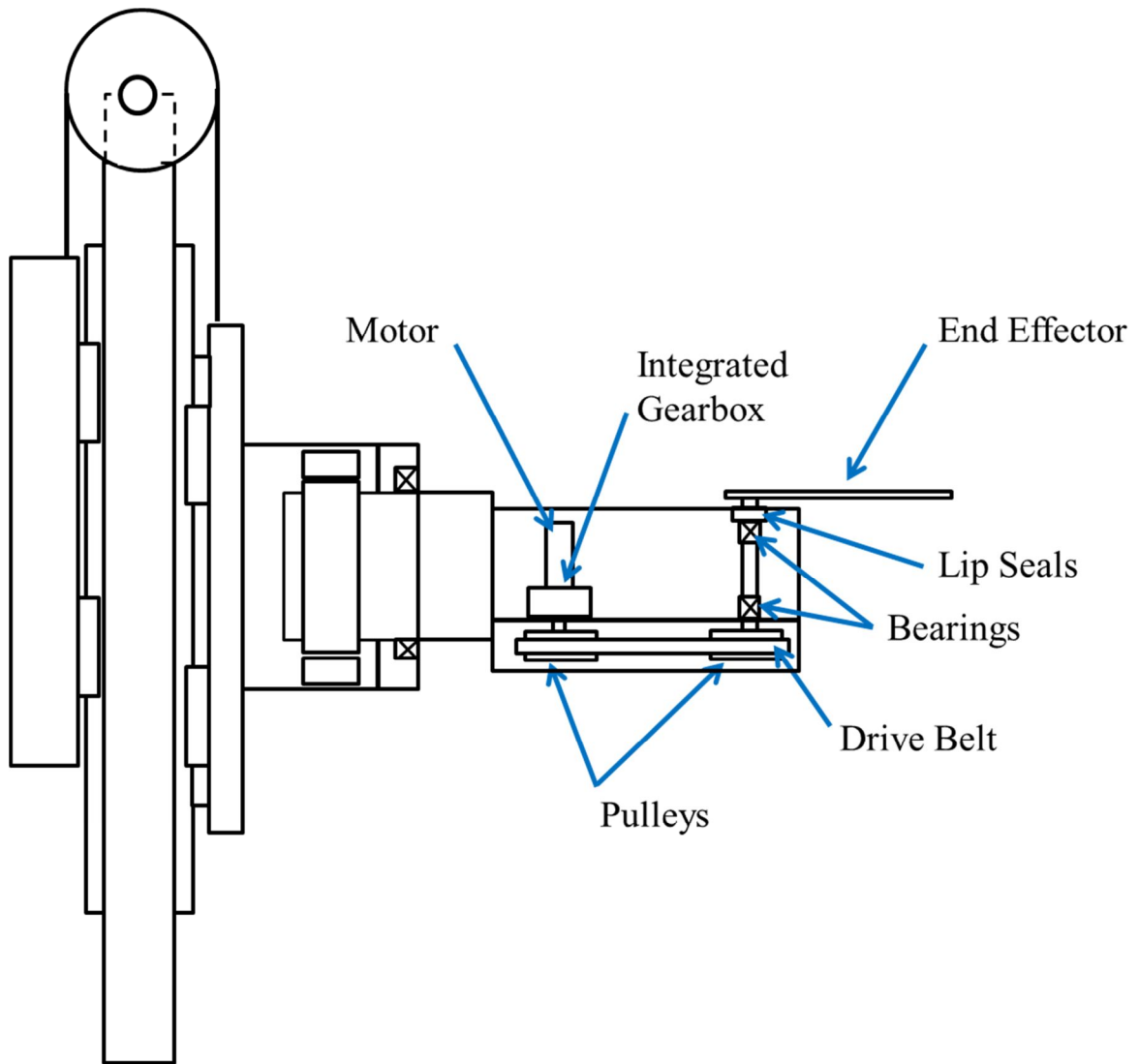
**Figure 1.10: Structure of the Y-Linear Axis. The payload is mounted to a pair of linear bearing rails with two trucks per rail. It is counterbalanced using a pair of cables over two pulleys. The counterweight also rides on a pair of bearing rails.**

The first rotary axis (X Rotate) in Figure 1.11 uses a direct-drive rotary motor about the  $x$ -axis through a ferrofluidic seal which allows the motor to remain at atmospheric pressure while the test mechanism functions in a vacuum environment. The payload of this axis is supported with a single cross-roller bearing. Feedback is provided by a high-resolution rotary encoder mounted to the motor rotor.



**Figure 1.11: Structure of the X-Rotate Axis. A rotary motor is supported by a cross-roller bearing and drives the payload through a ferrofluidic seal.**

In the second rotary axis (Y-Rotate), a rotary motor with an integrated two-stage planetary gearbox drives the mechanism payload through a timing belt. This mechanism provides rotation about the y-axis. Encoders on both the payload and the motor provide dual feedback (Figure 1.12).

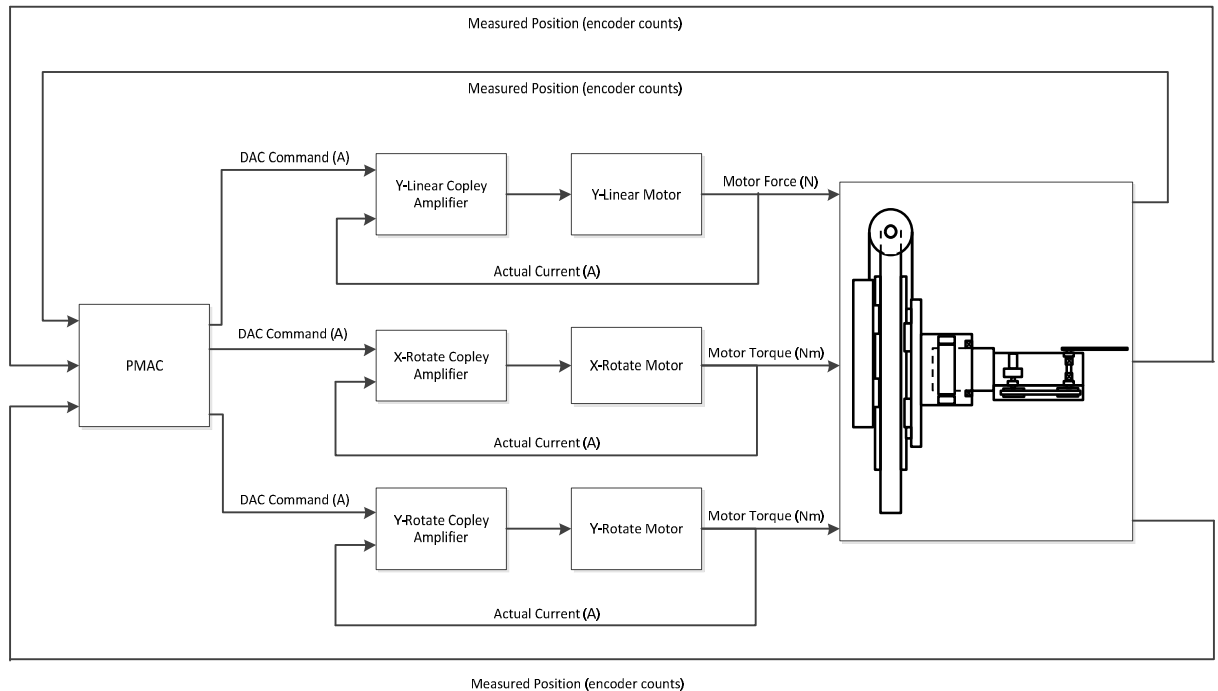


**Figure 1.12: Structure of the Y-Rotate Axis.** A rotary motor with an integrated two-stage planetary gearbox drives a pulley which is connected to the payload through a belt. The drive shafts pass to the vacuum environment through lip seals, allowing the motor to remain at atmospheric pressure.

### **Motion Control Architecture**

Motion control for all three axes of the test mechanism is performed using the Delta Tau Turbo PMAC2 Ultralite programmable multi-axis controller<sup>[33]</sup> (Figure 1.13). Motion profiles are generated by the PMAC based on user inputs, and position and velocity feedback from each mechanism is used to close the PID-based control loop. Each motion axis has a Copley Xenus MACRO amplifier to provide PI control of the

motor current <sup>[5]</sup>. The Copley amplifiers receive a command current signal from the PMAC and perform space-vector pulse width modulation (SVPWM) to generate the desired motor currents.



**Figure 1.13: Overview of Mechanism Control Architecture. The PMAC motion controller generates motion profiles for all three axes and outputs commands based on the positional error. Each axis has a separate Copley amplifier which closes the motor current control loop.**

### Dynamic Modeling of Robotic Mechanisms

The mathematical derivations for the kinematic and dynamic analyses of a rigid-body, serial mechanism with six or fewer degrees-of-freedom are well-known [6],[16],[21],[37]. Recent robotics literature tends to focus on mechanisms with more complex dynamics such as parallel mechanisms, walking and hopping, compliance in the linkages or joints, and mechanisms that exhibit nonlinear behavior. Some models include details of the control or actuation scheme. Applications for these models include prediction of

dynamic behavior during the design process and model-based control. The model development is primarily mathematical; however, some other methods such as finite-element analysis software and graphical lumped-parameter modeling tools are also used.

Li et al. <sup>[19]</sup> describe the development of a model for a 4-DOF parallel SCARA robot. The kinematics and rigid-body dynamics of the mechanism are derived mathematically, and an elastic dynamic model is created by integrating a parameterized CAD model of the robot geometry with elastic joints (using a spring and damper in parallel). Kinematic constraints are determined from the kinematic analysis, and dynamic performance indices are generated from the rigid-body dynamic equations. These are combined to formulate an optimization problem which is solved using a goal-attainment algorithm in Matlab. The optimized finite-element model is then used to create a virtual prototype of the mechanism, and its dynamic performance and first four natural frequencies are plotted across the workspace.

Das and Dülger <sup>[8]</sup> generate a closed-loop dynamics model of a SCARA robot and validate their simulated results with experimental data. Kinematics and rigid-body dynamics are mathematically derived. Transmission losses and friction are not considered. PD control is used for permanent magnet DC motors. Responses for simulated and measured pick-and-place performance are compared, and the simulated response is determined to be satisfactory based on the tolerances required for operation.

Ferretti et al. <sup>[12]</sup> compare a number of alternatives including SimMechanics, ADAMS, and Dymola. They define the requirements of modeling software to be “multi-domain scope, software reuse, reliability and efficiency of numerical simulation, [and] integration with mechanical CADs” <sup>[12]</sup>. Dymola is selected to predict resonant behavior

of manufacturing equipment in a machining center. The models are sufficiently accurate to improve the performance of existing equipment and to predict the performance of new machines with similar structures.

For this thesis three graphical, lumped-parameter modeling software platforms were compared by generating a model of a gearmotor-driven rotary axis. Dymola was eventually selected because of its ease of use, the large number of existing libraries, ease of library component modification, and ease of library generation. This study will only discuss plant models generated mathematically in Simulink and graphically in Dymola.

### **Conclusions**

Moore's Law drives aggressive schedules for the design of new semiconductor manufacturing equipment. Hardware iteration as part of the mechanism design process increases development cost and tool time to market. Closed-loop mechanism dynamics modeling in the design process can significantly reduce hardware iteration by enabling the prediction of mechanism performance prior to prototype parts procurement. Plant dynamics models can be lumped or distributed-parameter, but it is desirable to minimize the complexity of the plant to decrease simulation time and in some applications to enable real-time or hardware-in-loop simulation. For many robotics applications, lumped-parameter models are sufficient for the degree of accuracy required.

In the literature, lumped-parameter plant dynamics models are often derived mathematically, but in an industrial setting a graphical, modular modeling approach is more desirable because it enables rapid model creation and updating. Modular modeling lends itself to the development of libraries of components and subsystems which can be assembled quickly by users who do not have the time to develop a model mathematically.

Multiple lumped-parameter software platforms support a graphical approach.

This thesis develops a flexible modeling approach for semiconductor robotics and demonstrates model performance using a three degrees-of-freedom test mechanism. Plant models will be developed in Dymola, verified mathematically in Simulink, and combined with motor and motion controller models in Simulink to predict the closed-loop performance of the test mechanism.



## CHAPTER 2

# CLOSED-LOOP MODEL GENERATION WITH A RIGID-BODY PLANT

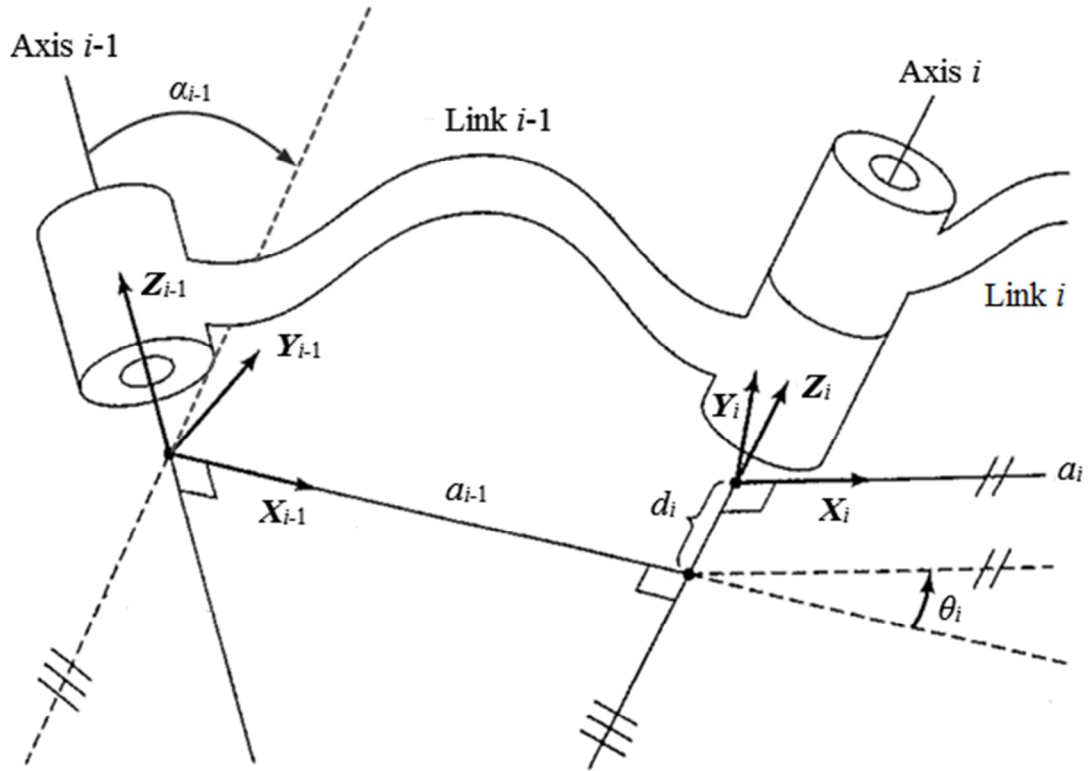
In this chapter, the kinematics of the three degrees-of-freedom test mechanism are described using a Denavit-Hartenberg convention. The governing equations for the rigid-body dynamics are derived using the Euler-Lagrange equation.

### Rigid-Body Plant Model Generation

#### Mathematical Derivation

##### Denavit-Hartenberg Coordinate System Definition

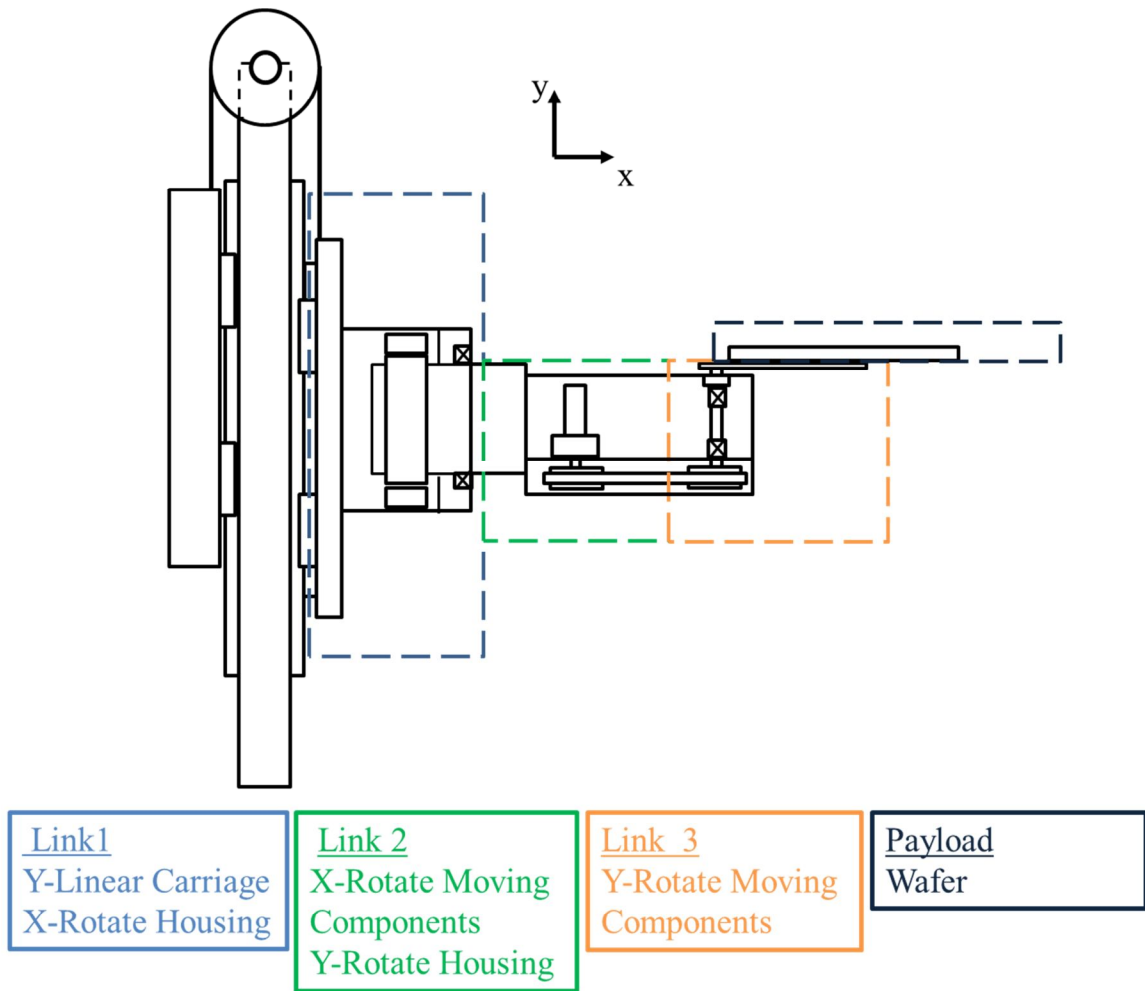
For the mathematical model coordinate systems are positioned according to a Denavit-Hartenberg convention <sup>[6]</sup>. The only exception to this convention is the base, frame 0, is not coincident with frame 1 when the origins are aligned but instead aligns with the system hardware. Frame 0 is fixed in space, but all subsequent frames are body-fixed. Figure 2.1 shows a generic link with attached coordinate frames and the standard parameter definitions used to characterize the link geometry. For the  $n^{\text{th}}$  link the  $z_n$  axis is aligned with the  $n^{\text{th}}$  joint axis, the  $x_n$  axis points along the mutual perpendicular between  $z_n$  and  $z_{n+1}$ , and the orientation of the  $y_n$  axis is determined by the right-hand rule. A set of four parameters defines the coordinate transformation from one coordinate system to the next.



**Figure 2.1: Denavit-Hartenberg Parameters for a Generic Two-Link Configuration.** Image adapted from “Introduction to Robotics Mechanics and Control” [6].

The parameter  $a_i$  measures the distance along  $\vec{x}_i$  from  $\vec{z}_i$  to  $\vec{z}_{i+1}$ . Similarly, the parameter  $d_i$  measures the distance from  $\vec{x}_{i-1}$  to  $\vec{x}_i$  along  $\vec{z}_i$ . There are also two relative angles between the coordinate frames:  $\alpha_i$  is the angle from  $\vec{z}_i$  to  $\vec{z}_{i+1}$  about  $\vec{x}_i$ , and  $\theta_i$  is the angle from  $\vec{x}_{i-1}$  to  $\vec{x}_i$  about  $\vec{z}_i$ .

Figure 2.2 shows the test mechanism divided into three body links. Figure 2.3 shows the locations and coordinate directions of each joint in the test mechanism. A payload frame is also included so that wafer position, velocity and acceleration may be calculated. However, due to the proprietary nature of this mechanism no wafer frame response data is presented or discussed in this study.



**Figure 2.2. Link Structure of Test Mechanism.**

Table 2.1 identifies which link parameters operate under closed-loop control and will therefore be variables in the rigid-body dynamics equations. The remaining parameters are defined or identified as geometry-specific constants. Because of the proprietary nature of this mechanism, specific parameter values are not provided.

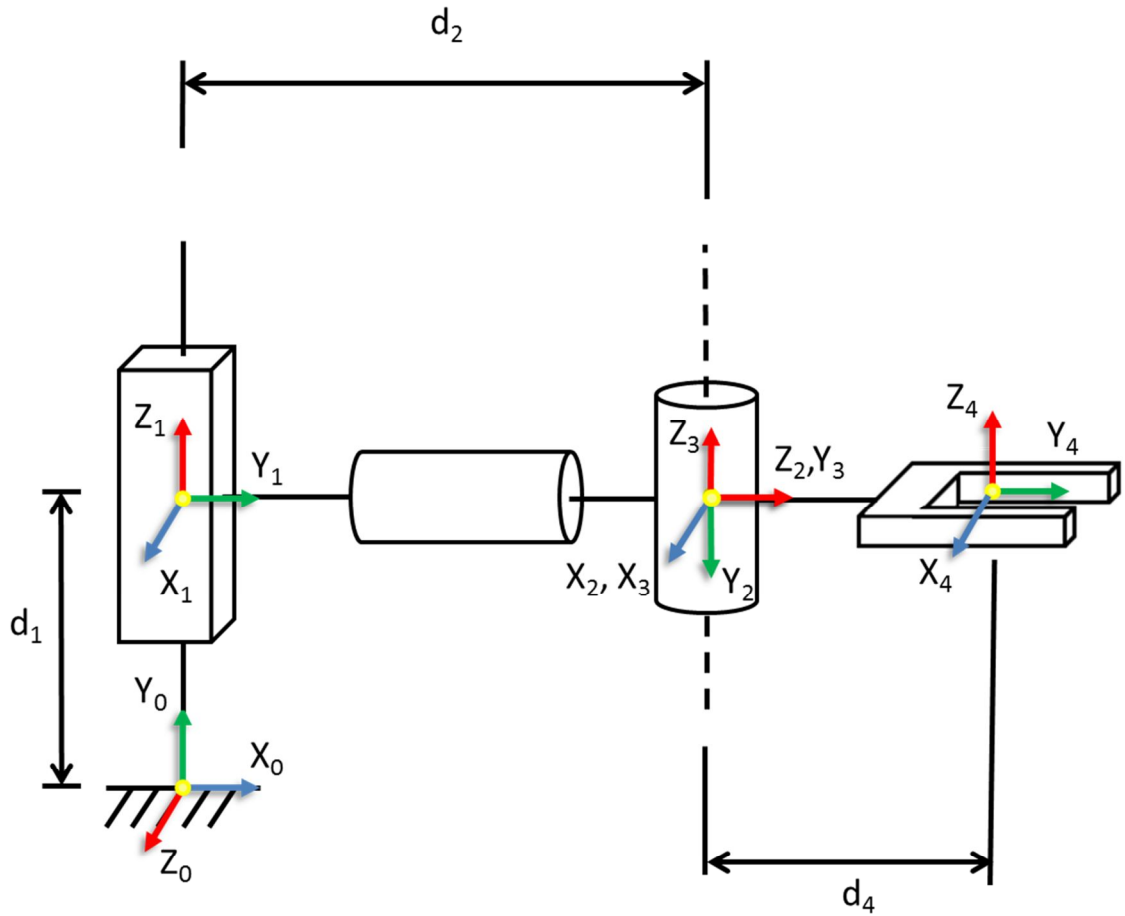


Figure 2.3: Denavit-Hartenberg Coordinate Definitions for Test Mechanism.

Table 2.1: Denavit-Hartenberg Parameters for Test Mechanism.

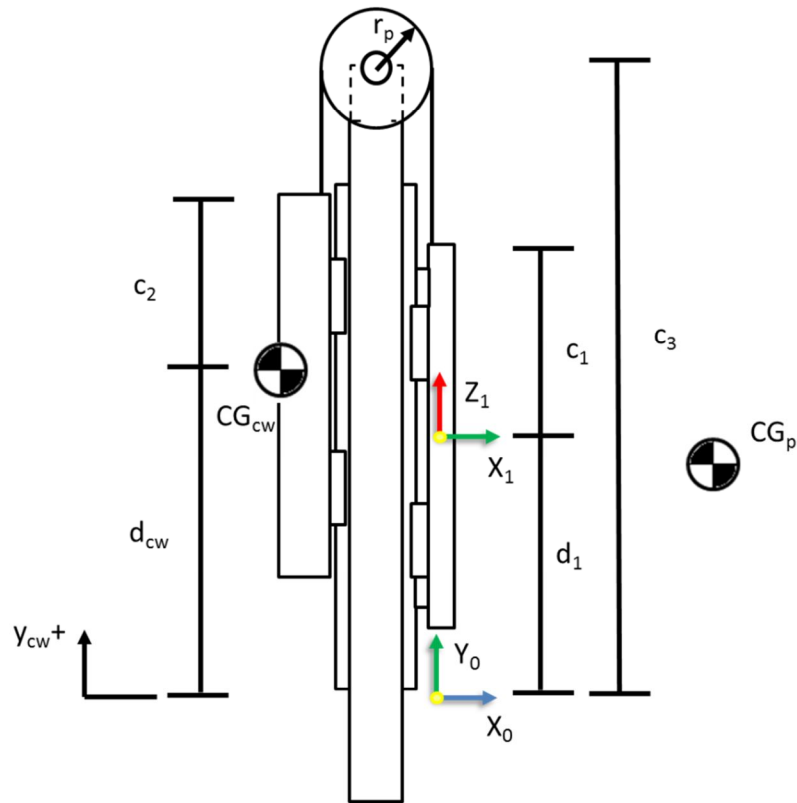
$i$	$a_{i-1}$	$\alpha_{i-1}$	$d_i$	$\theta_i$
1	0	$-90^\circ$	variable	$-90^\circ$
2	0	$-90^\circ$	constant	variable
3	0	$90^\circ$	0	variable
4	0	$0^\circ$	constant	0

## Configuration Constraints

The Denavit-Hartenberg kinematic model includes the three body links, but two critical components of dynamic response are missing. The counterbalance on the Y-Linear axis and the belt drive on the Y-Rotate axis each initially add an additional degree-of-freedom to the system, but these degrees-of-freedom are eliminated in the dynamic response through a pair of geometric configuration constraints.

### *Y-Linear Axis Counterweight*

The position of the counterweight center of gravity,  $CG_{cw}$ , is measured relative to the base frame (Figure 2.4).



**Figure 2.4: Simplified Y-Linear Drive Diagram.**

For inextensible cables the counterweight position  $d_{cw}$  is calculated from the payload position  $d_1$  by

$$d_{cw} = 2c_3 - c_1 - c_2 - L_{cable} + \pi r_p - d_1 \quad (2.1)$$

where  $c_3$  is the vertical distance between the pulleys and the base frame,  $c_1$  is the vertical distance between frame 1 and the cable mount on the payload carriage,  $c_2$  is the vertical distance between the counterweight center of gravity and the cable mount on the counterweight carriage,  $L_{cable}$  is the total cable length, and  $r_p$  is the pulley radius. As  $d_1$  increases,  $d_{cw}$  decreases with an initial offset that is determined by the length of the cables and the system geometry. Differentiating (2.1) yields the counterweight velocity.

$$v_{cw} = -\dot{d}_1 \quad (2.2)$$

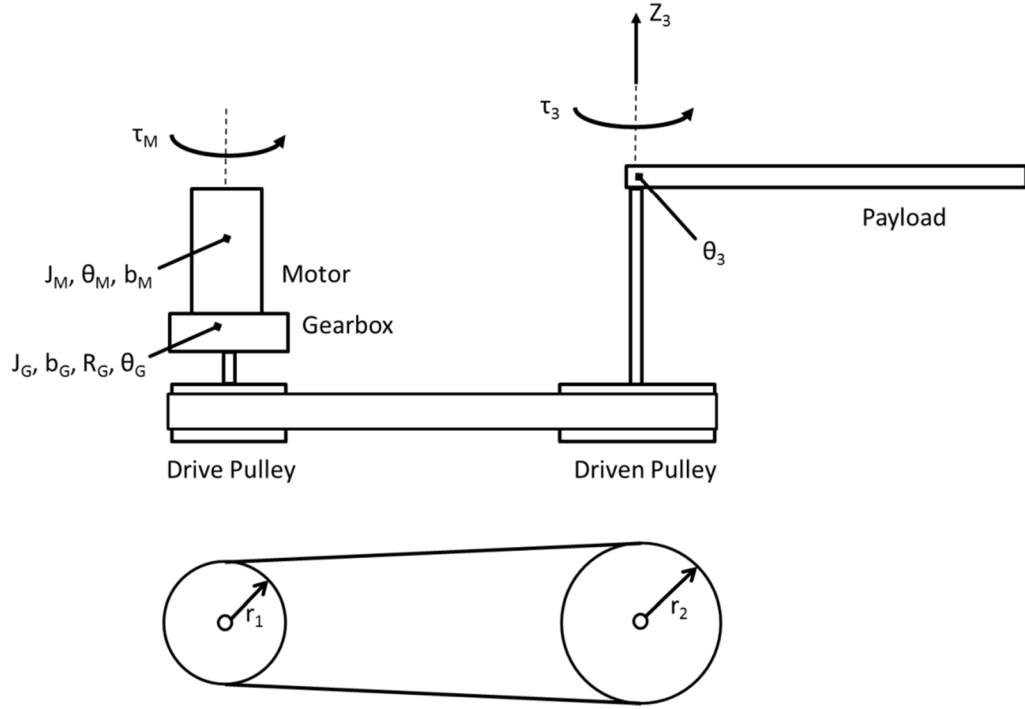
### *Y-Rotate Axis Belt Drive*

Figure 2.5 illustrates the Y-Rotate drive with the variables used for drive dynamics calculations. The Y-Rotate belt drive is described by a pair of configuration constraints relating the motor output shaft position  $\theta_M$ , the gearbox output shaft position  $\theta_G$ , and the payload position  $\theta_3$ .

Assuming the gearbox is infinitely stiff and there is no angular offset, the motor angle,  $\theta_M$ , and the gearbox output angle,  $\theta_G$ , are proportional,

$$\theta_G = \frac{1}{R_G} \theta_M \quad (2.3)$$

where  $R_G$  is the gearbox gear ratio.



**Figure 2.5: Simplified Y-Rotate Drive Diagram.**

Differentiating (2.3) yields the equations for gearbox angular velocity and acceleration.

$$\dot{\theta}_G = \frac{1}{R_G} \dot{\theta}_M \quad (2.4)$$

$$\ddot{\theta}_G = \frac{1}{R_G} \ddot{\theta}_M \quad (2.5)$$

Treating the belt as inextensible yields

$$\theta_3 = \frac{r_1}{r_2} \theta_G \quad (2.6)$$

where  $r_1$  is the drive pulley radius and  $r_2$  is the driven pulley radius. This assumes there is no timing offset between the drive and driven pulleys. Differentiating (2.6) yields the equations for link 3 angular velocity and acceleration.

$$\dot{\theta}_3 = \frac{r_1}{r_2} \dot{\theta}_G \quad (2.7)$$

$$\ddot{\theta}_3 = \frac{r_1}{r_2} \ddot{\theta}_G \quad (2.8)$$

### Kinematics

Using Denavit-Hartenberg notation the homogeneous transformation matrix from coordinate system  $i$  to coordinate system  $i-1$  is

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i \cdot \cos\alpha_{i-1} & \cos\theta_i \cdot \cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1} \cdot d_i \\ \sin\theta_i \cdot \sin\alpha_{i-1} & \cos\theta_i \cdot \sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1} \cdot d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

Since the first coordinate frame has a constant orientation its angular velocity is

$$\vec{\omega}_1 = \vec{0} \quad (2.10)$$

For the remaining frames the angular velocities with respect to the base are,

$$\vec{\omega}_2 = \dot{\theta}_2 \vec{z}_2 \quad (2.11)$$

$$\vec{\omega}_3 = \vec{\omega}_2 + \dot{\theta}_3 \vec{z}_3 \quad (2.12)$$

$$\vec{\omega}_4 = \vec{\omega}_3 \quad (2.13)$$

The linear velocities,  $\vec{v}_i$ , of each of the four coordinate frame origins are,

$$\vec{v}_1 = \dot{d}_1 \vec{z}_1 \quad (2.14)$$

$$\vec{v}_2 = \vec{v}_1 \quad (2.15)$$

$$\vec{v}_3 = \vec{v}_2 \quad (2.16)$$

$$\vec{v}_4 = \vec{v}_3 + \vec{\omega}_4 \times d_4 \vec{y}_3 \quad (2.17)$$

For kinetic energy calculations, the velocity at the center of gravity of each link,  $\vec{v}_{G_i}$  with  $i=1, 2, 3$  are needed. The mass and inertia of the end effector are included in the third link.

$$\vec{v}_{G1} = \vec{v}_1 \quad (2.18)$$



$$\vec{v}_{G2} = \vec{v}_2 + \vec{\omega}_2 \times \vec{r}_{G2/2} \quad (2.19)$$

$$\vec{v}_{G3} = \vec{v}_3 + \vec{\omega}_3 \times \vec{r}_{G3/3} \quad (2.20)$$

In the above equations  $\vec{r}_{Gi/i}$  denotes a vector from coordinate frame  $i$  to the center of gravity of link  $i$ .

### Dynamics

The rigid-body mechanism dynamics are calculated using energy methods. First, the kinetic energy of each of the three links, the counterweight, and the Y-Rotate gearmotor are calculated. The total kinetic energy  $T_{tot}$  is the sum of the kinetic energies of the links  $T_i$ , the counterweight  $T_{cw}$ , and the Y-Rotate integrated gearmotor  $T_{3M}$ ,

$$T_{tot} = T_1 + T_2 + T_3 + T_{cw} + T_{3M} \quad (2.21)$$

For the  $i^{th}$  link kinetic energy is

$$T_i = \frac{1}{2} m_i \vec{v}_{G,i}^T \vec{v}_{G,i} + \frac{1}{2} \vec{\omega}_i^T \underline{I}_{j/Gi} \vec{\omega}_i \quad (2.22)$$

where  $m_i$  is the mass and  $\underline{I}_{j/Gi}$  is the inertia tensor.

Next, the potential energy for each link is calculated as the sum of the link potential energies. It should be noted that the potential energy of the Y-Rotate gearmotor is included in the  $V_3$  term,

$$V_{tot} = V_1 + V_2 + V_3 + V_{cw} \quad (2.23)$$

The potential energy of the  $i^{th}$  link,  $V_i$ , is defined as

$$V_i = -m_i \vec{g} \cdot \vec{r}_{Gi/0} \quad (2.24)$$

where  $\vec{r}_{Gi/0}$  is a vector from the base frame to the center of gravity of the  $i^{th}$  link and  $\vec{g} = [0 \ -9.81\text{m/s}^2 \ 0]$ . Thus gravitational potential energy is defined from the base frame as a datum.

The total kinetic and potential energies are used to form the Lagrangian,

$$\mathcal{L} = T_{tot} - V_{tot} \quad (2.25)$$

The dynamic model follows by applying the Euler-Lagrange equations,

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}_j} \right) - \frac{\partial \mathcal{L}}{\partial q_j} = Q_j \quad (2.26)$$

where the  $q_j$ 's are the generalized system coordinates

$$q_1 = d_1 \quad (2.27)$$

$$q_2 = \theta_2 \quad (2.28)$$

$$q_3 = \theta_3 \quad (2.29)$$

and the  $Q_j$ 's are the generalized forces applied to the system. The generalized forces are determined from the principle of virtual work.

$$\delta W = \sum_{j=1}^m Q_j \delta q_j = \sum_{j=1}^k F_j \delta r_j + \sum_{j=1}^p \Gamma_j \delta \theta_j \quad (2.30)$$

Virtual work  $\delta W$  is the sum of the products of each generalized force  $Q_j$  with its virtual displacement  $\delta q_j$ , and  $m$  is the total number of degrees-of-freedom. The equation then expands the generalized forces into translational and rotational terms. Index  $k$  is the number of translational degrees-of-freedom, and  $F_j$  is the applied force. Index  $p$  is the number of rotational degrees-of-freedom, and  $\Gamma_j$  is the applied moment. The total number of generalized coordinates used to describe the system is  $k+p=m$ . The generalized forces are,

$$Q_1 = F_{1M} \quad (2.31)$$

$$Q_2 = \tau_{2M} \quad (2.32)$$

$$Q_3 = \tau_{3P} \quad (2.33)$$

where  $F_{1M}$  is the force applied by the linear motor to link 1,  $\tau_{2M}$  is the torque applied by the direct-drive rotary motor to link 2, and  $\tau_{3P}$  is the torque applied by the motor to link 3 through the belt drive determined using the gear and pulley ratios,

$$\tau_{3P} = R_G \frac{r_2}{r_1} \tau_{3M} \quad (2.34)$$

The dynamic equations are generated symbolically in Matlab (Appendix B). First the Denavit-Hartenberg parameters are defined as a structure. Transformation matrices between coordinate frames are generated, and position vectors between frames are defined and transformed into frame 0 coordinates. The inertia tensors are defined and transformed into frame 0 coordinates. Angular and linear velocity vectors are defined for each frame and used to calculate the kinetic energy of each link. Link kinetic energies are summed to determine the total kinetic energy. Potential energies of each link are determined and summed. The Lagrangian is calculated and input into a Lagrange function which symbolically differentiates to determine the equations of motion.

## **Conclusions**

The kinematic and dynamic equations for a rigid model of the test mechanism are relatively simple to implement using a symbolic solver. However, the resulting equations are quite long due to the required coordinate transformations. For a design application, these equations do not provide much insight into the system response without the development of a complete numerical solution. For model-based control applications, each mathematical operation requires processor time, so significant simplification is required to enable the models to run real-time. These equations will be solved numerically and used to verify a rigid Dymola model in the following chapter.

## **CHAPTER 3**

### **SOFTWARE MODELING**

Motion controller and amplifier models are generated in Simulink, and motor and rigid mechanism dynamic models are created in Dymola. While Dymola has the capability for control modeling, Matlab/Simulink is a standard software for control design and simulation, so implementing the control models in Simulink enables more effective model sharing. A rigid-body mechanism dynamic model is also created in Simulink for mathematical verification of the Dymola plant and verification of the Dymola/Simulink integration process.

#### **Mechanism Dynamic Model Generation**

Dynamic models of the test mechanism are developed in Simulink and Dymola. The Simulink model is used to verify the performance of the Dymola model.

#### **Matlab Implementation**

The symbolically generated dynamic equations are implemented in Simulink through the Matlab function block (Figure 3.1). The inputs to the model are the actuator currents for each the three motion axes. They are converted to forces/torques through the force/torque constant and applied to the mechanical plant model. The model outputs acceleration which is integrated to determine the position and velocity of each axis.

#### **Dymola Implementation**

The same plant dynamics are also modeled in Dymola. Dymola is a GUI-based front end for the Modelica physical-system modeling language. Modelica is an open-source language for multi-domain modeling. Most of the models in this study are built

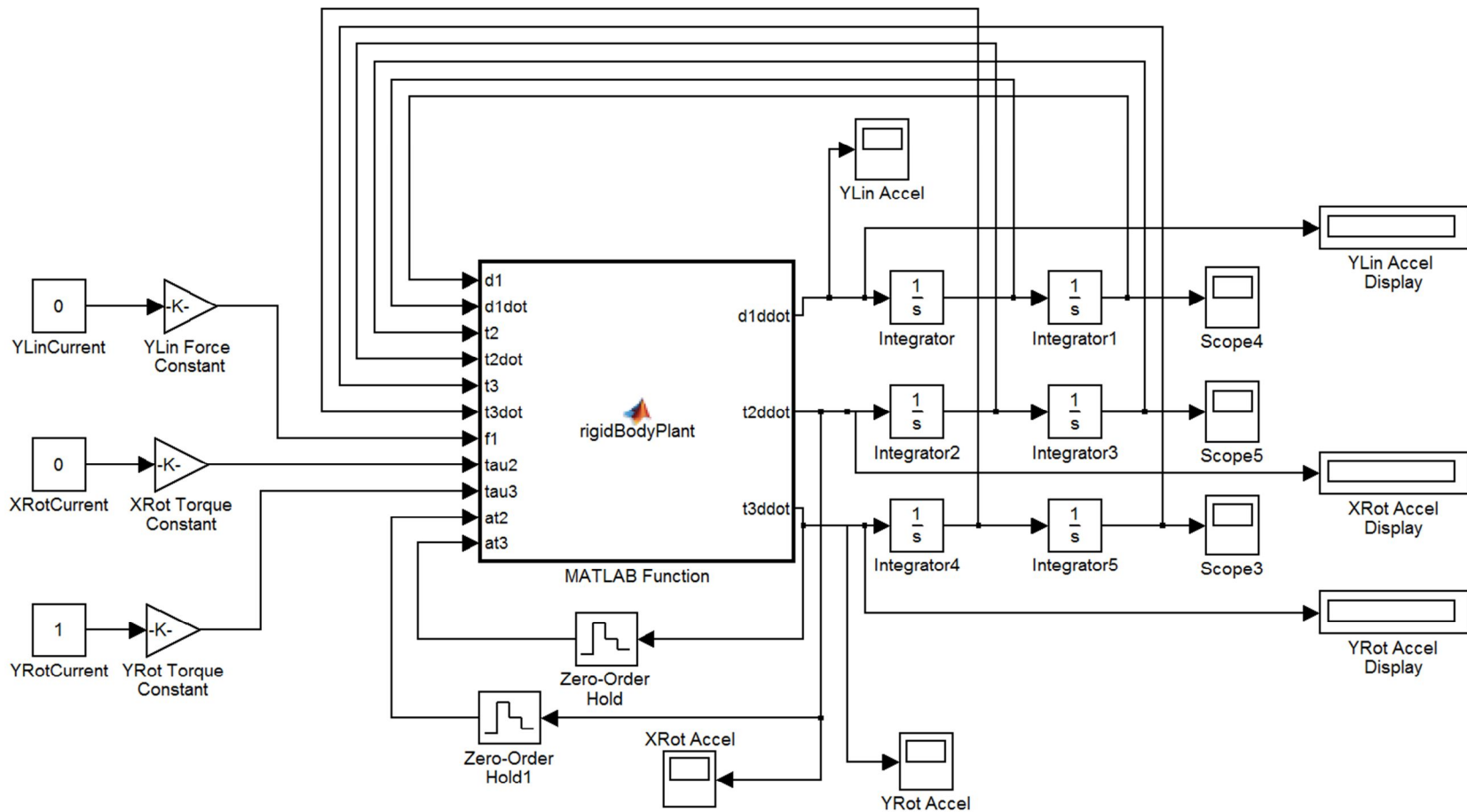
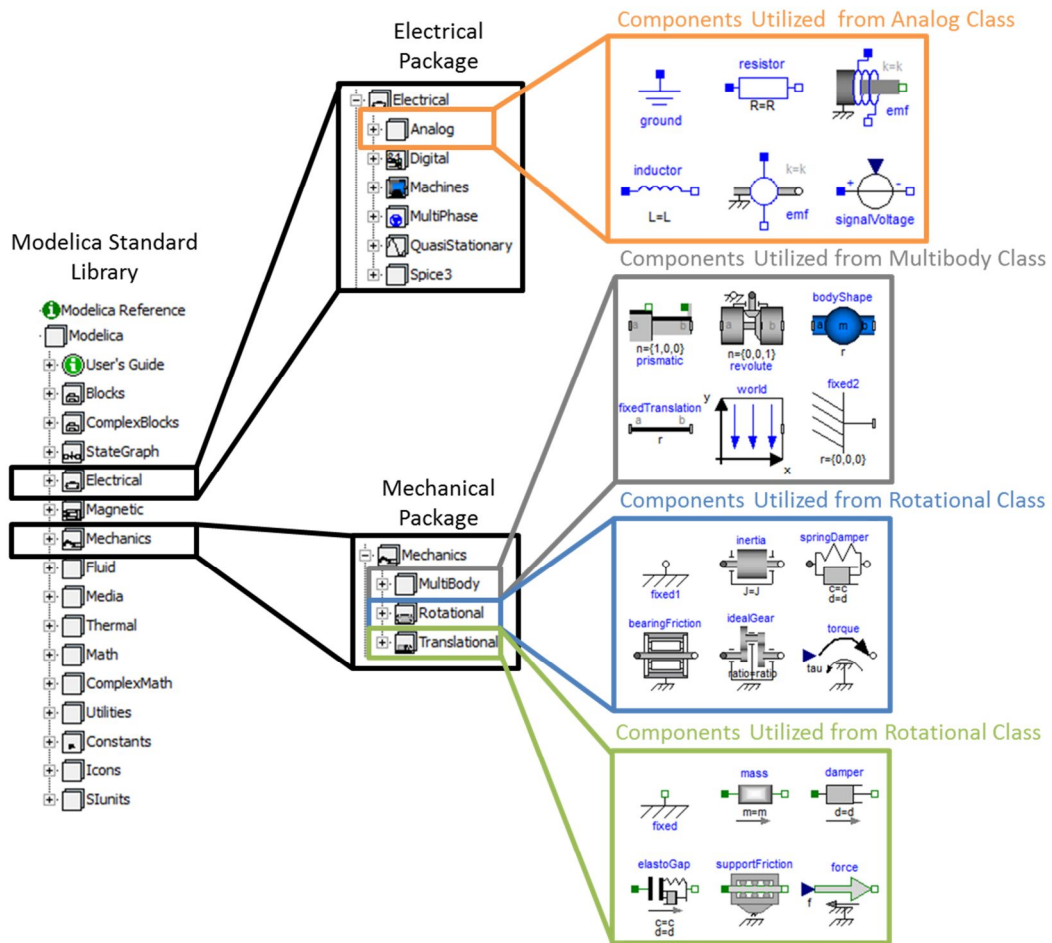


Figure 3.1: Open-Loop Plant Dynamic Model.

from or adapted from component models in the Modelica Standard Library which is a set of verified open-source component models spanning a wide range of domains including mechanical, electrical and thermal. Figure 3.2 shows the components used in this study and their location in the Modelica Standard library.



**Figure 3.2: Modelica Standard Library Structure and Utilized Components**

Models in Modelica are energy-based. Energy is exchanged between components through the ports, which is also how components are connected graphically. There are two types of variables, across and through. Across variables such as velocity and voltage are measured across two ports and their values are passed between components. For example, if two rotational components are graphically connected the angular position of

the terminal flange (flange b) on the first component is identical to the position of the base (flange a) of the second. Through variables, such as forces and currents, are measured through the port, and their values sum to zero at a connection between components. Figure 3.3 shows a Dymola model of a torque applied to a simple lumped inertia mounted on a torsion spring. The ports on the inertia and spring are identified and the relationships defined by the graphical connection are listed.

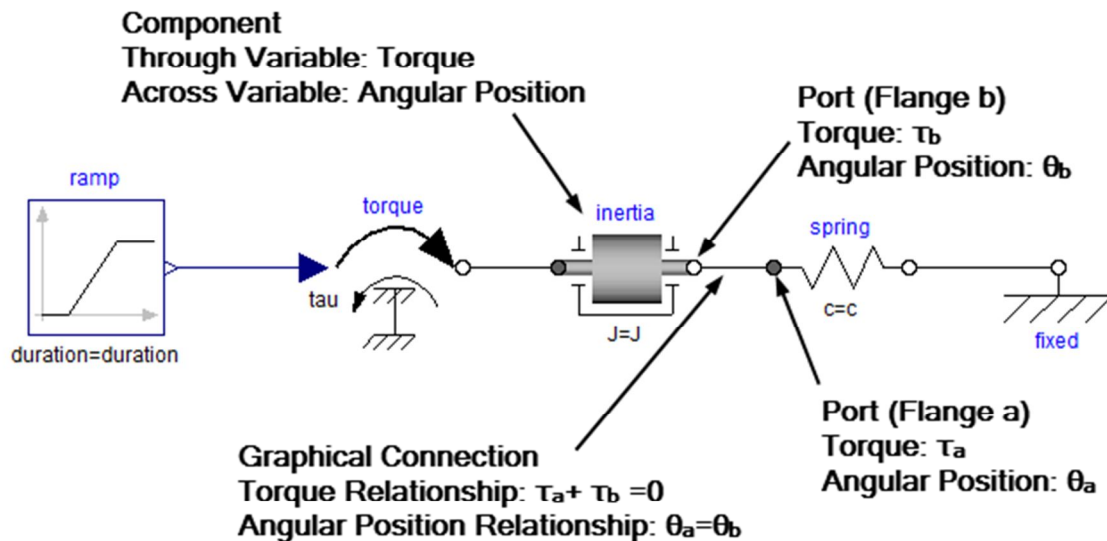







Figure 3.3. Across and through variables defined for a simple rotational Dymola model.

It should be noted that through variables are identified in Modelica with the keyword “flow”, which may cause confusion in the mechanical domain where bond graph flow variables actually correspond to across variables. Each connector type is defined to pass specific variables, so connections between differing connector types are not possible. The connectors differ in shape and color to enable the user to quickly identify where connections are possible. Table 3.1 shows the types of connectors used in this study and identifies the signals passed through each connector type.

**Table 3.1: Relevant Modelica Connectors.**

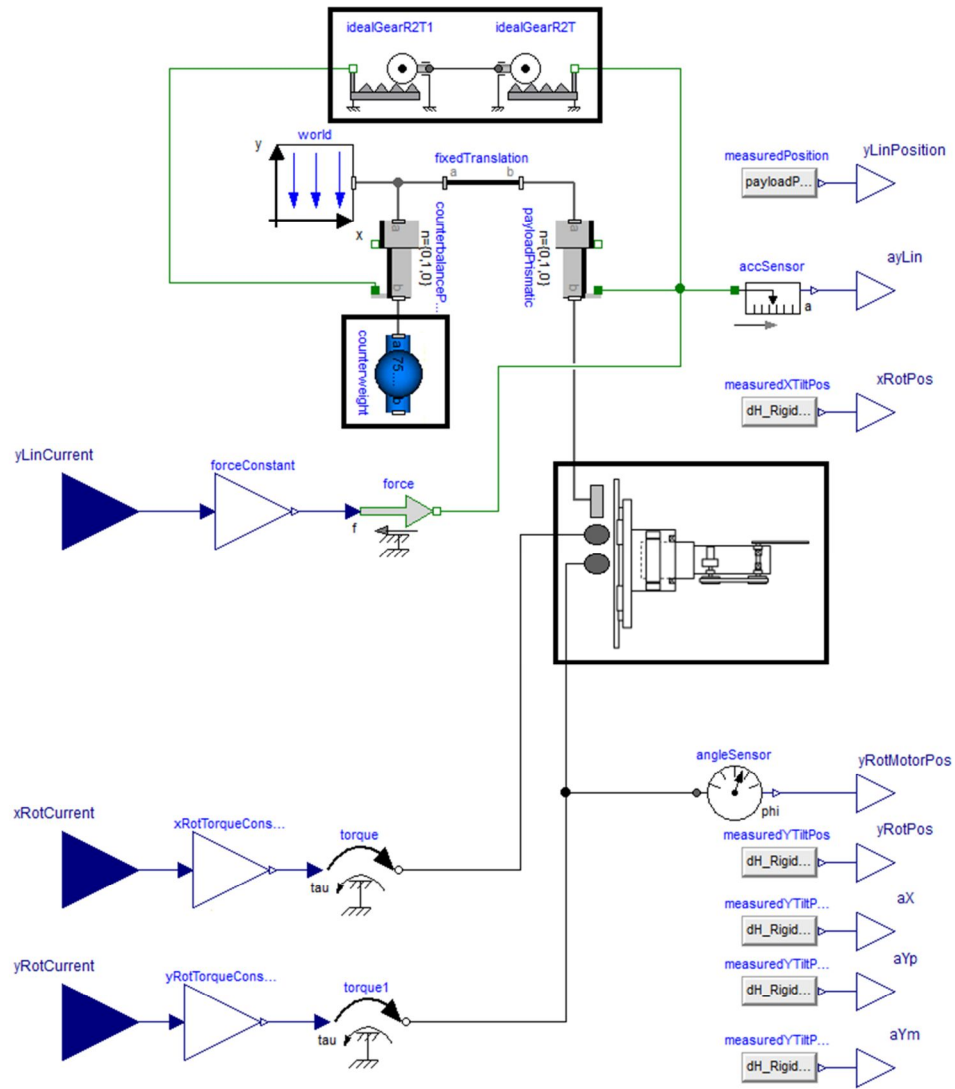
Connector Type	Signals	Connector Style
Real Value	Real-valued Signal	
Electrical Pin	Voltage Current	
Rotational Flange	Torque Angle	
Translational Flange	Force Position	
Multibody Frame	Position Vector Orientation Matrix Force Vector Torque Vector	

The 3 DOF mechanism model in Dymola is shown in Figure 3.4. Actuator currents for the Y-Linear, X-Rotate, and Y-Rotate axes are input on the left side. They are multiplied by the force/torque constants for each motor and converted to forces/torques. For the Y-Linear axis, the force is used to drive the payload prismatic joint through the translational axis flange. The flange also connects to the pulley which is modeled with a pair of idealGearR2T components which convert linear to rotary motion through a user-specified gear ratio. On the output of the pulley is the counterweight prismatic joint which drives the motion of the counterweight modeled as a single rigid link.

The multi-body output flange of the prismatic joint is connected to the payload subsystem model, shown graphically as an image of the Y-Linear carriage payload in Figure 3.5. The X-Rotate and Y-Rotate applied torques are input into the payload



subsystem. The X-Rotate input drives a revolute joint about the X-axis through a lumped inertia representing the motor rotor and ferrofluid seal inertias. The Y-Rotate input drives

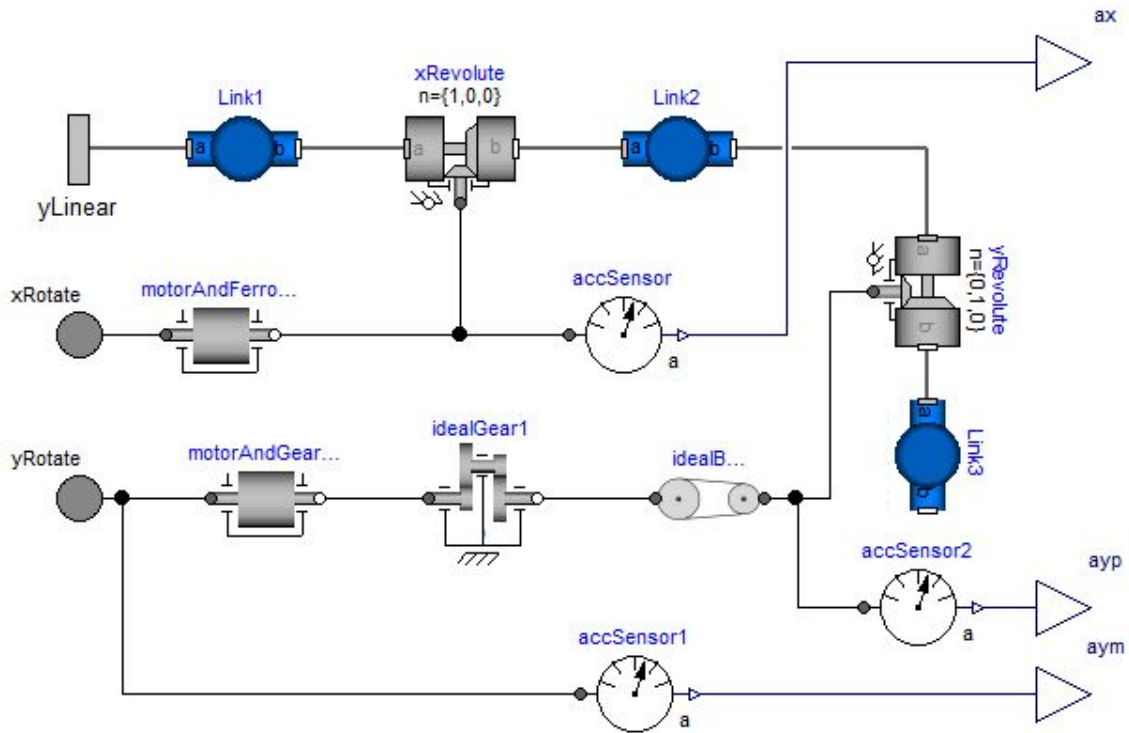


**Figure 3.4: Rigid-Body Model of Test Mechanism in Dymola.**

a revolute joint about the y-axis through a drive train consisting of the motor inertia, the gear ratio, and the ratio of the driven pulley radius to the drive pulley radius in the belt drive. All drive components are assumed to be frictionless and rigid.

## Open-Loop Simulation

Models in Dymola can be utilized in Simulink in one of two ways. The functional mock-up interface (FMI) standard enables model exchange across a number of software

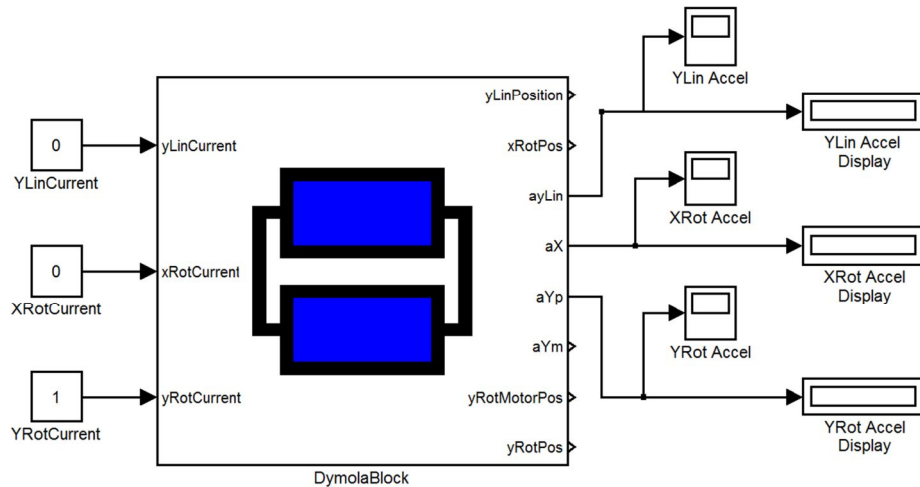


**Figure 3.5: Rigid-body Payload Sub-Model in Dymola.**

platforms. Models are exported from Dymola as a functional mock-up unit (FMU) and can then be imported into Simulink in the same format. A second older method for model transfer uses the Dymola-Simulink interface where the Dymola model is converted to C-code and is run in Simulink as an S-function <sup>[10]</sup>. The second method is sufficient for the purposes of this study since only model transfer from Dymola to Simulink is of interest. Figure 3.6 shows the Dymola rigid-body plant model implemented as an S-function block in Simulink.

Both the Simulink plant and the Dymola plant were run open loop with a 1A current applied to each axis. The acceleration for each of the three motion axes was

measured in both models, and as expected, they were nearly identical. The Simulink and Dymola models differed by  $1.4e-5\%$ ,  $4.3e-4\%$ , and  $2.7e-3\%$  for the Y-Linear, X-Rotate and Y-Rotate axes, respectively. This verifies that the model in Dymola is performing as expected.



**Figure 3.6: Dymola Open-Loop Plant Dynamic Model.**

### Controller Model Generation

Motion and motor controller models are generated in Simulink.

#### PMAC Motion Controller

The multi-axis motion controller used is the Delta Tau Turbo PMAC2 Ultralite that can control up to 24 motion axes simultaneously <sup>[33]</sup>. The PMAC provides proportional-integral-derivative (PID) control for position, along with feedforward terms for friction, velocity, and acceleration. Servo updates are at 2.25kHz, and encoder feedback is communicated at 9kHz. This means that the encoder data used in the motion calculations is more recent than the previous servo cycle. User-specified motion control programs in the PMAC generate the commanded motion profile for each axis. At a high level, every servo cycle the PMAC increments the commanded position, compares the commanded value to the measured position (read from sensor feedback), and outputs a

command to the current controller based on the difference. A simplified diagram of the control loop is shown in Figure 3.7. In this figure, some scale factors and limits are not shown for clarity.

The controller inputs are the commanded position from the motion profile generator, position feedback from the load-side encoder and position feedback from the motor-side encoder (in the case of a mechanism with a single encoder, the two positional feedbacks will be identical). The direction of the friction feedforward is determined by the sign of the commanded velocity. The velocity and acceleration feedforward terms are proportional to the first and second derivatives of the commanded position, respectively. The error between the commanded position and the measured load-side position is integrated when the integrator is on. The state of the integrator is determined by a user input variable. For this study, the integrator is active only when the commanded velocity is zero. This prevents the integrator from affecting the system stability during motion. The integral feedback is added to the positional error. This sum is then added to the acceleration and velocity feedforward terms. Then the derivative of the measured motor position for a dual-encoder system, or measured load position if only a single encoder is being used, is subtracted. This total is multiplied by the proportional gain and filtered before being added to the friction feedforward term. This sum is the commanded current in the units used by the digital to analog converter (DAC units). The commanded current in DAC units is converted to a commanded current in Amps via the transconductance value, the ratio of peak amplifier current to peak DAC voltage output. The controller gains are defined by user-input I-variables. Table 3.2 provides an overview of the key motion control parameters required to set up the controller for a single motor<sup>[33]</sup>.

Figure 3.8 shows the Simulink model of the PMAC controller. The foundation of the model is provided by Delta Tau with a few key additions. The model combines the simplified loop structure shown in Figure 3.6 with additional scale factors and limits. The provided model does not include the integration mode selector or the option to read

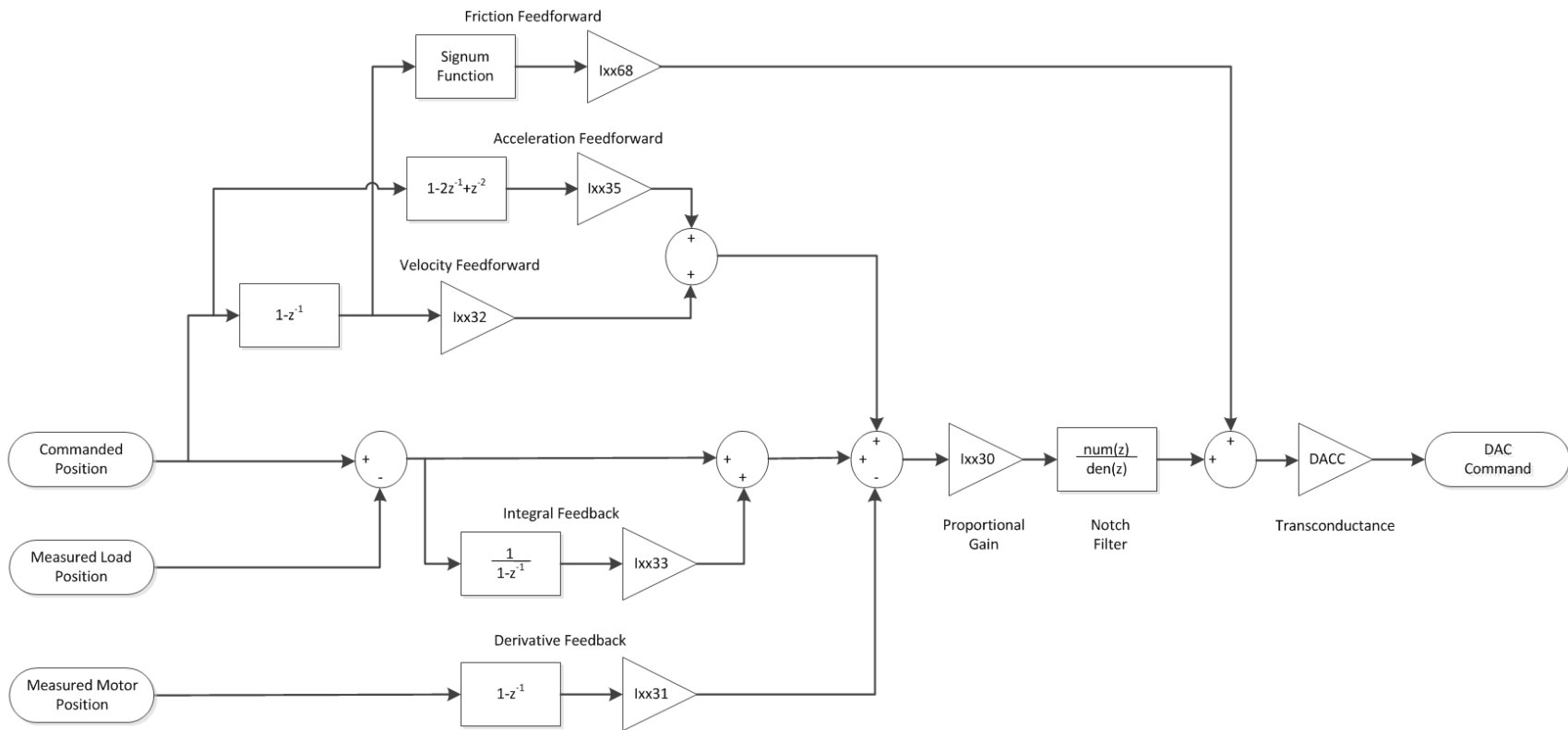


Figure 3.7: Simplified Structure of PMAC Control Loop. Some scale factors and limits are not shown for clarity

**Table 3.2: Relevant PMAC Motor Parameters.**

Parameter	Definition
Ixx03	Motor xx Position Loop Feedback Address
Ixx04	Motor xx Velocity Loop Feedback Address
Ixx07	Motor xx Master Scale Factor
Ixx08	Motor xx Position Scale Factor
Ixx09	Motor xx Velocity Loop Scale Factor
Ixx16	Motor xx Maximum Program Velocity
Ixx17	Motor xx Maximum Program Acceleration
Ixx30	Motor xx PID Proportional Gain
Ixx31	Motor xx PID Derivative Gain
Ixx32	Motor xx PID Velocity Feedforward Gain
Ixx33	Motor xx Integral Gain
Ixx34	Motor xx Integration Mode
Ixx35	Motor xx PID Acceleration Feedforward Gain
Ixx36	Motor xx PID Notch Filter Coefficient N1
Ixx37	Motor xx PID Notch Filter Coefficient N2
Ixx38	Motor xx PID Notch Filter Coefficient D1
Ixx39	Motor xx PID Notch Filter Coefficient D2
Ixx63	Motor xx Integration Limit
Ixx65	Motor xx Deadband Size
Ixx68	Motor xx Friction Feedforward
Ixx69	Motor xx Output Command Limit

velocity feedback from a second encoder in the case of a dual-encoder system. The block diagram of the integration mode selector is shown in Figure 3.9. The selector first looks at Ixx34, the user-input integration mode. If Ixx34=0, then the integrator is always on. If Ixx34=1, then the integrator is turned on when the velocity is zero, and the integrator is turned off otherwise.

The feedback portion of the model has been modified to read feedback from two encoders. The positional error is always calculated from the primary encoder, whose address is defined by Ixx03. If Ixx03=Ixx04, then the velocity loop also uses the primary encoder to calculate the velocity feedback. However, Ixx04 may also address a secondary

Model PMAC  
 PID Filter  
 Delta Tau  
 Data Systems, Inc.

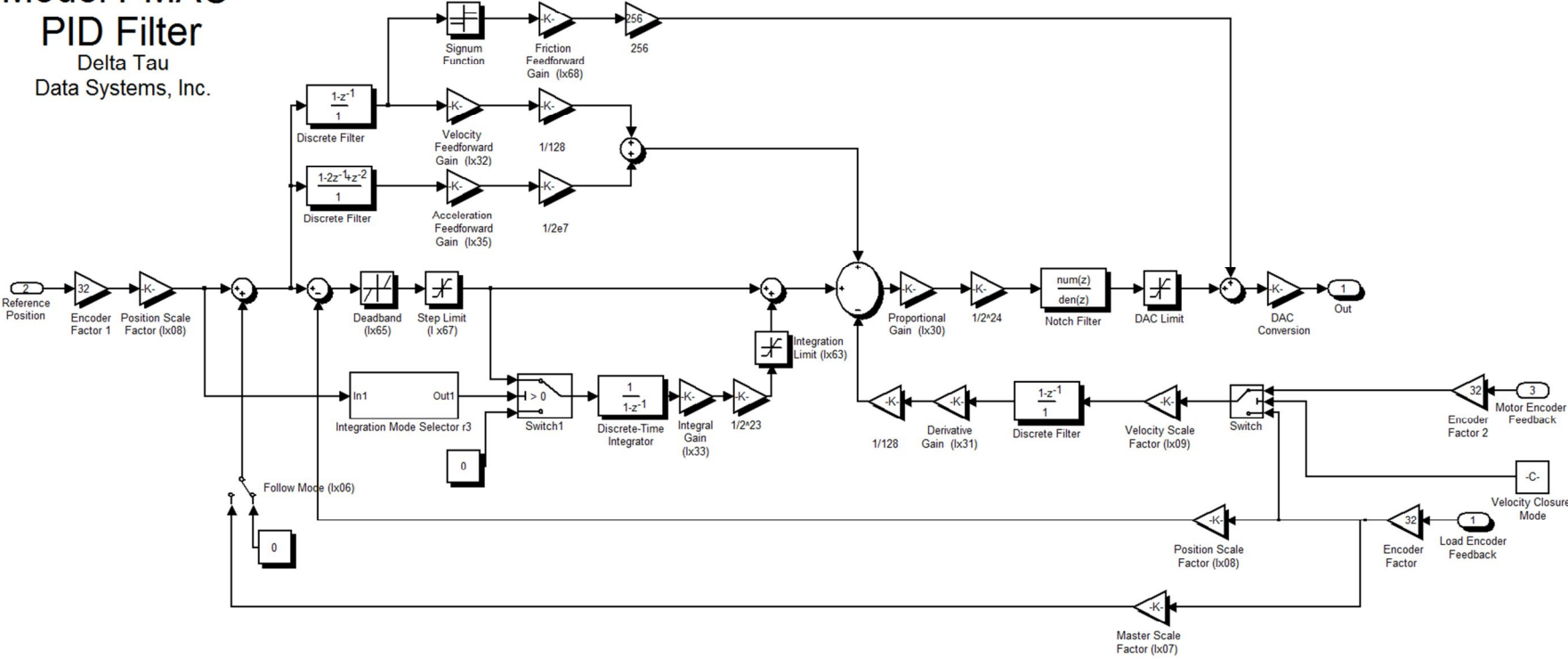
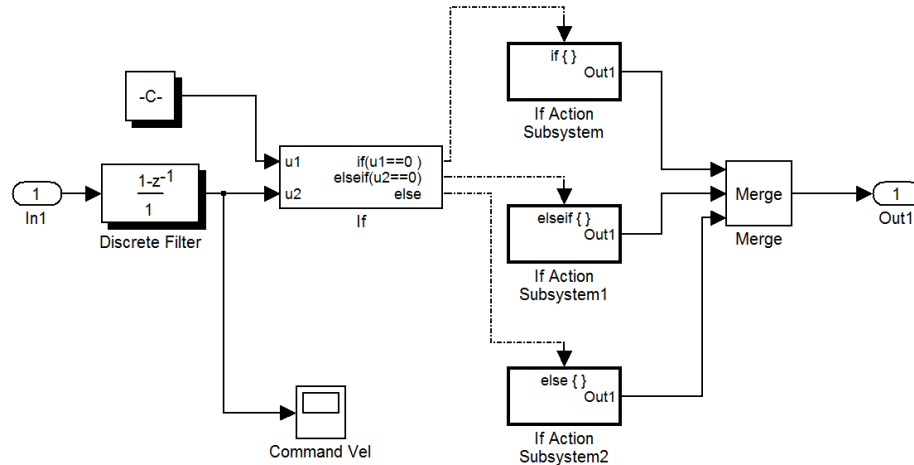


Figure 3.8: Simulink Model of the PMAC Motion Controller.



**Figure 3.9: Simulink Model of the Integration Mode Selector.**

encoder. If this is the case, the secondary encoder is used to calculate the velocity feedback. Dual-encoder systems are typically used for compliant mechanisms where it is possible for the motor to “wind up” relative to the payload. In a wind-up situation, the motor position changes, but because of mechanism compliance the payload is initially stationary. For the dual-encoder configuration, a large motor velocity will reduce the controller command, in turn reducing the system wind up. No wind-up is possible in the model developed in this chapter because the belt is modeled as inextensible, but in later chapters the Y-Rotate gearbox and belt stiffnesses are considered.

### **Copley Motor Controller**

The command current from the PMAC (in Amps) is then input to a Copley Xenus MACRO amplifier. In this application the Copley is used in current mode to provide closed-loop control of the motor current at a rate of 18kHz. For a three-phase motor, the Copley controls two of the three motor phases. Figure 3.10 shows the Copley Amplifier Simulink block diagram.



For phases 1 and 2 the command current is multiplied by the phasing. The phasing generates an analog approximation of the motor commutation by changing the commanded current in each phase based on the motor position,

$$phasing = \sin\left(n\theta - \frac{2\pi}{3}(\zeta - 1)\right) \quad (3.1)$$

where  $\zeta$  is the phase number,  $n$  is the number of poles for a rotary motor, and  $\theta$  is the motor angle. In the case of a linear motor,  $n\theta$  is replaced by  $\frac{2\pi x}{\rho}$ , where  $x$  is the linear motor position and  $\rho$  is the motor electrical cycle length, the distance traveled in one complete electrical cycle (provided by the motor supplier). Thus the commands to each motor phase are equal in peak amplitude but 120 degrees out of phase. A zero-order hold is used on the current feedback to the controller to limit the rate to 18kHz and to establish priority, ensuring the feedback portion of the model is solved after the control portion. The phase 3 command current is calculated from the negative of the sum of the phase 1 and 2 command currents.

A block diagram of the Copley control loop is shown in Figure 3.11. The DC current offset is added to the commanded phase current <sup>[5]</sup>. The command is then filtered. The current loop has two built-in configurable Butterworth/Biquadratic filters with user-defined filter parameters. For the purposes of this study, the DC current offset and the command current filters are not utilized, so the blocks in these locations serve only as placeholders for future model development. The commanded phase current is then limited to the user-specified peak and continuous current values. The actual phase

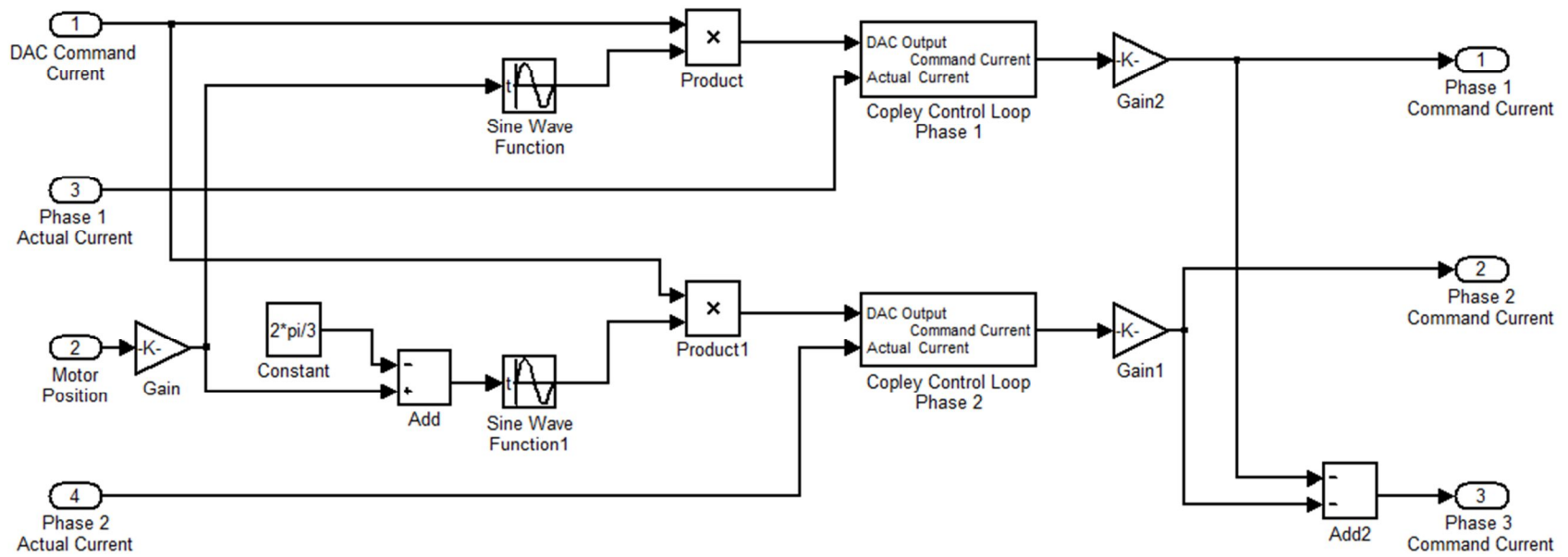
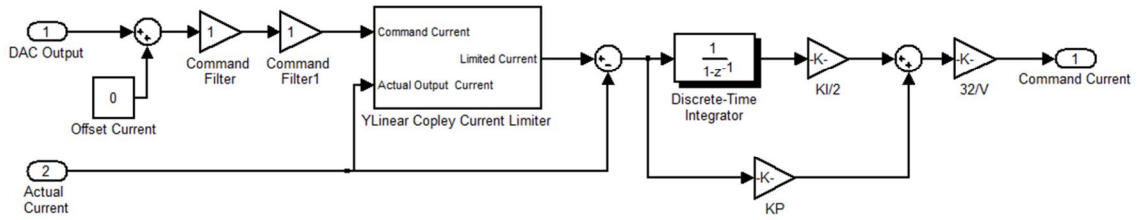


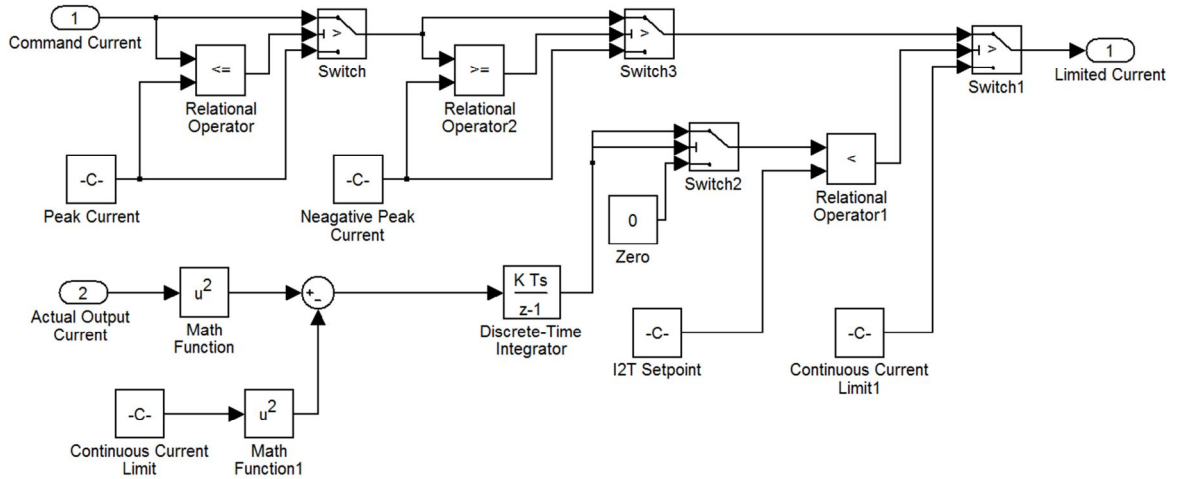
Figure 3.10: Simulink Model of Copley Motor Controller.



**Figure 3.11: Simulink Model of the Copley Phase Current Control Loop.**

current is subtracted from the limited command current, and the error is then input into a PI control loop. The output of the control loop is then multiplied by a gain of 32 divided by the bus voltage. This value is then multiplied by the ratio of total bus voltage to peak current (Figure 3.10), generating a command voltage which is sent to the pulse-width modulation (PWM) stage to generate sinusoidal commutation in each of the three motor phases. It is assumed that the frequency of the switching in the PWM stage is high enough that it has no significant effect on the response of the mechanism. Additionally, modeling the PWM stage forces the simulation to take extremely small solver time steps, increasing overall simulation time and complexity. For these reasons the PWM stage is not included in the mechanism model. The controller is instead treated as analog, and the commanded phase voltage is applied directly to the motor phase.

The Simulink diagram of the current limiter block in the Copley control loop model is shown in Figure 3.12. The current limiter has two primary functions: ensure the amplitude of the current never exceeds the user-specified peak current limit and ensure the integrated current never exceeds a calculated set point value <sup>[5]</sup>. The magnitude of the command current is first compared to the peak current limit. If the magnitude exceeds the peak current limit the command current is limited to the peak current.



**Figure 3.12: Simulink Model of Copley Current Limiter.**

The continuous current set point is calculated as

$$I_{2T} \text{ Setpoint} = (i_{peak}^2 - i_{cont}^2) \cdot t_{I2T} \quad (3.2)$$

where  $i_{cont}$  is the continuous current limit and  $t_{I2T}$  is the  $I^2T$  time limit (the allowable time for which the continuous current may be applied). An accumulator variable is used to track the integrated current and must be non-negative. The accumulator value is calculated incrementally using

$$accumulator_{n+1} = accumulator_n + (i_{actual}^2 - i_{cont}^2) \cdot \Delta t \quad (3.3)$$

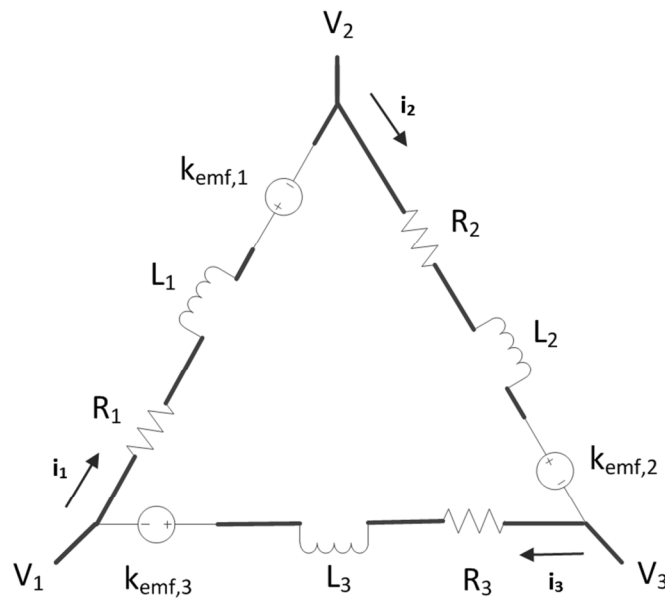
Every time step the accumulator value,  $accumulator_n$ , is incremented by adding the difference between the square of the actual current  $i_{actual}$  and the continuous current limit scaled by the time step  $\Delta t$ . The accumulator value is compared to the set point, and if the accumulator is larger than the set point the current is limited to the continuous current value. Otherwise the commanded current is passed through the current limiter unmodified.

## Motor Model Generation

Models are developed for a three-phase Delta wound linear motor and a three-phase Wye wound rotary motor.

### Three-Phase Delta Wound Linear Motor

The linear motor used to drive the Y-Linear axis is a three-phase delta wound brushless DC (BLDC) motor. Figure 3.13 shows a schematic of the motor circuit <sup>[18]</sup>.



**Figure 3.13: Circuit Diagram of Three-Phase Delta Wound Brushless DC Motor.**

Each phase of the motor has an associated resistance  $R_\zeta$ , inductance  $L_\zeta$ , and back EMF  $k_{emf,\zeta}$ , where  $\zeta$  is the phase number,  $\zeta=1, 2, 3$ . For the purposes of this analysis, the resistances and inductances of the three phases are assumed equal,  $R_\zeta = R$  and  $L_\zeta = L$  respectively. It is important to note that motor manufacturers typically specify the lead-to-lead resistance and inductance values rather than phase resistance and inductance <sup>[20]</sup>. From the circuit diagram, the phase resistance and inductance values are calculated as

$$R = \frac{3}{2} R_{lead-to-lead} \quad (3.4)$$

$$L = \frac{3}{2} L_{lead-to-lead} \quad (3.5)$$

The magnitude of the back-emf constant for each of the three phases is assumed to be equal, but the back-emf voltage loss in each phase is scaled by the phasing. As with the resistance and inductance values, motor vendors may also provide the back-emf constant as measured between two leads. For the Delta winding, the voltage measured between two leads is equivalent to the phase voltage, so no additional scaling of the back-emf constant is required.

Using Kirchoff's laws the relationship between current and voltage in each of the three motor phases is,

$$V_1 - V_2 = R \cdot i_1 + L \cdot \frac{di_1}{dt} + k_{emf} \cdot v \cdot phasing_1 \quad (3.6)$$

$$V_2 - V_3 = R \cdot i_2 + L \cdot \frac{di_2}{dt} + k_{emf} \cdot v \cdot phasing_2 \quad (3.7)$$

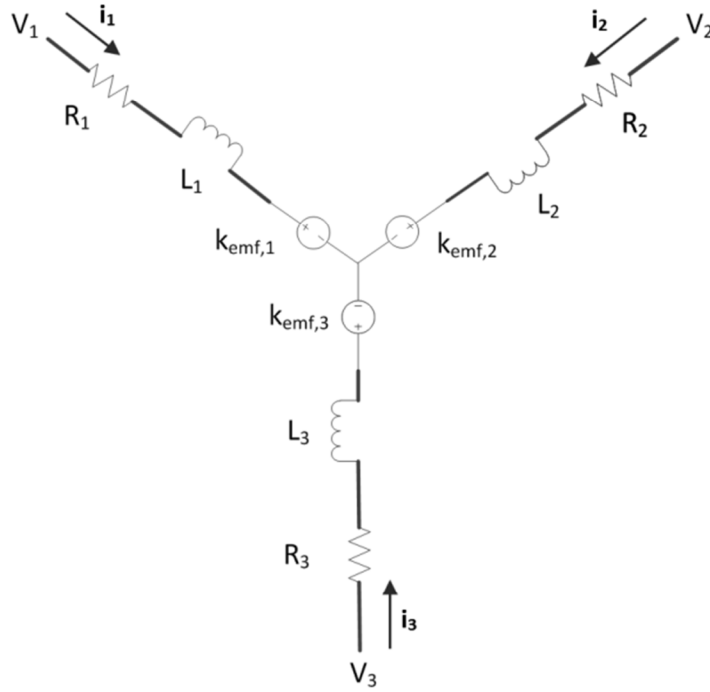
$$V_3 - V_1 = R \cdot i_3 + L \cdot \frac{di_3}{dt} + k_{emf} \cdot v \cdot phasing_3 \quad (3.8)$$

Figure 3.14 shows the graphical implementation of the three-phase Delta motor in Dymola. The inputs are the commanded currents in each of the three phases. The outputs are motor current and a force/position combination through the translational flange.

### Three-Phase Wye Wound Rotary Motor

The rotary motors driving the X-Rotate and Y-Rotate axes are three-phase Wye wound BLDC motors. A circuit schematic of a Wye wound motor is shown in Figure 3.15. As with the Delta wound motor, each phase of the motor has an associated





**Figure 3.15: Circuit Diagram for Three-Phase Wye Wound Brushless DC Motor.**

$$L = \frac{1}{2} L_{lead-to-lead} \quad (3.10)$$

In the Wye wound motor the phase voltage is not equal to the voltage between two leads. From the back-emf constant between two leads the phase back-emf constant is calculated as

$$k_{emf} = \frac{k_{emf,lead-to-lead}}{\sqrt{3}} \quad (3.11)$$

Kirchhoff's laws yield the relationship between current and voltage in each of the three motor phases and the voltage at the central node <sup>[11]</sup>,

$$V_1 - V_n = R i_1 + L \frac{di_1}{dt} + k_{emf} \dot{\theta} \quad (3.12)$$



$$V_2 - V_n = Ri_2 + L \frac{di_2}{dt} + k_{emf} \dot{\theta} \quad (3.13)$$

$$V_3 - V_n = Ri_3 + L \frac{di_3}{dt} + k_{emf} \dot{\theta} \quad (3.14)$$

$$i_1 + i_2 + i_3 = 0 \quad (3.15)$$

Figure 3.16 shows the graphical implementation of the three-phase Wye wound rotary motor in Dymola. The inputs are the commanded currents in each of the three phases, and the outputs are motor current and a torque/angle combination through the rotational flange.

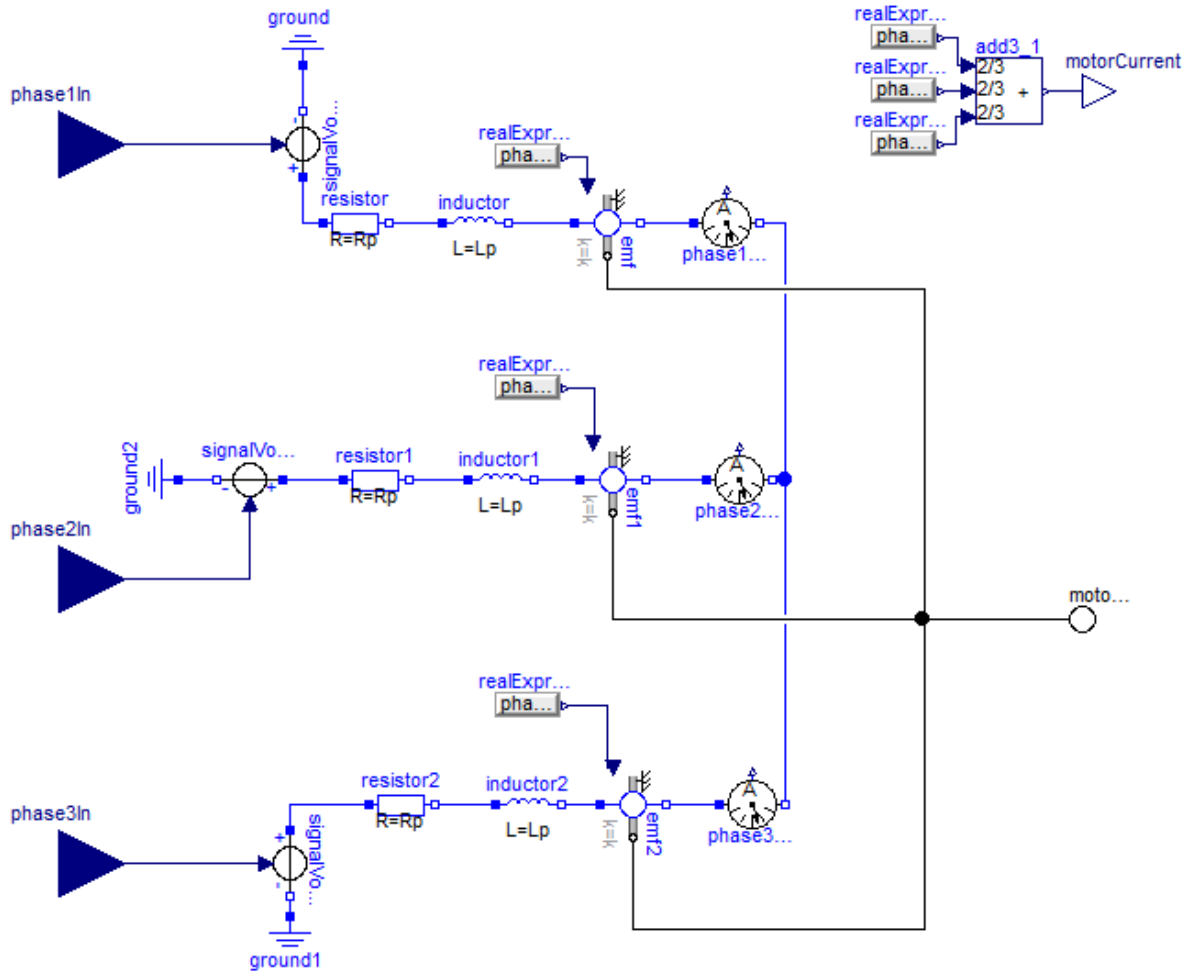


Figure 3.16: Three-Phase Wye Wound Brushless Rotary Motor Model in Dymola.

## Conclusions

Components from the Modelica Standard Library are used to create a rigid model of test mechanism in Dymola. The open-loop performance of the Dymola model is within  $2.7e-3\%$  of the mathematical model in Simulink for all motion axes. This verifies not only the construction of the model but also the use of the Dymola-Simulink interface. Motor models are similarly developed in Dymola from standard library parts. Motion and motor controller models are generated in Simulink. These models will be combined in the following chapter to predict the closed-loop performance of the test mechanism.

## **CHAPTER 4**

### **SIMULATED RIGID CONNECTION MODEL AND EXPERIMENTAL RESULTS**

The closed-loop response of the Dymola plant is verified with the mathematical model implemented in Simulink. The simulated step response is then compared to measured data.

#### **Closed-Loop Dynamic Simulation**

##### **Closed-Loop Model Development**

The PMAC, Copley, motor and plant models are combined in Simulink to create a model of the closed-loop dynamics of the 3 DOF test mechanism. Figure 4.1 shows the closed-loop model in Simulink with the Dymola plant (combining the motor and mechanism models). The model with the Simulink plant is not shown as the only difference is the replacement of the Dymola plant block with the Matlab function mathematical plant block.

A positional command for each axis is generated by the input block shown in Figure 4.2. This block is designed to output a constant value, bi-directional step, random or swept sine input based on the value of a user-specified parameter. All commands are generated in encoder counts. The position command and load encoder feedback (and motor encoder for the Y-Rotate axis with dual-feedback) are input into the PMAC block which outputs a command current (Figure 4.1).

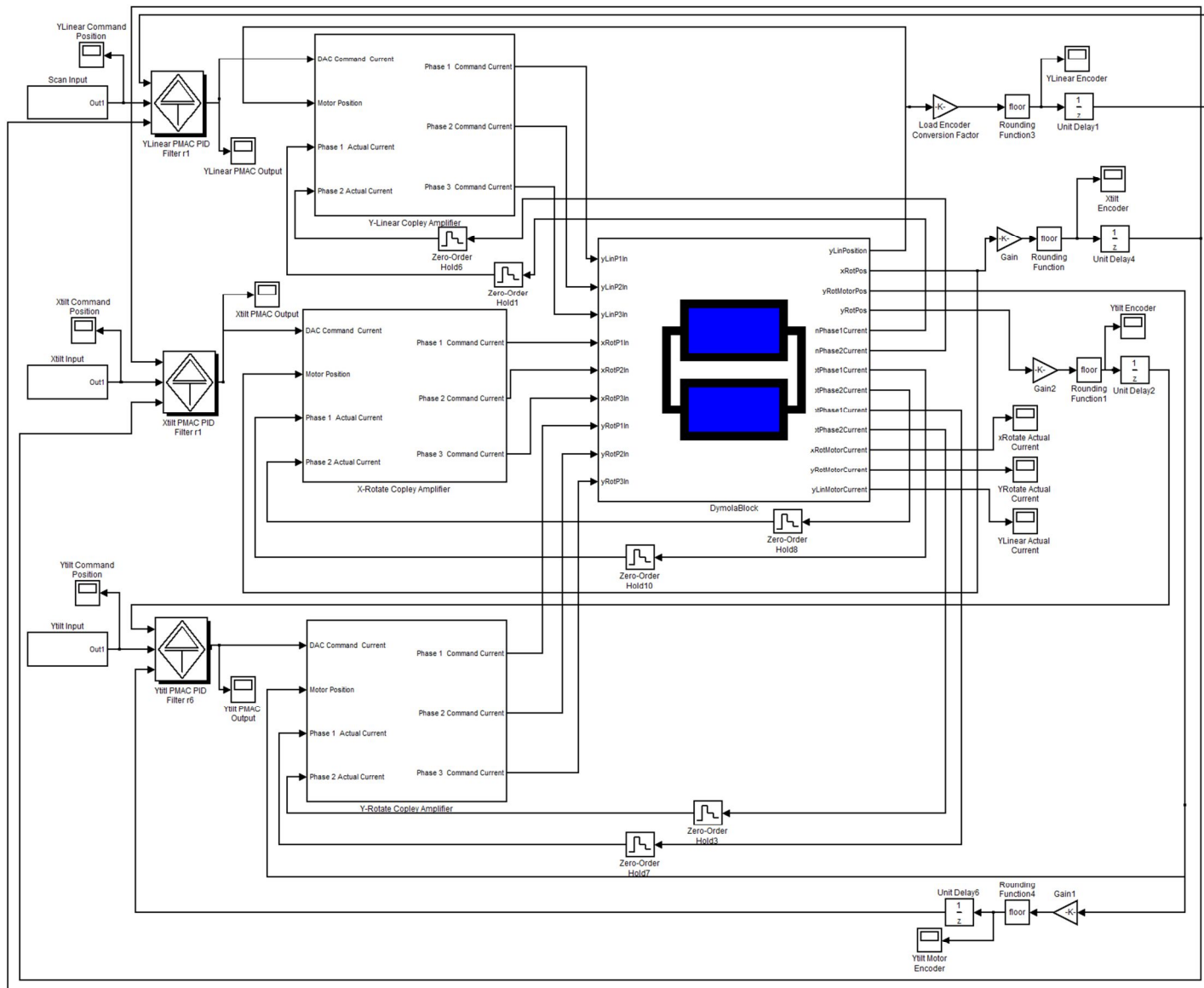
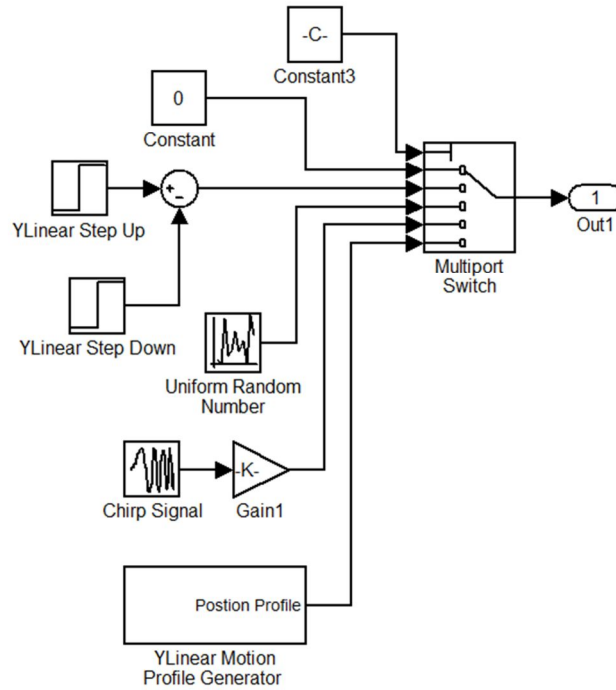


Figure 4.1: Closed-Loop Dynamic Simulink Model with Dymola Plant.

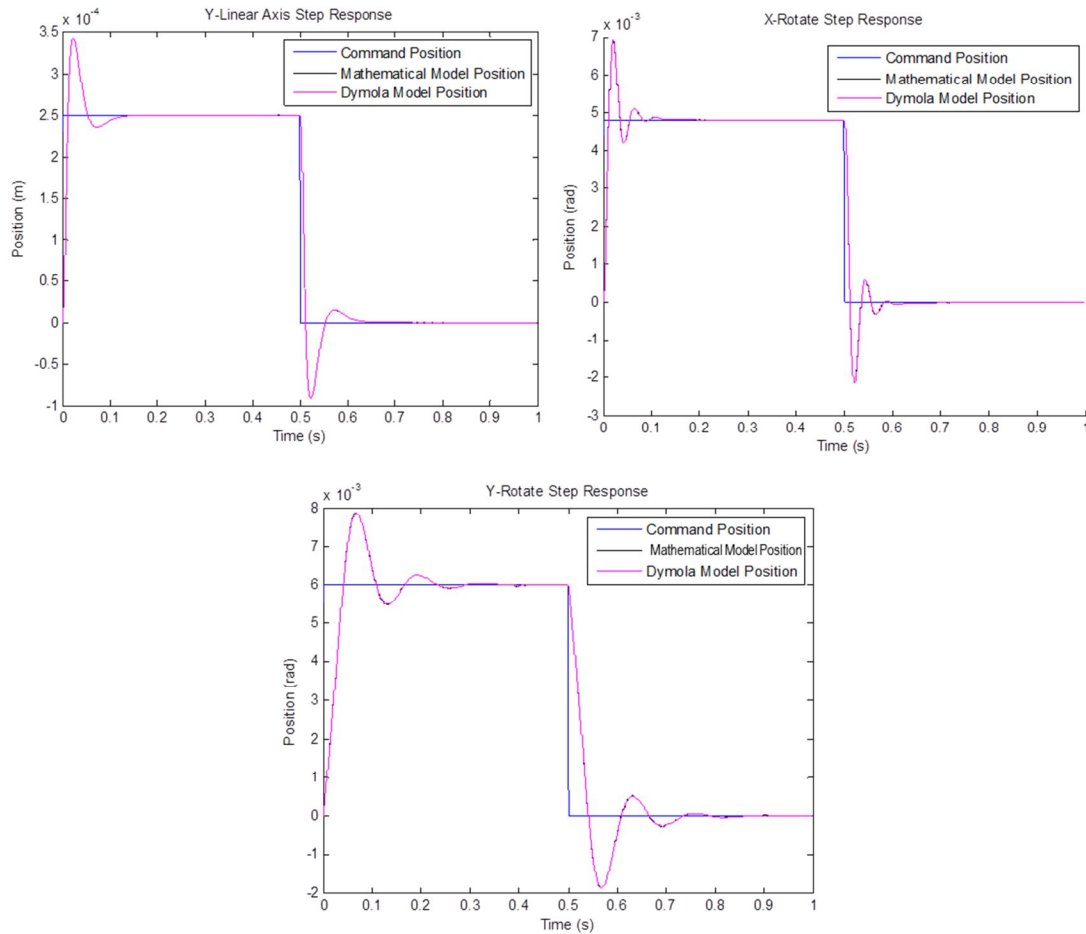


**Figure 4.2: Input Block Generates Position Command in Simulink.**

This command is held for one servo cycle using the zero-order hold block, ensuring that the motion command updates at the servo rate. The command is then passed to the Copley Amplifier along with motor velocity and phase current feedback. The Copley outputs current commands for each of the three motor phases at the current loop rate (the zero-order hold is included in the Copley subsystem). The motor phase current commands for each of the three motion axes are input into the Dymola plant, which contains both the motor and mechanical system dynamics. The Dymola plant outputs position and phase currents for each of the three axes. The position outputs are multiplied by their respective encoder conversion factors, rounded down to the nearest integer (since the encoder will only output position to the nearest count), and passed through a unit delay. The unit delay at the phase rate ensures that the encoder feedback is always from the previous communication cycle.

## Mathematical Verification

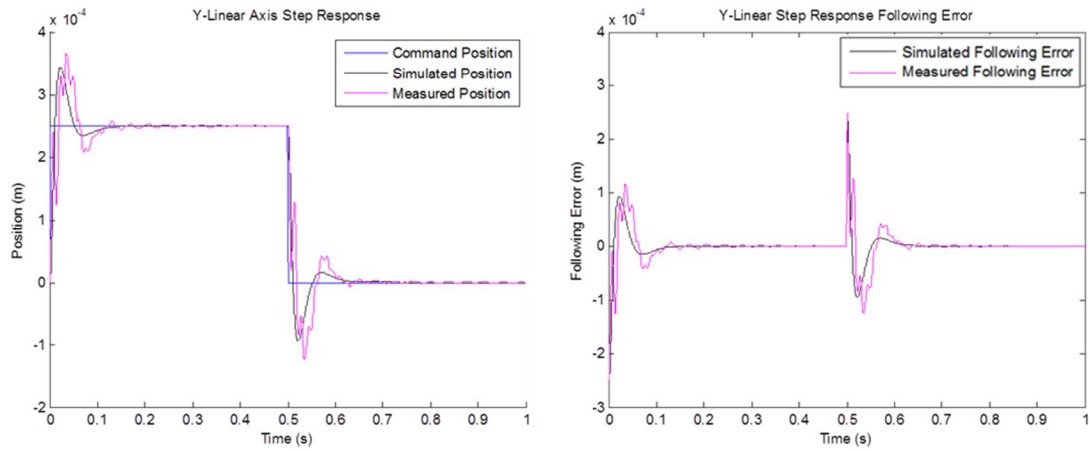
The response to a commanded positional step is simulated using both the Dymola and Simulink plant models. The step size is chosen to be as large as possible without saturating the control output. Figure 4.3 shows that the responses are approximately identical as expected.



**Figure 4.3: Closed-Loop Verification of Rigid Connection Plant. Top Left: Y-Linear Axis Top Right: X-Rotate Axis Bottom Center: Y-Rotate Axis. For all three axes, the simulated step responses of the Mathematical and Dymola Plants are approximately identical.**

## Experimental Validation

The same step inputs are commanded to the physical system and the responses measured for each motion axis. Figures 4.4 through 4.10 compare the simulated and measured responses.



**Figure 4.4: Simulated and Measured Y-Linear Step Response.**

Figure 4.4 shows the simulated and measured position response and following error for a positional step commanded to the Y-Linear axis. As expected, the rigid body plant captures the gross dynamics of the system; however, the oscillatory behavior at 75Hz observed in the measured response is not present in the simulation. This frequency does not align with the 37Hz frequency from the stiffness of cables connecting the payload to the counterweight, and is instead believed to be from the  $z$ -direction moment stiffness of the X-Rotate bearing which has a calculated first frequency of 76Hz. A more quantitative comparison of the simulated and measured responses is provided in Table 4.1. The percent difference is calculated by

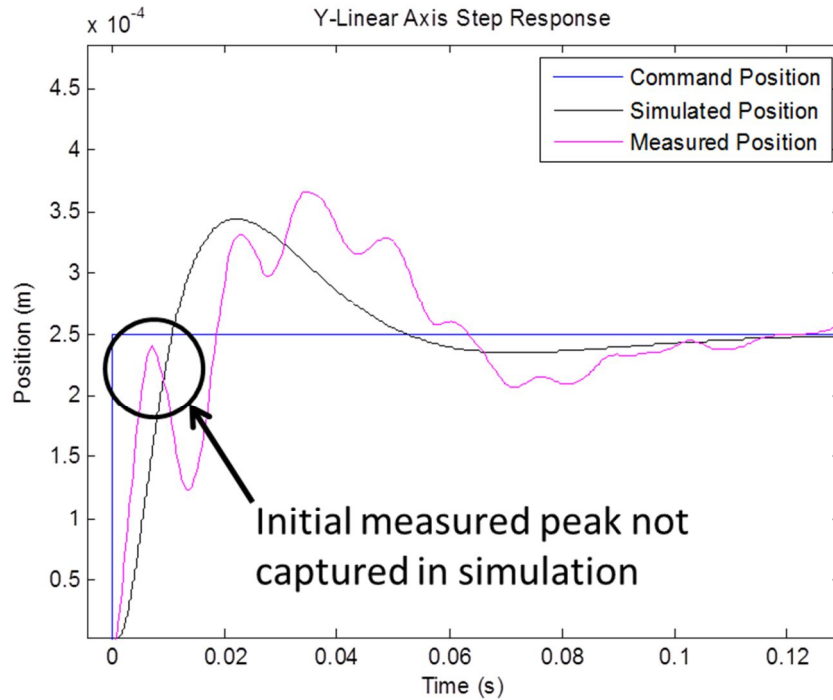
$$\text{Percent Difference} = 100 \frac{|\text{simulated} - \text{measured}|}{|\text{measured}|} \quad (4.1)$$

**Table 4.1: Comparison of Rigid Connection Simulation and Measured Y-Linear Step Response.**

<b>Parameter</b>	<b>Simulated</b>	<b>Measured</b>	<b>Percent Difference</b>
Delay Time (s)	6.263e-3	3.559e-3	76.97
Rise Time (s)	0.01080	0.01861	41.97
Peak Response (m)	3.439e-04	3.3667e-04	6.205
Peak Overshoot (%)	37.56	46.66	19.50
Peak Time (s)	0.02204	0.03455	36.22
Settling Time (s)	0.08479	0.09919	14.51

The delay time is the time for the response to reach 50% of the final commanded value. The rise time is the time for the response to rise from 0% to 100% of the final commanded value. The peak time is the time for the response to reach the peak of its overshoot. The maximum overshoot is the difference between the peak value and the commanded value as a percentage of the commanded value. The settling time is the time required for the response to reach and remain within 5% of the commanded value. The difference in peak response, peak overshoot and settling time between the simulated and measured responses are all under 20%. The delay time and rise time differ significantly between the simulation and the measurement. The simulated delay time is slower than the measured, while the simulated rise time is faster. This is because the simulated response does not capture the initial peak observed in the measured data (Figure 4.5). This initial peak is believed to be from the stiffness of the cables which are modeled as rigid in this simulation.





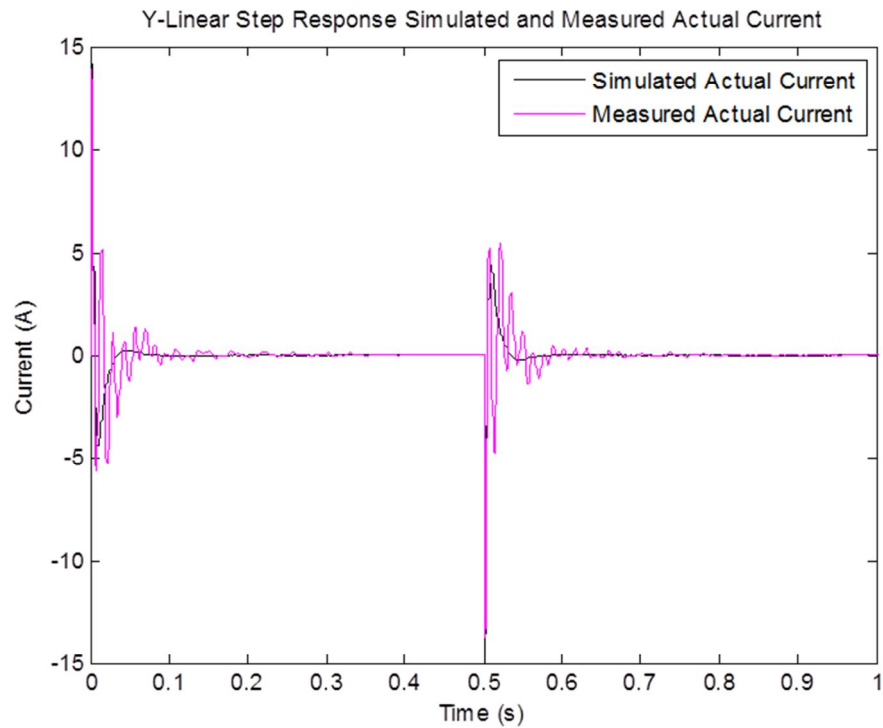
**Figure 4.5. Initial Peak on Measured Y-Linear Response Crosses Command.**

The peak time of the simulated response aligns reasonable well graphically with the first of the three largest amplitude measured peaks. However, because the second of these measured peaks has the largest amplitude, this is the peak that is used to calculate the peak time for the measured response, leading to a 36% difference between simulation and measurement. The overall root mean square (RMS) error between the simulated and measured responses is calculated by

$$RMS\ Error = \sqrt{\frac{\sum_{t=1}^{n_s} e_t^2}{n_s}} \quad (4.2)$$

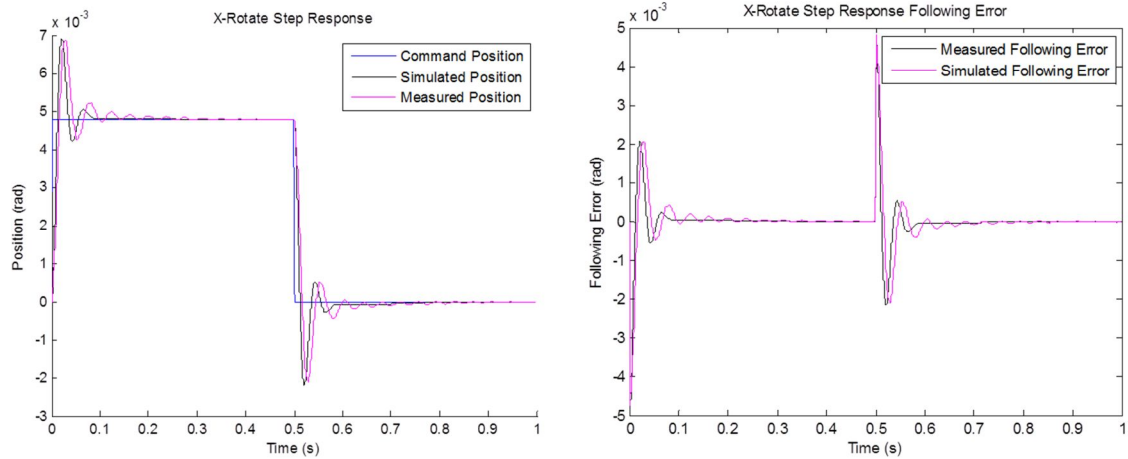
where  $e_t$  is the difference between the simulated and measured responses at time  $t$  and  $n$  is the total number of samples. For the rigid connection model the RMS error is 2.5594e-05m. This is 17.06% of the commanded step amplitude.

Figure 4.6 shows the simulated and measured current response to a position step on the Y-Linear axis. As with the positional response, there are oscillations in the measured current that are not present in the simulated current. These oscillations occur at a frequency of 75Hz.



**Figure 4.6: Simulated and Measured Step Response Current on Y-Linear Axis.**

A positional step input is applied to the X-Rotate axis, and the simulated and measured position and following error are shown in Figure 4.7. Graphically there is an offset in the phasing of the oscillations in the simulated and measured responses. However, the amplitude of the overshoot and the settling time agree well between the simulated and measured response. A quantitative comparison of the response characteristics confirms this observation (Table 4.2).



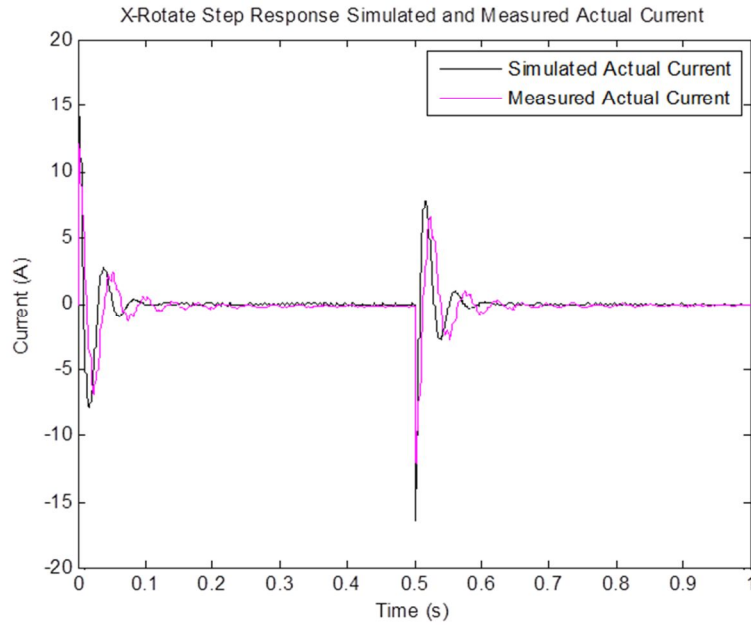
**Figure 4.7: Simulated and Measured X-Rotate Step Response.**

**Table 4.2: Comparison of Rigid Connection Simulation and Measured X-Rotate Step Response.**

Parameter	Simulated	Measured	Percent Difference
Delay Time (s)	7.425e-3	0.01107	34.95
Rise Time (s)	0.01196	0.01727	30.74
Peak Response (rad)	6.927e-3	6.879e-3	0.6969
Peak Overshoot (%)	44.50	43.50	2.299
Peak Time (s)	0.02037	0.02967	31.32
Settling Time (s)	0.07162	0.09121	21.47

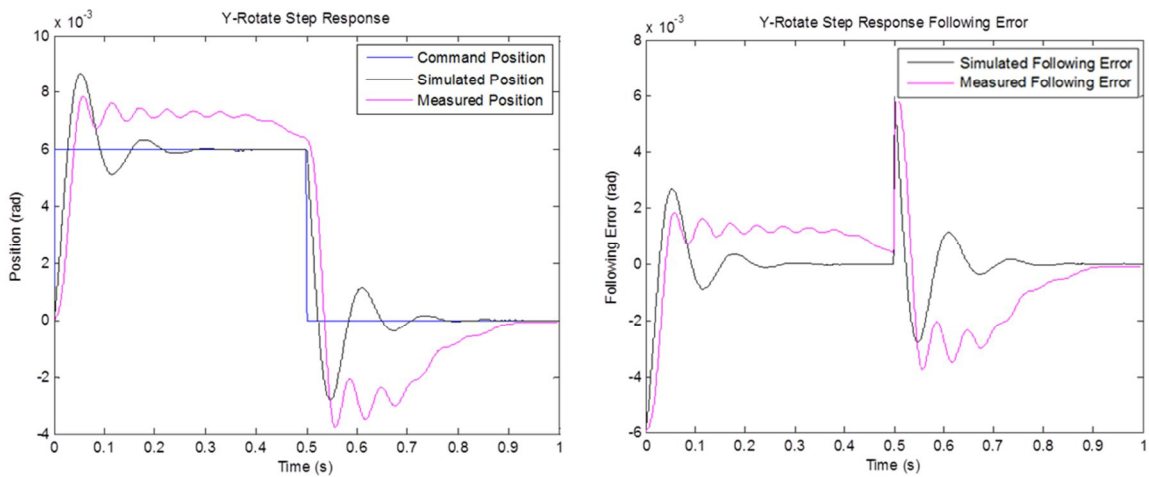
The percent difference between the simulated and measured peak overshoot and peak response are both under 3%. The delay, rise, and peak times all show a similar offset between the simulated and measured responses of 31-35%. This is reasonable given the visible timing difference, which may be due to the lack of any friction or damping in the simulated model. The simulated and measured settling times differ by 21%, with the simulation setting faster. This is likely because the simulated model is assumed to be inextensible and it includes no noise. The RMS error between the simulated and measured response is  $2.904e-4$  rad. This is 16.15% of the commanded step amplitude.

Figure 4.8 shows the simulated and measured current response to the same positional step. As in the positional response, the simulated system responds and settles faster than the measured.



**Figure 4.8: Simulated and Measured Current for X-Rotate Step Response.**

A positional step input is applied to the Y-Rotate motion axis with the simulated and measured position response and following error shown in Figure 4.9.



**Figure 4.9: Simulated and Measured Y-Rotate Step Response.**

The significant difference in response shape between simulation and measurement suggests that the dynamics of the gearbox and belt drive (not modeled in the rigid-body plant) are key contributors to the response of the Y-Rotate mechanism. The simulated response shows symmetric overshoot with minimal oscillatory behavior while the measured response shows significant oscillation. On the upward step the measured response has a lower peak value than the simulation and does not settle to the commanded position before the downward step begins. On the downward step the measured response has a larger peak amplitude than the simulation and settles to the commanded position. The directionality in the measured response may be related to the small step amplitude. With a smaller step size nonlinear effects and backlash may be more visible in the measured response. A quantitative comparison of the simulated and measured responses is provided in Table 4.3.

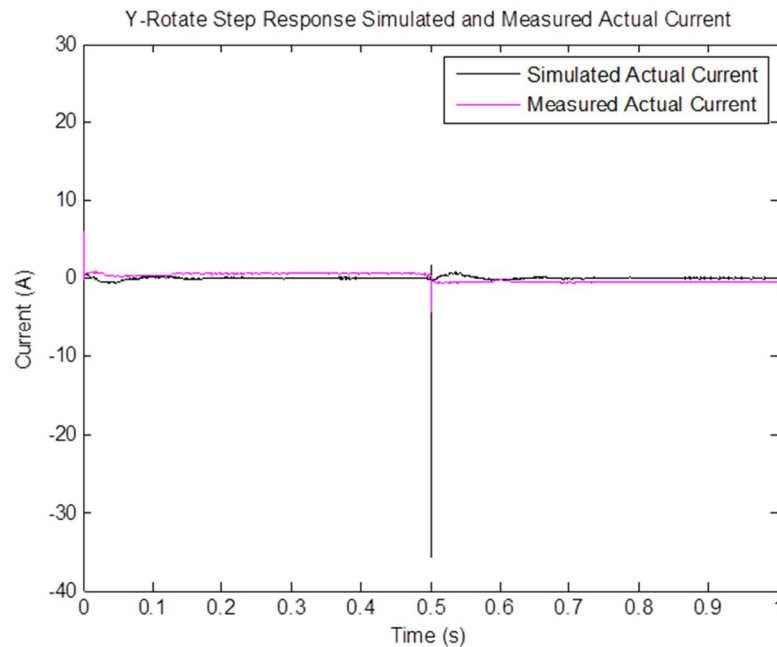
**Table 4.3: Comparison of Rigid Connection Simulation and Measured X-Rotate Step Response.**

<b>Parameter</b>	<b>Simulated</b>	<b>Measured</b>	<b>Percent Difference</b>
Delay Time (s)	0.01467	0.02745	46.55
Rise Time (s)	0.02762	0.04029	31.44
Peak Response (rad)	8.677e-3	7.838e-3	10.70
Peak Overshoot (%)	44.80	30.80	45.45
Peak Time (s)	0.05220	0.05623	7.175
Settling Time (s)	0.1918	0.5038	61.93

The table shows good agreement between the simulated and measured response peak time and peak response with differences under 11%. Slightly more variation is observed between the simulated and measured delay time, rise time and peak overshoot, but the most significant difference is the settling time, with a difference of more than 60%. The

RMS error for the y-rotate axis is 0.001163rad. This corresponds to 27.72% of the commanded step amplitude.

Figure 4.9 compares the simulated and measured Y-Rotate current response for the same positional step. A clear difference between the simulated and measured current is observed. The simulated current spikes to a significantly higher amplitude than the measured current at the onset of each positional step. In both cases the amplitude is high only for a short duration before returning to nearly zero.



**Figure 4.10: Simulated and Measured Current for Y-Rotate Step Response.**

## Conclusions

For the Y-Linear and X-Rotate axes the rigid connection model captures the gross dynamic response, with less than 20% error between the simulated and measured peak overshoot and settling time. In both cases, the errors are larger for rise time and delay time. The rise and delay time errors may be reduced in the case of the Y-Linear axis by

incorporating the stiffness of the cables. The RMS errors for both the Y-Linear and X-Rotate axes are under 18% of the commanded step amplitude.

The simulated Y-Rotate performance is visibly different from the measured response. Though the peak response and peak time agree within 11%, all other aspects of the response differ by at least 31% between simulation and measurement. The RMS error is 27.72% of the commanded step amplitude. This suggests the flexibility of the connections plays a key role in the response.

In the following chapter, flexible connections are added to the Dymola mechanism model. The simulated Y-Linear and Y-Rotate responses are expected to better reflect the oscillatory behavior observed in the measured response.

## CHAPTER 5

### PLANT MODEL GENERATION WITH FLEXIBLE CONNECTIONS

Models for flexible connections and damping are added to each axis. For the Y-Linear model stiffness and damping of the cables and bearing supports are included. For the X-Rotate model the stiffness and damping of the drive shaft and bearing support and the damping of the ferrofluid in the seal are added. For the Y-Rotate model gearbox torsional stiffness and damping are modeled, and the axial stiffness and damping of the drive belt, and the damping of the lip seals are added.

#### Modeling of Flexible Connections

##### Y-Linear Axis

In the Y-Linear axis, stiffness and damping are modeled for the counterweight cables and the linear bearing supports on both the payload and counterweight sides.

##### Cables

Cables can be modeled as a continuous structure or as a series of lumped stiffness, damping and mass elements. Yamamoto et al. <sup>[36]</sup> treat a cable moving around a pair of pulleys as a continuous string and model the free and forced vibratory response as the length between boundaries varies. The predicted response to forced vibration is compared to measured results for both the lengthening and shortening sides of the string with reasonable agreement <sup>[36]</sup>. Chi and Shu <sup>[3]</sup> create a lumped stiffness and mass model of an elevator hoist rope to investigate vertical (axial) vibration in response to harmonic forcing. The lumped-parameter model is compared to a continuous rope model, and good agreement between the fundamental frequencies of both models is demonstrated <sup>[3]</sup>. As



expected, for higher modes there is a significant difference between the lumped and continuous models. For this study a lumped stiffness and damping approach is used since the mass of the cable is assumed to be small. It is also assumed the cable bends fully around the pulley since the pulley diameter and cable tension are sufficiently large, and there is no slip between the cable and the pulley since the friction in the pulley bearing is sufficiently small. The cable is initially treated as two separate lengths each with a lumped stiffness and damping. The axial stiffness of each length is calculated by

$$k = \frac{EA}{L} \quad (5.1)$$

where  $E$  is the elastic modulus of the cable material,  $A$  is the cross-sectional area, and  $L$  is the length of the cable, which is dependent on the position of the payload. This positional dependence is not desirable, so the stiffnesses of the separate cable lengths are combined into an effective stiffness as springs in series

$$\frac{1}{k_{eff}} = \frac{L_1}{EA} + \frac{L_2}{EA} \quad (5.2)$$

where  $k_{eff}$  is the effective cable stiffness,  $L_1$  is the length of cable on the payload side, and  $L_2$  is the length of cable on the counterweight side. If the cable is assumed to be in contact over the top half of the pulley

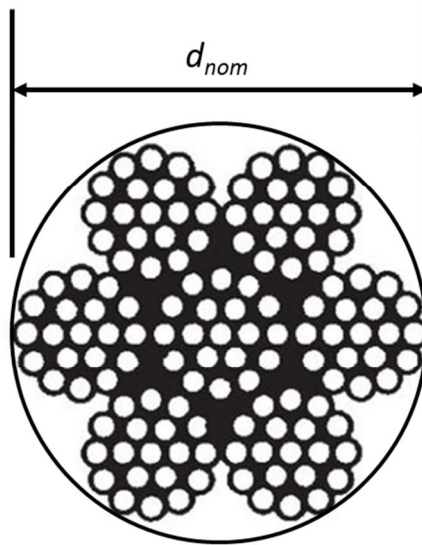
$$L_1 + L_2 = L_{cable} - \pi r_p \quad (5.3)$$

where  $L_{cable}$  is the total cable length between the payload and the counterweight and  $r_p$  is the pulley radius.

This is implemented in Dymola using a modified version of the Elastogap element from the Modelica Standard Library which only transmits force when the spring-damper is being compressed. This is useful for modeling contact surfaces where the two faces are

not rigidly connected <sup>[10]</sup>. The modification simply reverses the condition so force is transmitted only when the spring-damper is in tension (Appendix C). This models a cable with no compressive stiffness.

The Y-Linear axis uses cables with a 7x19 stranded construction which consist of 133 wires grouped into 7 strands of 19 wires laid concentrically about the center and helically wound. For stranded cables, calculating the area by  $A = \pi \left( \frac{d_{nom}}{2} \right)^2$  with nominal diameter,  $d_{nom}$ , overestimates the cross-sectional area of the cable. The difference in area is illustrated in Figure 5.1 for a stranded cable with a 7x19 construction.



**Figure 5.1. Cross-Section of a 7x19 Stranded Cable.**

An effective cross-sectional area (metallic area) or an effective modulus can be used to account for stranded cable construction <sup>[34]</sup>. In this application an effective area is used with the elastic modulus for the specific cable stainless steel alloy. The stiffness and damping input to the Dymola model are twice the calculated values to account for the combined effect of both cables.

## Damping

The three damping models used in this study to characterize losses and their Dymola implementations are discussed. A structural damping model is applied to the cables.

### *Damping Models*

Viscous damping is used when energy losses are proportional to velocity. One application of viscous damping is the movement of objects through a fluid. The viscous damping force,  $F_b$ , is characterized by

$$F_b = b\dot{x} \quad (5.4)$$

where  $b$  is the viscous damping coefficient and  $x$  is the displacement. Structural or hysteretic damping is applied when energy losses are proportional to displacement. Structural damping is typically used to characterize energy lost in the deformation of components <sup>[15]</sup>. In this study, structural damping is implemented using an equivalent viscous damping coefficient. For single degree-of-freedom systems, the equivalent viscous damping coefficient,  $b_s$ , can be approximated for a single frequency by

$$b_s = \frac{\eta k}{\omega} \quad (5.5)$$

where  $\eta$  is the loss factor,  $k$  is the stiffness of the single degree-of-freedom, and  $\omega$  is the frequency of approximation <sup>[15]</sup>. In this study the first natural frequency is used for all structural damping approximations. The third loss model is Coulomb damping which is proportional to normal force. Coulomb damping is often used to describe frictional interactions between surfaces. The Coulomb damping force  $F_c$  is calculated by

$$F_c = \mu N \text{sgn}(\dot{x}) \quad (5.6)$$

where  $\mu$  is the friction coefficient and  $N$  is the normal force.

### *Dymola Implementation*

In Dymola, viscous and structural damping (as an equivalent viscous coefficient) are implemented using the damper model from the Modelica Standard Library which generates a force proportional to the relative velocity between the input and output ports. Coulomb damping is implemented using the bearing friction model. This model assumes that the normal force is constant. The user inputs pairs of velocity and force values into a linear interpolation table for velocity values greater than or equal to 0. The negative velocity values are generated automatically by inverting the signs on both the velocity and the force. A multiplier can be used to increase the force at zero, simulating the maximum static friction force. This model includes stick-slip phenomena, so when the velocity is zero the model enters a stuck state and only produces the force that is required to prevent motion (up to the peak static friction). If the peak static friction is exceeded, the model then begins to move.

### *Structural Damping in Cables*

For the cable a structural damping model is used. The stiffness is the combined axial stiffness of both cables  $2k_{eff}$ , and the loss factor is determined based on the cable material to be 0.001<sup>[29]</sup>. This assumes there is no friction between the strands of the cable, and is expected to underestimate the measured damping value. The natural frequency  $\omega_n$  is calculated for the lumped payload and counterweight masses connected by a spring (Figure 5.2).

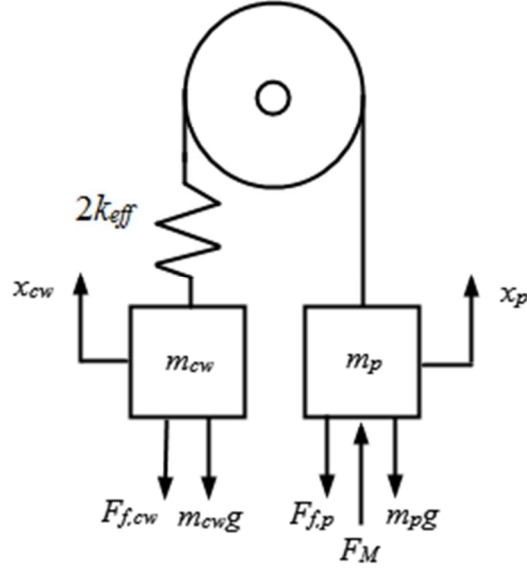


Figure 5.2. Diagram of Simple System for Calculation of Cable Natural Frequency.

The kinetic energy is the sum of the translational kinetic energies of the payload and counterweight masses

$$T = \frac{1}{2} m_p \dot{x}_p^2 + \frac{1}{2} m_{cw} \dot{x}_{cw}^2 \quad (5.7)$$

The potential energy is the energy stored in the spring

$$V = \frac{1}{2} k_{eff} (-x_p - x_{cw})^2 \quad (5.8)$$

The Lagrangian determined as in (2.25) and used with the Euler-Lagrange equation (2.26) to calculate the equations of motion. The generalized system coordinates are,

$$q_1 = x_p \quad (5.9)$$

$$q_2 = x_{cw} \quad (5.10)$$

and the generalized forces are

$$Q_1 = F_M - m_p g - F_{f,p} \quad (5.11)$$

$$Q_2 = -m_{cw} g - F_{f,cw} \quad (5.12)$$

The resulting dynamic equations are

$$m_p \ddot{x}_p - k_{eff}(-x_p - x_{cw}) = F_M - m_p g - F_{f,p} \quad (5.13)$$

$$m_{cw} \ddot{x}_{cw} - k_{eff}(-x_p - x_{cw}) = -m_{cw} g - F_{f,cw} \quad (5.14)$$

For modal analysis the applied forces are set to zero, and the natural frequencies are determined by finding the eigenvalues of

$$\begin{bmatrix} m_p & 0 \\ 0 & m_{cw} \end{bmatrix} \begin{bmatrix} \ddot{x}_p \\ \ddot{x}_{cw} \end{bmatrix} - \begin{bmatrix} k_{eff} & k_{eff} \\ k_{eff} & k_{eff} \end{bmatrix} \begin{bmatrix} x_p \\ x_{cw} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5.15)$$

The first eigenvalue is zero,

$$\omega_1 = 0 \quad (5.16)$$

corresponding to a rigid body mode where the masses move equal amounts in opposite directions.

$$[\varphi_1] = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad (5.17)$$

This frequency is not used since this mode does not change the relative length of the cables (spring). Assuming  $m_p = m_{cw} = m$ , the second eigenvalue is

$$\omega_2 = \sqrt{\frac{2k_{eff}}{m}} \quad (5.18)$$

This corresponds to a mode where the masses move in the same direction, thus changing the length of the cables.

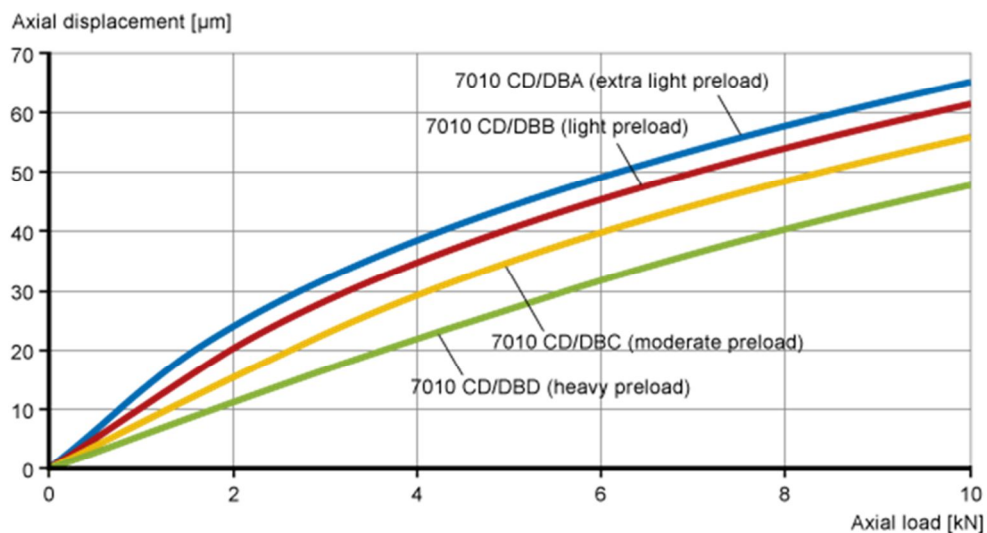
$$[\varphi_2] = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (5.19)$$

The second frequency is used in the calculation of structural damping.

## Linear Bearings

Petersen et al.<sup>[25]</sup> create a 2-dimensional model of a double row rotary ball bearing that includes the stiffness and damping of the support structure in two dimensions, the Hertzian contact stiffness and damping of each rolling element, and the measured high frequency resonant response as a pair of spring-mass-dampers in two orthogonal directions. The variations in stiffness and contact force as a function of position are investigated for different defect profiles in simulation, and the simulated vibratory response is compared to the measured response with good agreement.

A simplified approach is applied in this study. The high-frequency bearing resonant stiffness is assumed to be significantly higher than the stiffness of the bearing support (2 orders of magnitude in <sup>[25]</sup>), so this term is neglected. The Hertzian contact stiffness of the rolling elements and the support stiffness are lumped into a single stiffness parameter which is determined from the nominal load conditions and a supplier-provided bearing force-deflection diagram. A sample diagram for rotary bearing axial stiffness is included for reference (Figure 5.3).



**Figure 5.3. Sample Axial Force-Deflection Diagram for a Rotary Bearing. Image courtesy of SKF.**

The line colors distinguish different bearing preloads. A complete specification for a rotary bearing would also include similar diagrams for radial and moment loads. For linear bearings radial, reverse, transverse and moment stiffnesses in three directions are typically provided.

The nominal load in each direction is determined from static loading conditions (Figure 5.4). Assuming the payload and counterweight carriages are rigid, the reaction forces to the gravitational moment loads from the mass of the payload and counterweight can be distributed equally among the bearing blocks based on the horizontal and vertical bearing spacing<sup>[32]</sup>. This results in equal magnitude loads in the radial/reverse and transverse directions on each bearing block.

$$F_r = \frac{m g p_x}{2 b_y} \quad (5.20)$$

$$F_t = \frac{m g p_z}{2 b_y} \quad (5.21)$$

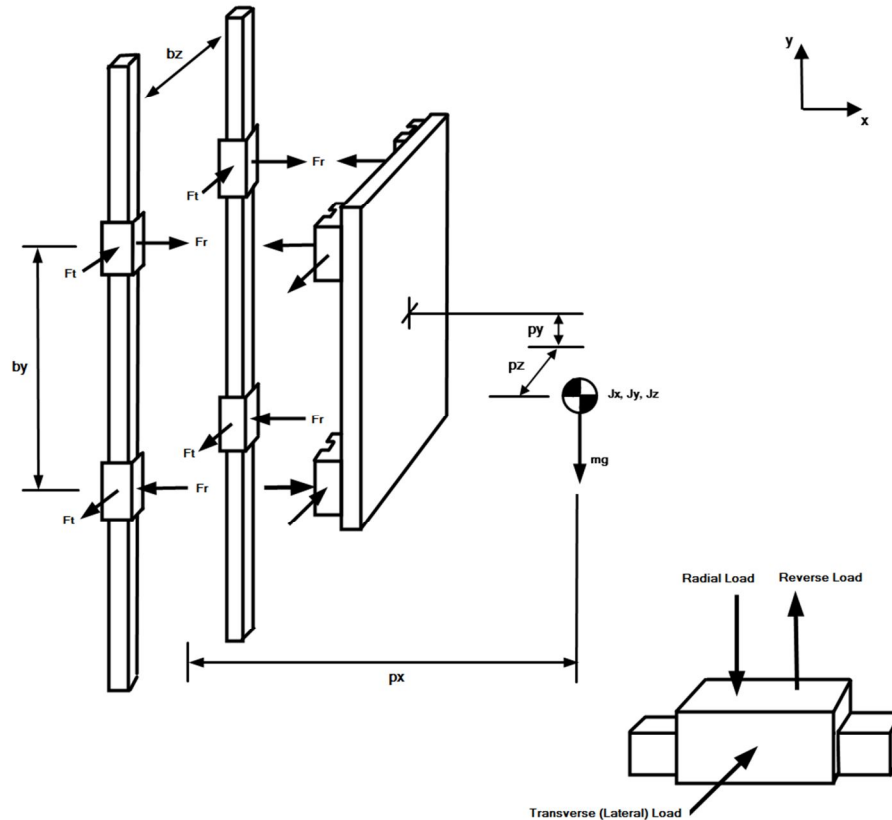
where  $F_r$  is the load in the radial/reverse direction,  $m$  is the mass of the carriage payload,  $p_x$  and  $p_z$  are the distance between the payload center of gravity and the bearing center in the  $x$  and  $z$ -directions respectively,  $b_y$  is the vertical bearing spacing, and  $F_t$  is the transverse bearing load.

A range around the nominal load  $F$  in each direction is used for a secant approximation of the nonlinear stiffness. The range is determined by

$$range = [(1 - \gamma)F, (1 + \gamma)F] \quad (5.22)$$

where  $\gamma$  is a value between 0 and 1 selected to characterize the variability in loading conditions. For this mechanism the primary source of load variation is from vertical motion, so  $\gamma$  is the peak vertical acceleration in g units.





**Figure 5.4. Static Bearing Loading on Y-Linear Axis.**

Because the blocks are connected by a rigid carriage, it is assumed only force loads can be applied to individual blocks, so no moment stiffnesses are included. The stiffness behavior of the bearings is overall nonlinear as a function of applied load, but this model uses a linear approximation over a load region of interest selected based on expected peak acceleration and deceleration. For large accelerations (on the order of 1g) the accuracy of the stiffness model is diminished since large changes in bearing loads occur.

For simplicity, in Dymola the linear stiffnesses at the individual bearing blocks are combined as moment stiffnesses in  $x$ ,  $y$  and  $z$  directions (Figure 5.5). This eliminates the change in stiffness when the bearing transitions from radial to reverse loading since for moment loading two blocks are always being loaded in the radial direction and two in the reverse. Rotary joints in the  $x$ ,  $y$  and  $z$ -directions are added at the bearing center.

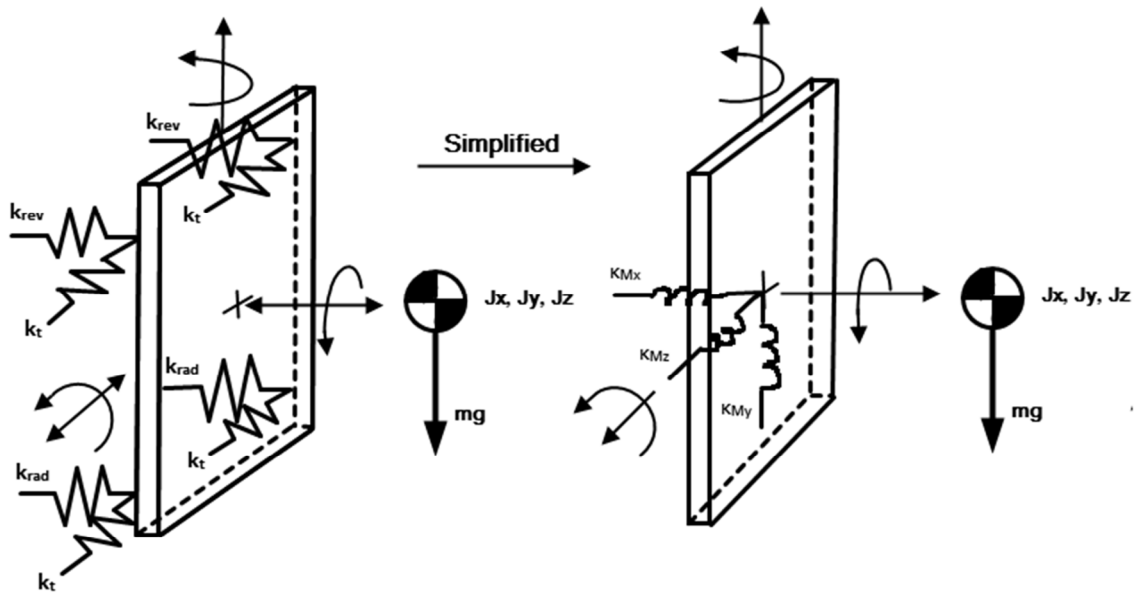


Figure 5.5. Simplified Bearing Y-Linear Stiffness Model. 8 translational stiffnesses are converted to equivalent moment stiffnesses about the x, y and z axes. The translational stiffnesses in the y and z directions are ignored as they do not directly couple into the motion directions.

Each joint has a spring-damper connecting the input to the output. The rotational stiffnesses are calculated by

$$\kappa_{Mx} = k_t b_y^2 \quad (5.23)$$

$$\kappa_{My} = \frac{(k_{rad} + k_{rev})}{2} b_z^2 \quad (5.24)$$

$$\kappa_{Mz} = \frac{(k_{rad} + k_{rev})}{2} b_y^2 \quad (5.25)$$

where  $\kappa_{Mx}$  is the rotational stiffness about the x-axis,  $k_t$  is the transverse bearing stiffness,  $b_y$  is the vertical bearing spacing,  $\kappa_{My}$  is the rotational stiffness about the y-axis,  $k_{rad}$  is the radial bearing stiffness,  $k_{rev}$  is the reverse bearing stiffness,  $b_z$  is the horizontal bearing spacing, and  $\kappa_{Mz}$  is the rotational stiffness about the z-axis. The combined translational stiffnesses in the x and z-directions are not included in this model. Since they do not

directly couple into any of the motion axes it is reasonable to assume they will have little effect on the closed-loop mechanism response.

Bearing losses are separated into contact losses from deformation of the rolling elements and frictional losses from the bearing lubricant and seals. The contact losses are modeled as a structural damping using a rotational damper in parallel with the torsional spring in the  $x$ ,  $y$  and  $z$ -directions. The loss factor is determined by the material of the rolling element to be 0.008 [29]. Because the three springs are orthogonal, the natural frequency,  $\omega$ , is calculated in each direction by

$$\omega = \sqrt{\frac{\kappa}{I}} \quad (5.26)$$

where  $\kappa$  is the rotational stiffness and  $I$  is the moment of inertia of the payload about the same axis.

Frictional losses are modeled in the direction of travel using a linear bearing friction element. A constant normal force is assumed, which is a reasonable when expected mechanism accelerations are small. The coefficient of friction, 0.003, and seal friction force are provided by the supplier, and the normal force used is the cumulative static normal force applied to the four bearing blocks. Two equal force values at different velocities are input into the interpolation table to define a constant magnitude force.

A diagram of the flexible connections added to the Y-Linear model is shown in Figure 5.6. Cable axial stiffness and damping are implemented as modified Elastogap element which transmits force only when in tension. The bearing  $x$ ,  $y$  and  $z$ -direction moment stiffness and equivalent damping on the payload and carriage side are implemented as 3 rotary spring-damper elements in parallel with 3 rotary joints.

Bearing friction in the direction of travel for the payload and the counterweight are implemented with linear support friction elements.

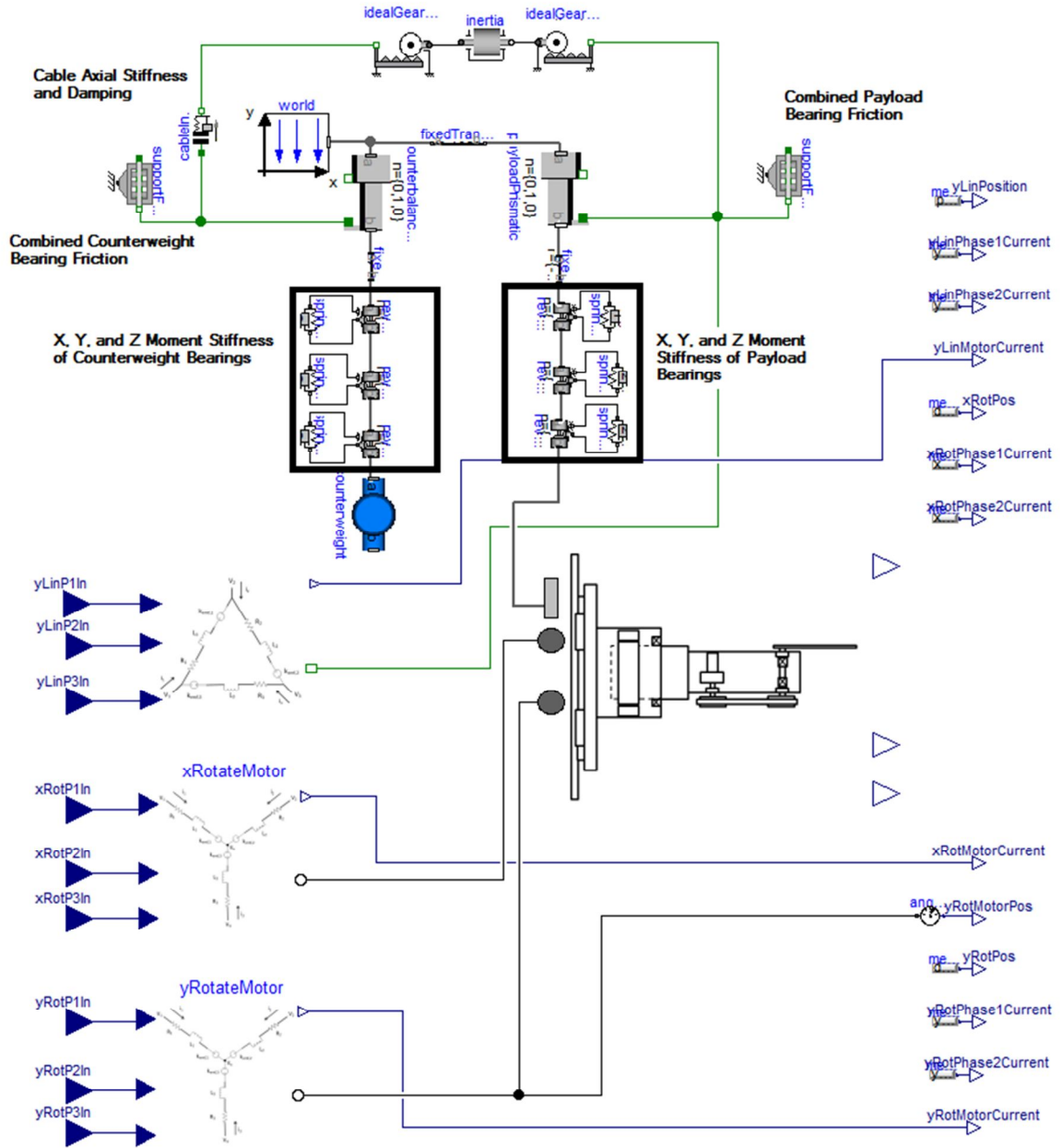


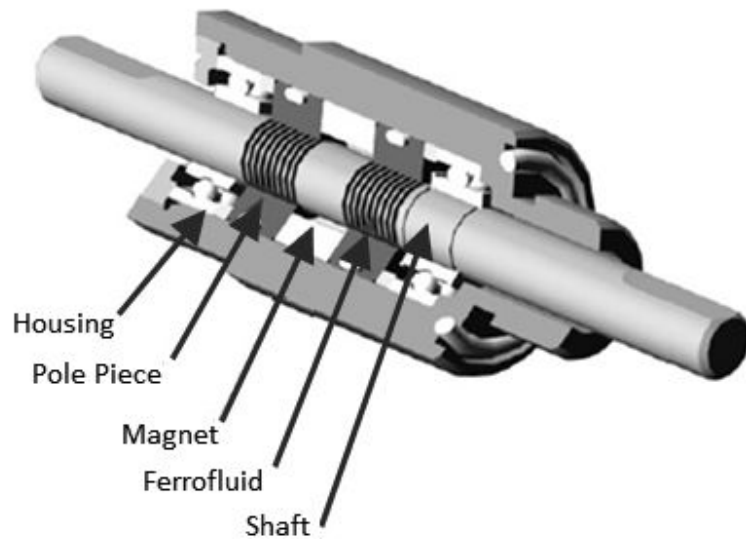
Figure 5.6. Dymola Model of Y-Linear Axis with Flexible Connections and Damping.

## **X-Rotate Axis**

The compliance in the X-Rotate axis comes from the ferrofluid seal. The compliant elements modeled are the drive shaft and the bearing support, and both are also sources of damping. The ferrofluid provides additional viscous damping.

### Ferrofluid Seal

A ferrofluid seal uses ferrofluid, a colloidal mixture of magnetic nanoparticles in a carrier liquid, held in place by a stationary magnetic field to create a seal between two environments, in this case vacuum and atmosphere. A typical ferrofluid seal design is shown in Figure 5.7.



**Figure 5.7. Typical Ferrofluid Seal Design. Ferrofluid seal comprised of a drift shaft, bearing supports and rings of ferrofluid held in place by permanent magnets. Image courtesy of Sealing Technologies.**

For modeling purposes, the seal is divided into three key elements: the drive shaft, the bearing support, and the viscous effect of the ferrofluid.

### *Drive Shaft*

The torsional rigidity of the drive shaft is given by

$$\kappa_s = \frac{GJ}{L} \quad (5.27)$$

where  $G$  is the modulus of rigidity of the shaft,  $L$  is the length, and  $J$  is the second moment of area calculated as

$$J = \frac{\pi(d_o^4 - d_i^4)}{32} \quad (5.28)$$

where  $d_o$  is the outer diameter and  $d_i$  is the inner diameter of the hollow round shaft. The hysteretic losses are modeled with an equivalent viscous damping (5.5) with a loss factor selected based on the shaft material to be 0.003<sup>[29]</sup>. The natural frequency is calculated by

$$\omega_{torsion} = \sqrt{\frac{\kappa_s}{I_x}} \quad (5.29)$$

where  $\kappa_s$  is the torsional stiffness of the drive shaft and  $I_x$  is the moment of inertia of the payload of the X-Rotate drive about the  $x$ -axis. The torsional rigidity and damping are implemented in Dymola using a rotary joint with a rotational spring-damper element connecting the input and output.

The bending stiffness of the shaft is calculated by

$$k_{sb} = \frac{3EJ}{L^3} \quad (5.30)$$

where  $E$  is the elastic modulus of the shaft and  $L$  is the length of the shaft. Again a hysteretic damping model is implemented by (5.5) with the natural frequency calculated from

$$\omega_{bending} = \sqrt{\frac{k_{sb}}{m}} \quad (5.31)$$

where  $k_{sb}$  is the bending stiffness and  $m$  is the mass of the X-Rotate axis payload. The bending stiffness and damping are implemented in Dymola with a prismatic joint that has a linear spring-damper element connection the input to the output.

### Cross-Roller Bearing

The static bearing reaction loads are calculated from the free body diagram in Figure 5.8. As with the linear bearings discussed above, lumped moment stiffnesses about the  $x$  and  $z$  axes and linear stiffness in the  $y$  (radial) direction are determined from a range around the static loading conditions from a supplier-provided chart of applied load and corresponding displacement in each direction.

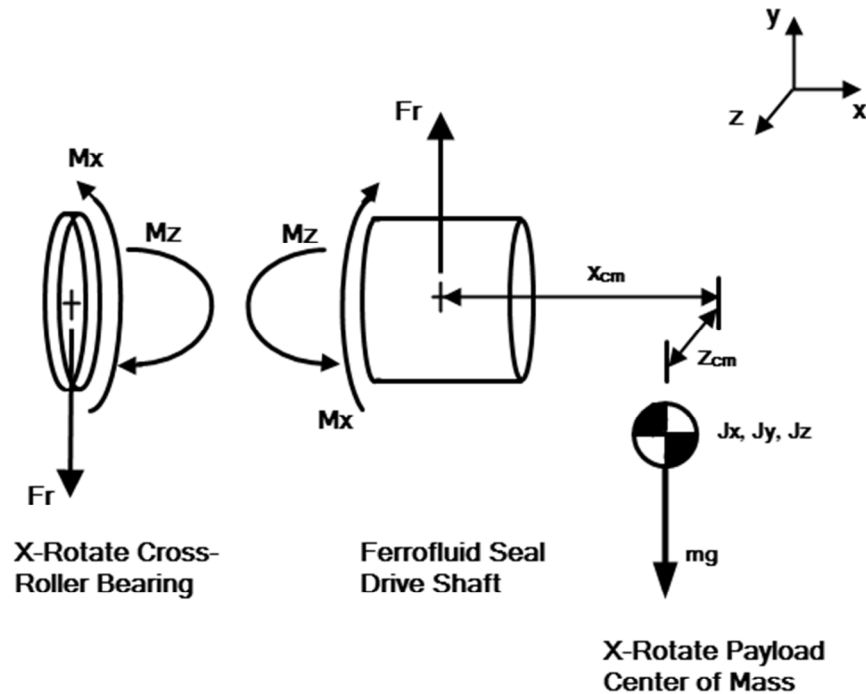


Figure 5.8. Static Loading on X-Rotate Cross-Roller Bearing. (Left) An equal and opposite reaction force in the  $y$ -direction and reaction moments in the  $x$  and  $z$  directions are applied to the bearing (Right) Gravitational force on the X-Rotate payload generates reaction loads at the bearing support location on the ferrofluid seal drive shaft.

The translational stiffness along the  $x$ -axis (axial stiffness) is not directly coupled to controlled motion directions, so it is not included in this model.

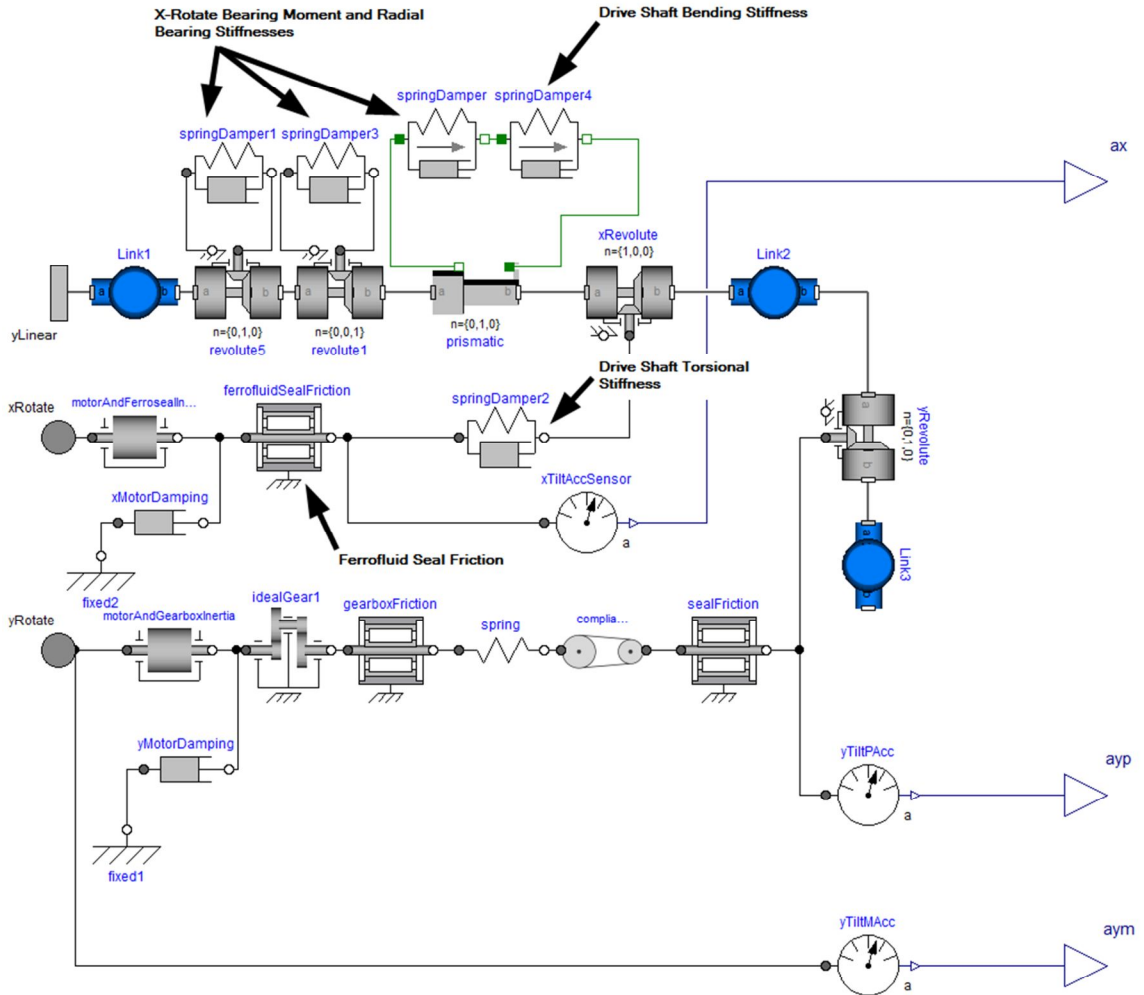
Damping is again separated into contact damping and frictional damping components. Contact damping is modeled as a structural damper applied in parallel with each stiffness. The loss factor is selected based on the roller material to be 0.001<sup>[29]</sup> and the natural frequencies are calculated as a simple lumped mass/inertia on orthogonal springs. Frictional damping is included through the bearing friction element with a coefficient of friction of 0.01 based on the bearing type. Cross-roller bearing friction is lumped with the ferrofluid drag torque and is discussed in the following section.

#### *Viscous Damping of Ferrofluid*

When ferrofluid is subject to a magnetic field the nanoparticles align in the direction of the field<sup>[26]</sup>. This increases the viscosity of the mixture since the fluid carrier is forced to flow around the nanoparticle chains. The viscous damping of the ferrofluid is a function of the viscosity of the carrier and the applied magnetic field. Pinho et Al.<sup>[26]</sup> generated a single degree-of-freedom viscous damping model for a ferrofluid seal used in a speaker as a function of shear rate, frequency, and a non-spatially uniform magnetic field. This model agrees well with experimental results for systems with approximately equal axial and radial thickness<sup>[26]</sup>. However, the generation of such models requires detailed knowledge of both the seal geometry and of the properties of the specific ferrofluid used in the seal. Since these details are proprietary, the supplier instead provided the viscous drag torque at a specified operating speed. The measured drag torque includes both frictional losses in the cross-roller bearing and viscous damping of the ferrofluid. This value was combined with the static friction from the cross-roller



bearing and input into a bearing friction element in Dymola. It should be noted that using the static friction of the bearing is a significant underestimate for cold-start conditions. When the seal is cold-started, the viscosity of the ferrofluid is initially very high, but it subsequently decreases during continuous operation due to viscous heating. The flexible connection elements for the X-Rotate axis are shown in Figure 5.9.



**Figure 5.9. Dymola Model of X-Rotate Axis with Flexible Connections and Damping.**

The stiffness and damping of the cross-roller bearing are implemented with spring-damper elements in parallel with rotary joints for the  $x$  and  $z$ -direction moment stiffnesses and a prismatic joint for the radial stiffness in the  $y$ -direction. The bending

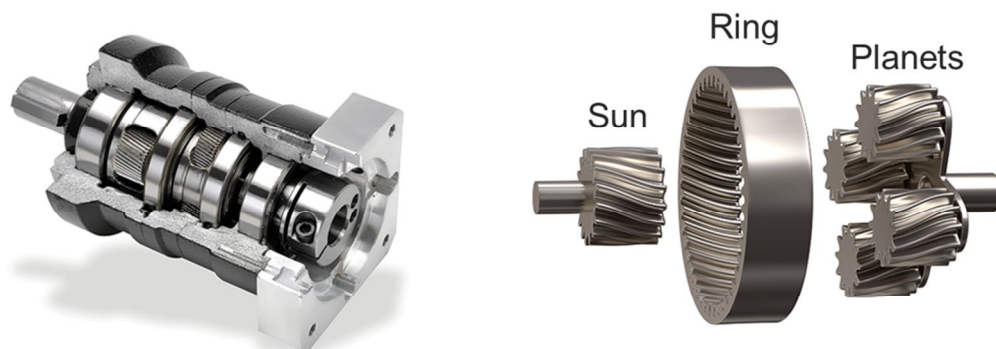
stiffness and damping of the drive shaft are added in series with the radial bearing stiffness. The torsional stiffness of the drive shaft is included with a rotary spring-damper in parallel with a revolute joint. Bearing and ferrofluid losses in the travel direction are modeled with a bearing support.

### **Y-Rotate Axis**

The Y-Rotate axis is dynamically the most complex with a planetary gearbox and belt in series. The torsional rigidity and frictional losses in the gearbox, the belt axial stiffness and structural damping, and the seal friction are modeled.

### **Two-Stage Planetary Gearbox**

A cutaway view of a two stage planetary gearbox is shown in Figure 5.10. The motor applies torque to the sun drive shaft which rotates the sun gear. The sun drives the rotation of the planets which in turn rotate the planet carrier as they traverse the outer ring gear. For a two-stage planetary gearbox, the carrier output shaft of the first stage is directly connected to the sun drive shaft of the second. Planetary gearboxes enable high-precision motion while generating a large output torques from high gear ratios.



**Figure 5.10. Two-Stage Planetary Gearbox. *Left:* Cutaway view of two stage planetary gearbox. *Right:* Exploded view of sun, ring and planets for a single stage. Images adapted from Machine Design.**

In the literature, lumped-parameter gearbox models typically include torsional mesh stiffness and damping and the torsional and bending stiffness and damping of the bearing supports <sup>[1], [7], [9], [17], [35]</sup>. For a planetary gearbox, bearing supports are included for the ring, carrier, sun, and each of the three planet gears. Meshing is considered between each of the three planets and the ring gear and between each of the three planets and the sun gear.

The torsional mesh stiffness is a combination of the gear body stiffness of each gear, the stiffness of the meshing teeth, and the contact stiffness at the meshing locations <sup>[17]</sup>. All three stiffness values can be determined from a finite-element model of the gear set (if loading is known). The gear body and tooth stiffness depend only on gear geometry; however, the contact stiffness is also a function of the torque applied to the system. Some formulations do not include the contact stiffness as it is typically significantly larger than the body and tooth stiffness values, and it is more difficult to determine due to its torque dependence <sup>[17]</sup>.

The phasing of the mesh stiffness must also be considered. As a pair of gears rotates, the meshing teeth alternate between a single contact point and a pair of contact points (double contact) <sup>[17]</sup>. A double contact will have twice the contact area, resulting in double the stiffness. For a planetary gearbox, the phasing of the mesh stiffness at each meshing location should be considered.

Gearbox losses come from mesh friction, viscous losses from the gearbox lubrication, bearing friction (including losses from rolling and sliding, lubrication and seals), and shaft seal friction <sup>[31]</sup>. The mesh frictional losses are torque dependent. Viscous losses from the lubrication are dependent on angular velocity. Bearing friction can be

modeled as previously discussed. Shaft seal friction is a function of shaft diameter and seal temperature (additional detail in following section).

For the system of interest, modeling of the gearbox is particularly challenging because the dimensions of the internal gearbox components, including gear and tooth dimensions, are proprietary. The supplier instead provided a lumped torsional stiffness from the gearbox input to output and two values of frictional torque data with corresponding angular velocity under no load conditions. The torsional rigidity is implemented in Dymola using a rotational spring, and the frictional torque as a function of velocity is entered into a bearing friction element.

### Belt Drive

The polyurethane timing belt has steel tension members. In operation, there is a difference in tension between the two sides of the belt. The side entering the driven pulley has increased tension, and the side exiting the driven pulley has reduced tension. It is this reduction in tension that necessitates sufficient belt pretension to keep the teeth in good contact with the pulley and prevent slip. For the determination of belt stiffness, the total length of the belt  $L$  is divided into the lengths of the tight (increased tension) side  $L_2$ , and the length of the slack (reduced tension) side  $L_1$ . For rotary positioning applications  $L_1=L_2=L/2$ . The effective stiffness of the belt includes the axial stiffness of the belt and the meshing stiffness<sup>[14]</sup>. Because the two sides act in parallel the axial belt stiffness,  $k_a$ , is the sum of the tight side and slack side stiffness and can be calculated from the specific stiffness  $c_{sp}$ ,

$$k_a = \frac{c_{sp}W}{L_1} + \frac{c_{sp}W}{L_2} = \frac{4c_{sp}W}{L} \quad (5.32)$$

where  $w$  is the belt width. The mesh stiffness,  $k_m$ , is determined by

$$k_m = z_{mv} k_{tooth} \quad (5.33)$$

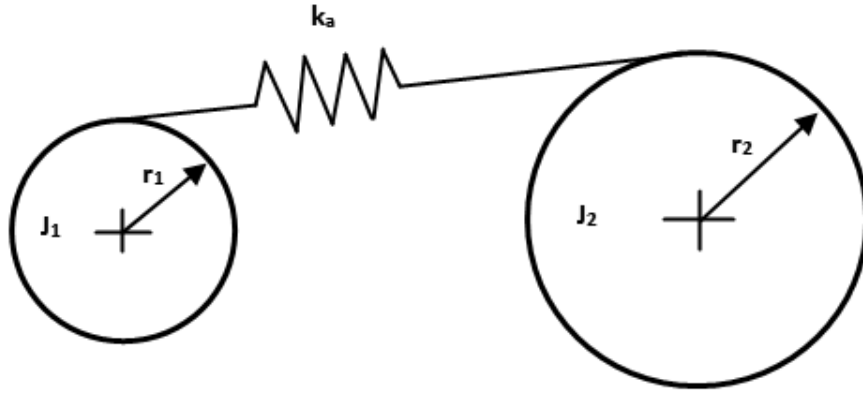
where  $z_{mv}$  is the virtual number of teeth in the mesh and  $k_{tooth}$  is the tooth stiffness <sup>[14]</sup>.

The virtual number of teeth is determined by a correction table based on the number of teeth in the mesh,  $z_m$ . The number of teeth in the mesh is

$$z_m = \frac{z_1}{180} \arccos\left(\frac{(z_2 - z_1)t}{2\pi a}\right) \quad (5.34)$$

where  $z_1$  is the number of teeth on the driven pulley,  $z_2$  is the number of teeth on the drive pulley,  $t$  is the tooth pitch, and  $a$  is the distance from drive pulley center to driven pulley center. The tooth stiffness can be estimated based on the tooth geometry using finite element analysis. Generally tooth stiffness is significantly larger than the axial belt stiffness, so the tooth stiffness can be neglected since the axial stiffness will dominate the response. The belt stiffness is modeled in Dymola as a linear spring between the two pulleys.

Belt losses are primarily from the structural deformation of the belt in the axial direction. The loss factor is determined from the belt material. A loss factor for the urethane belt material was not found, so a range of loss factors from 0.001 (belt stranding material) to 0.05 (rubber) were investigated and determined to have minimal effect on the model performance. The natural frequency was calculated from Figure 5.11.



**Figure 5.11. Simplified Belt Drive Model for Natural Frequency Calculation.**

The dynamic equations are derived from the Euler-Lagrange equation, set equal to zero, and the resulting eigenvalue problem is solved to determine the natural frequencies as in (5.7) through (5.15). The first frequency is

$$\omega_1 = 0 \quad (5.35)$$

This frequency corresponds to the rigid body motion of the two inertias in the same direction, the belt is not stretched. The second frequency is

$$\omega_2 = \sqrt{\frac{k_a J_1 r_2^2 + k_a J_2 r_1^2}{J_1 J_2}} \quad (5.36)$$

where  $J_1$  is the inertia of the Y-Rotate motor and gearbox,  $J_2$  is the inertia of the Y-Rotate payload,  $r_1$  is the drive pulley radius and  $r_2$  is the driven pulley radius. This frequency corresponds to the two inertias moving out of phase, deforming the belt. The second frequency is used for the calculation of structural damping which is modeled in Dymola as a damper in parallel with the belt axial stiffness.

## Lip Seal

Lip seals are commonly made from elastomers and plastics and use their geometry along with a radial spring to create an interference with the rotating shaft <sup>[13]</sup>.

Figure 5.12 shows a typical lip seal cross section. The interference between the seal and the shaft creates a frictional drag torque. The frictional force,  $F_f$ , at the shaft surface is

$$F_f = \mu F_N \quad (5.37)$$

where  $\mu$  is the coefficient of friction between the seal and the shaft and  $F_N$  is the normal force on the seal provided by the spring and the flexural stiffness of the seal material. The frictional drag torque  $\tau_F$  is then calculated by

$$\tau_F = r_s F_f \quad (5.38)$$

where  $r_s$  is the radius of the shaft. The frictional drag torque is combined with the bearing friction discussed in the following section and implemented in Dymola using a bearing friction element.

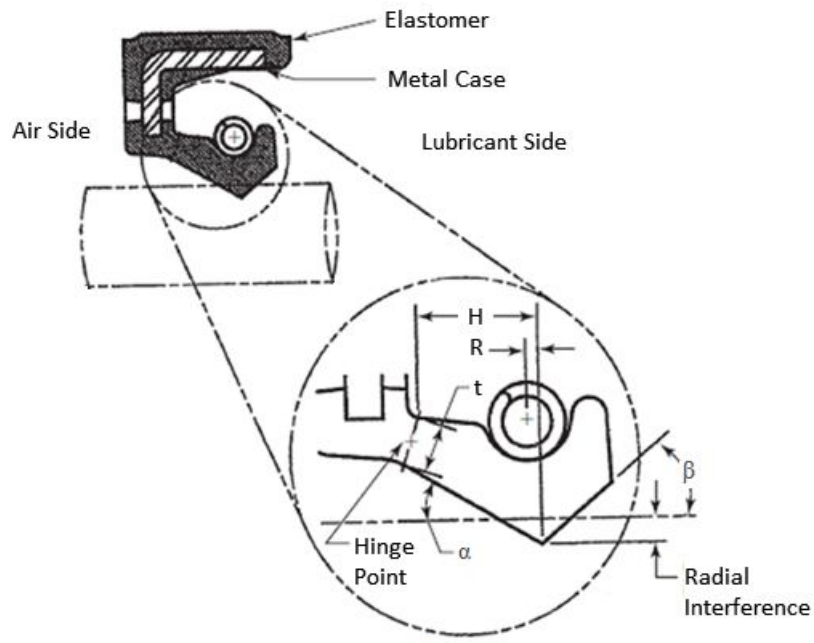


Figure 5.12. Typical Lip Seal Cross-Section. Image courtesy of Seals and Sealing Technology <sup>[13]</sup>.

## Bearing Friction

The friction of the Y-Rotate bearing supports is modeled as a Coulomb damping. The normal force at each bearing location is the resultant of the radial forces determined by the static bearing loading (Figure 5.13).

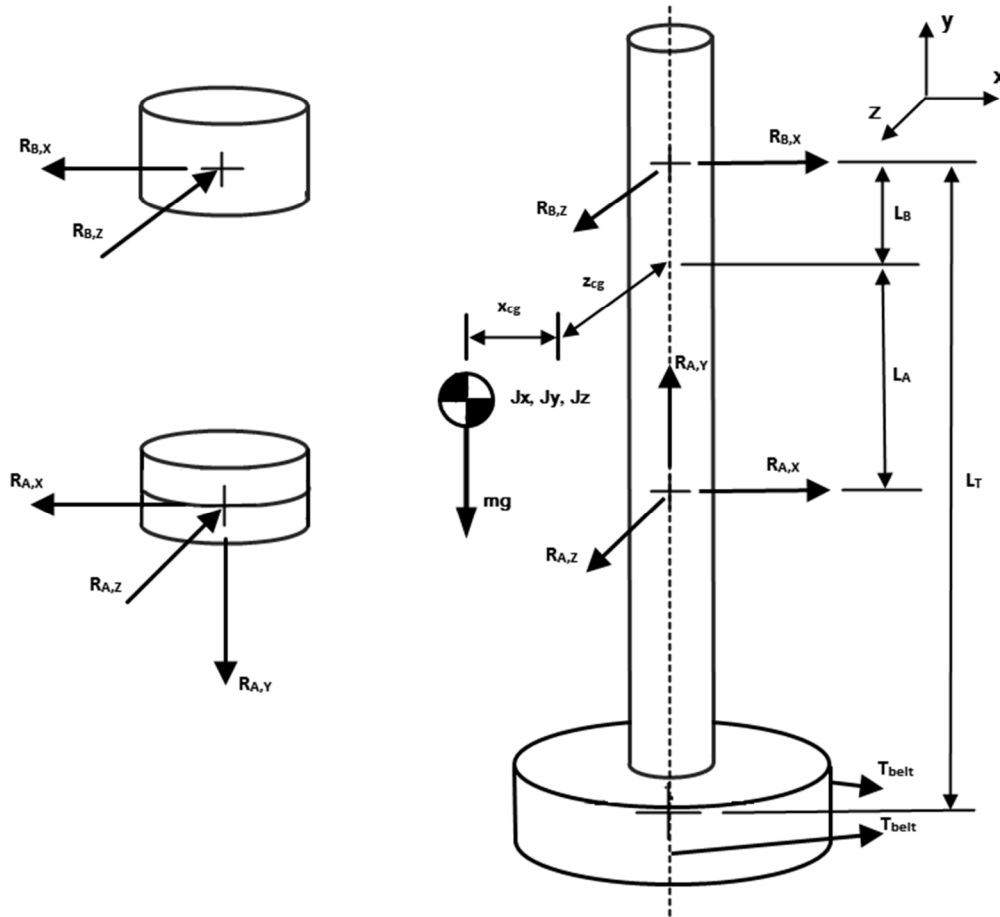


Figure 5.13. Y-Rotate Bearing Static Loading. *Left*: Reaction forces on needle bearing (top) and angular contact duplex pair (bottom). *Right*: Static load on drive shaft from Y-Rotate belt pretension and the mass of the bearing payload.

The radial forces on the angular contact duplex pair in the  $x$  and  $z$ -directions respectively are

$$R_{A,x} = \frac{x_{CG}mg + L_T T_{belt} \sin\beta}{L_A + L_B} \quad (5.39)$$



$$R_{A,Z} = z_{CG}mg - L_T T_{belt} \cos\beta \quad (5.40)$$

where  $x_{CG}$  is the offset of the CG from the bearing axis in the  $x$ -direction,  $m$  is the mass of the bearing payload,  $g$  is the gravitational acceleration,  $L_T$  is the distance from the center of the needle bearing to the center of the belt,  $T_{belt}$  is the belt tension force,  $\beta$  is the angle of the belt tension force relative to the pulley tangent vector,  $L_A$  is the distance from the center of gravity to the center of the duplex pair, and  $L_B$  is the distance from the center of gravity to the center of the needle bearing. The radial forces on the needle bearing in the  $x$  and  $z$ -directions are

$$R_{B,X} = -T_{belt} \cos\beta - R_{A,Z} \quad (5.41)$$

$$R_{B,Z} = T_{belt} \sin\beta - R_{A,X} \quad (5.42)$$

The bearing coefficients of friction are 0.0022 for the needle bearing and 0.0015 for the angular contact duplex pair.

The Y-Rotate Dymola model with flexible connections is shown in Figure 5.14. The gearbox is modeled as a gear ratio with the lumped torsional stiffness modeled as a rotary spring and the lumped gearbox friction modeled as a bearing friction. The belt drive converts rotary to linear motion and accounts for pulley radii using two IdealGearR2T elements, and the belt axial stiffness and damping are included as a linear spring-damper. The lip seal and bearing friction are combined in a bearing friction element.

## Conclusions

Flexible connections are included for each of the three motion axes. The Y-Linear axis incorporates cable and bearing stiffness and damping terms. Bearing friction is also considered in the travel direction. For the X-Rotate axis the bearing stiffness and

damping, the drive shaft stiffness and damping, and the combined frictional losses of the bearing and the ferrofluid are modeled. Motor viscous damping is included for both the X-Rotate and Y-Rotate axes. Updates to the Y-Rotate axis also include the gearbox torsional stiffness and lumped friction, the belt axial stiffness and damping, and the friction from the Y-Rotate bearings and the lip seals.

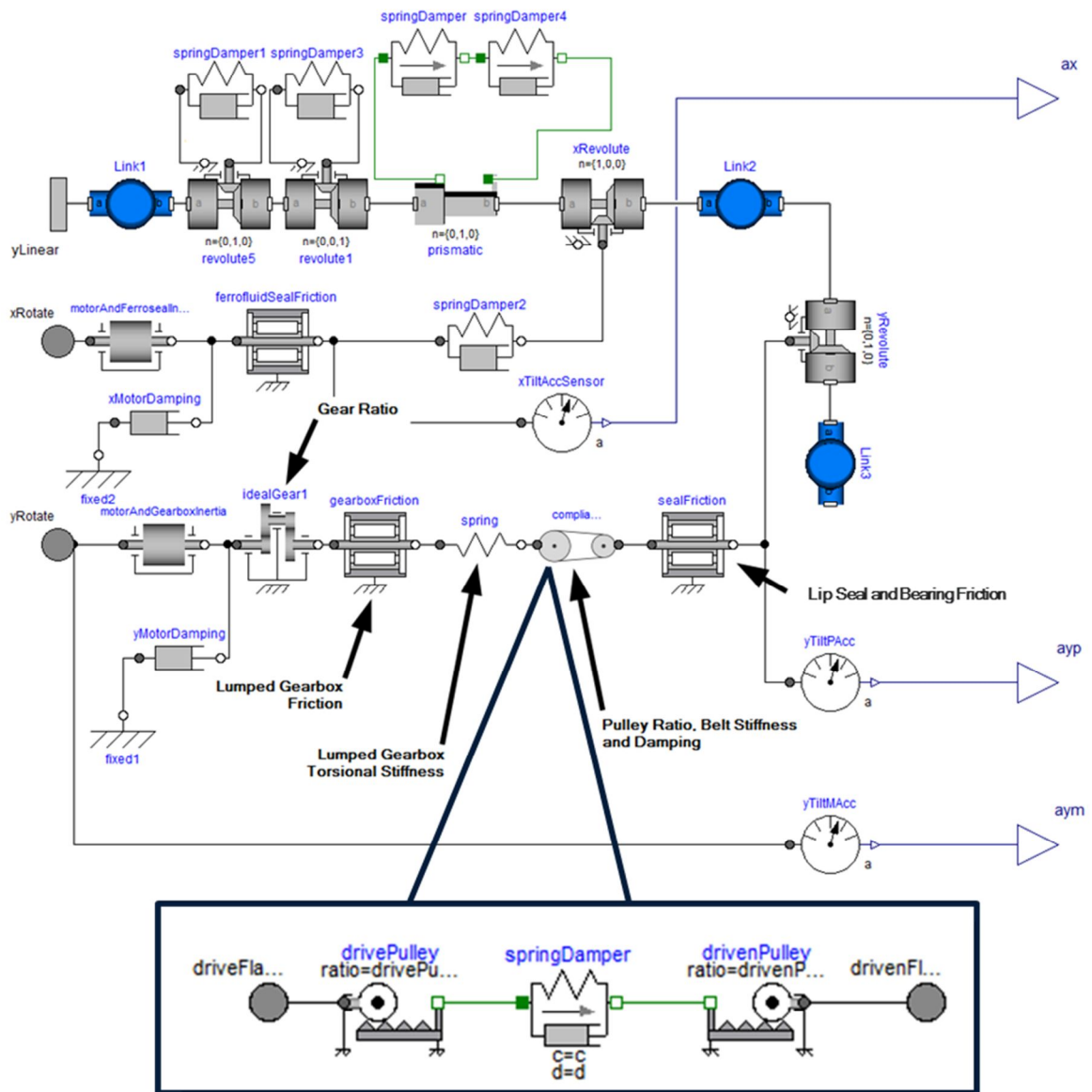


Figure 5.14. Dymola Model of Y-Rotate Axis with Flexible Connections and Damping.

## **CHAPTER 6**

### **SIMULATED FLEXIBLE CONNECTION MODEL AND EXPERIMENTAL RESULTS**

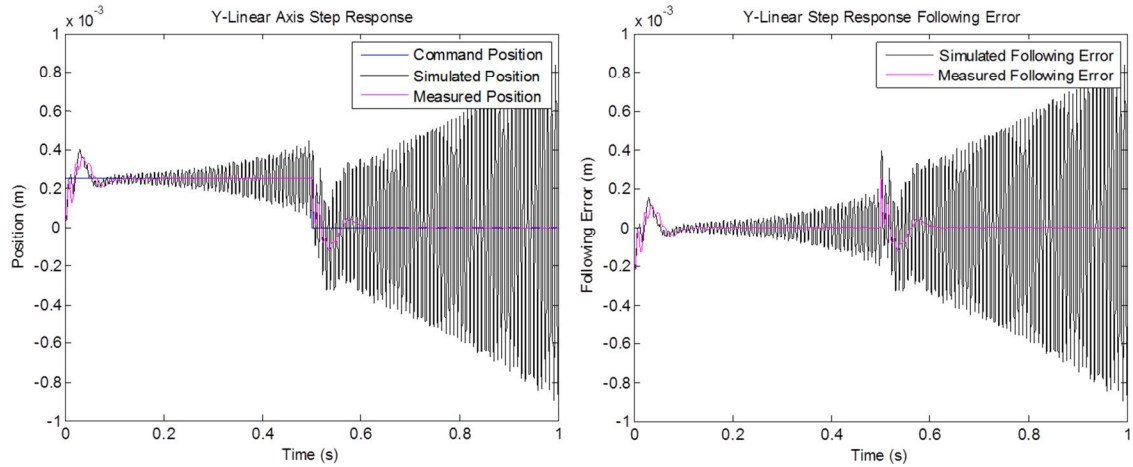
The simulated step response of each axis with flexible connections and damping is compared to the measured response. Calculated and updated damping values are used. The response with updated damping parameters is compared to the rigid connection response from Chapter 4.

#### **Experimental Validation**

The same positional step as in Chapter 4 is input into each of the three motion axes. When the calculated damping parameters are used, the responses for all three axes are unstable using the motion and motor controller tuning parameters from the physical system. The bearing damping is expected to be an underestimate since it includes only the contact losses. Additionally, the friction in the Y-Rotate axis gearbox is not well understood since values of loss torque are only provided for two steady-state velocities under no load conditions. These parameters are increased to stabilize the simulation and improve the correlation between the simulated and measured responses. The final model is compared to the rigid connection model to demonstrate improved performance.

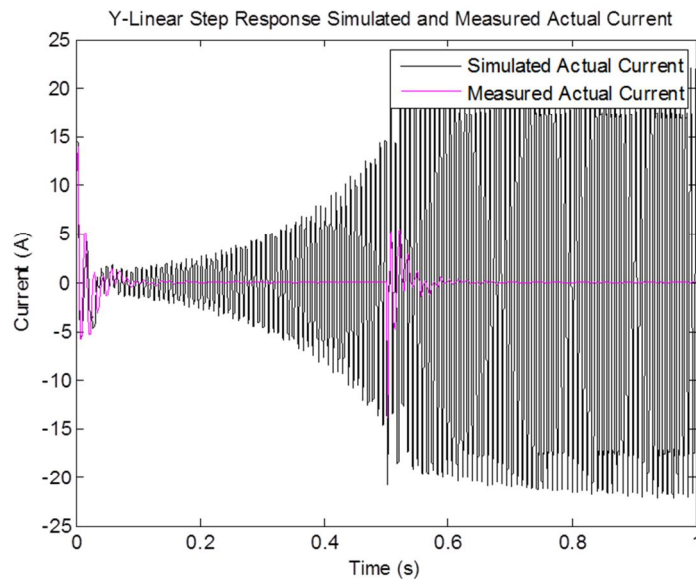
#### **Y-Linear Axis**

A positional step is applied to the Y-Linear axis. The simulated and measured position and following error are shown in Figure 6.1.



**Figure 6.1. Simulated and Measured Y-Linear Step Response with Calculated Damping Parameters.**

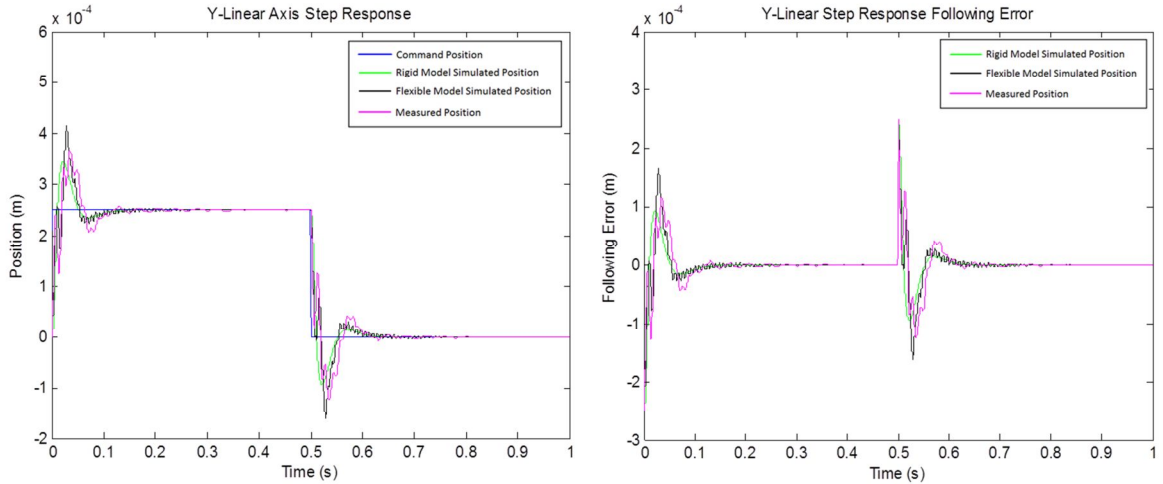
Oscillations in the simulated response begin in the step up and increase in amplitude through the remainder of the move. Similar oscillations are observed in the simulated current (Figure 6.2).



**Figure 6.2. Simulated and Measured Y-Linear Step Current Response with Calculated Damping Parameters.**

The addition of the bearings directly couples the responses of the three motion axes. Without damping similar to the physical system, this coupling leads to instability when using the experimentally determined controller gains. The calculated damping used in the counterweight, payload, and X-Rotate bearings is known to be an underestimate since it only accounts for contact damping not the effects of the lubricant or seals. The bearing damping is increased by 2.5x in the Y-Linear payload and counterweight bearings and 20x in the X-Rotate bearing, in both cases using an estimated lumped loss factor of  $\eta=0.01$ . Limited data on bearing loss factors is available, but this value seems reasonable based on the estimated bearing loss factor of 1% from <sup>[38]</sup>.

The friction at the Y-Rotate output (seal friction) is increased by 11.6x. This increase in friction accounts for additional gearbox friction (since the supplier data is provided for a no load condition) and additional belt drive losses. In reality these friction terms would be split between the gearbox and seal bearing friction elements, but because the relative friction distribution is not known they are applied as a lumped increase in frictional torque to the output. Frictional torque was measured at this location on the physical system to confirm that the modeled frictional torque is realistic, and the modeled value was found to be within 10% of the measurement. The step response of the model with improved damping is shown in Figure 6.3. The response of the rigid connection model is also included for comparison. With improved damping, the simulated step response is stable and tracks the measured response well through both the upward and downward steps. A quantitative comparison of the simulated and measured responses is shown in Table 6.1.



**Figure 6.3. Simulated and Measured Y-Linear Step Response with Improved Damping Parameters.**

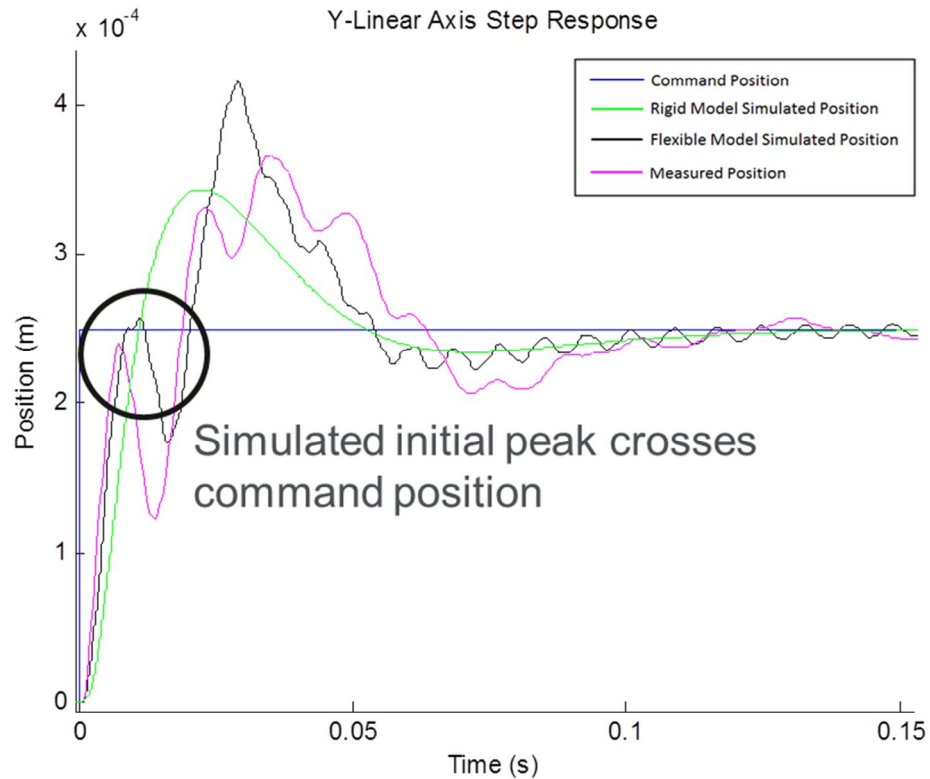
The simulated rise time, peak response, peak time and settling time are all less than 16% different from the measured values. The delay time is less than 25% larger than the measured, and the simulated peak overshoot is just under 35% larger. The RMS error between the flexible connection simulation and the measured response is 1.789e-5m, which is 11.93% of the amplitude of the commanded step.

**Table 6.1. Comparison of Flexible Connection Simulation and Measured Y-Linear Step Response.**

Parameter	Simulated	Measured	Percent Difference
Delay Time (s)	0.0044	0.0036	23.10
Rise Time (s)	0.0209	0.0186	12.44
Peak Response (m)	4.0715e-4	3.6665e-4	11.05
Peak Overshoot (%)	62.8600	46.6600	34.72
Peak Time (s)	0.0291	0.0346	15.72
Settling Time (s)	0.0904	0.0992	8.88

It should be noted that the simulated rise time is calculated from the third crossing (second peak) of the command position rather than the first. Both the simulated and

measured response have an initial peak, but the measured response amplitude is slightly smaller and does not reach the commanded value (Figure 6.4), so the second peak is used in both cases to calculate the rise time to ensure a more representative characterization of the simulated response.



**Figure 6.4. Initial Peak in the Simulated Y-Linear Step Response Exceeds the Commanded Position.**

The performance improvement between the flexible connection and rigid connection models is characterized by

$$Performance\ Improvement = \frac{100(\Delta_r - \Delta_f)}{\Delta_r} \quad (6.1)$$

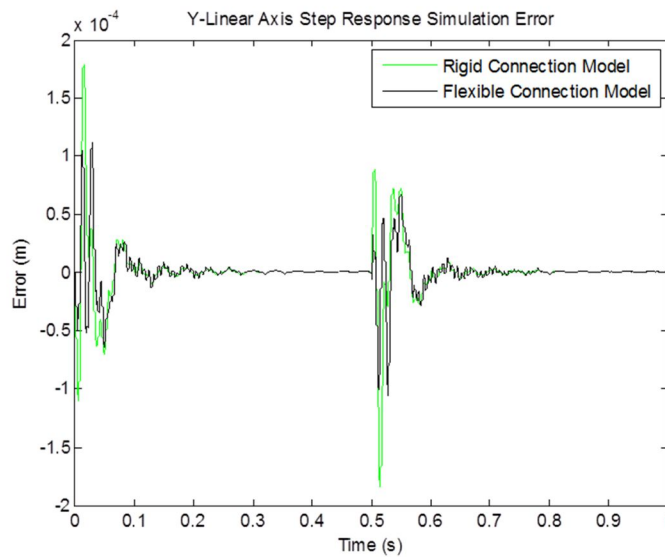
where  $\Delta_r$  is the percent difference between the simulation and measurement for the rigid connection simulation and  $\Delta_f$  is the percent difference between simulation and measurement for the flexible connection simulations. The rigid and flexible connection models are compared for the Y-Linear axis in Table 6.2. Prediction of delay time, rise

time, peak time and settling time are improved in the model with flexible connections. However, the peak response magnitude and peak overshoot are both less accurate in the flexible connection model. The RMS error is reduced by 30.07% with the addition of the flexible connections.

**Table 6.2. Performance Improvement from Flexible Connections in Y-Linear Step Response.**

Parameter	Rigid Connection (% Difference from Measured)	Flexible Connection (% Difference from Measured)	Performance Improvement (% Difference)
Delay Time	75.97	23.10	69.60
Rise Time	41.97	12.44	70.34
Peak Response	6.20	11.05	-78.02
Peak Overshoot	19.50	34.72	-78.02
Peak Time	36.38	15.72	56.79
Settling Time	14.51	8.88	38.82

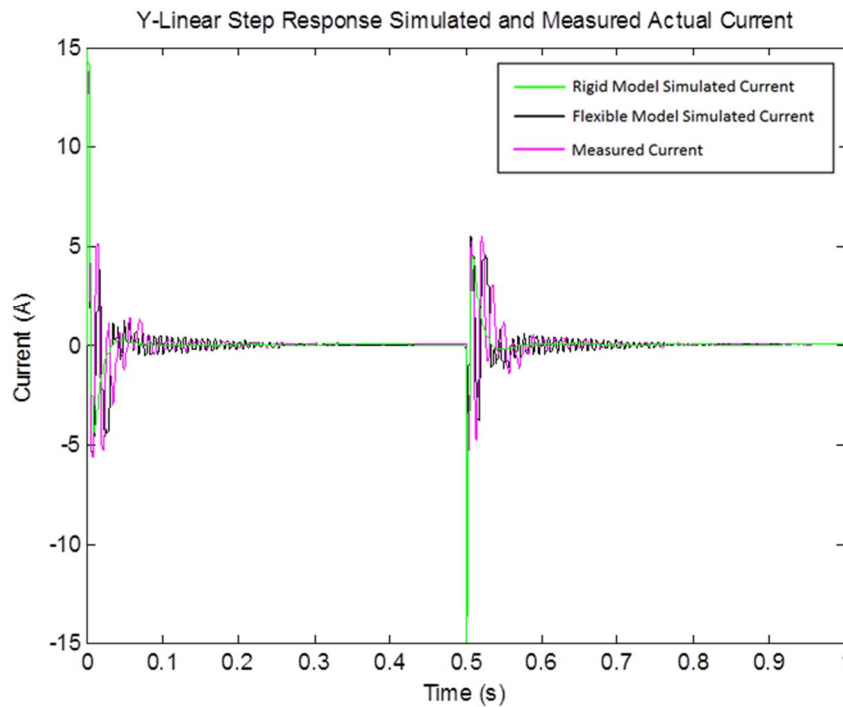
The simulation error, calculated as the difference between the simulated and measured values is shown for the rigid and flexible connection cases in Figure 6.5.



**Figure 6.5. Error Between Rigid and Flexible Connection Y-Linear Models and the Measured Response.**



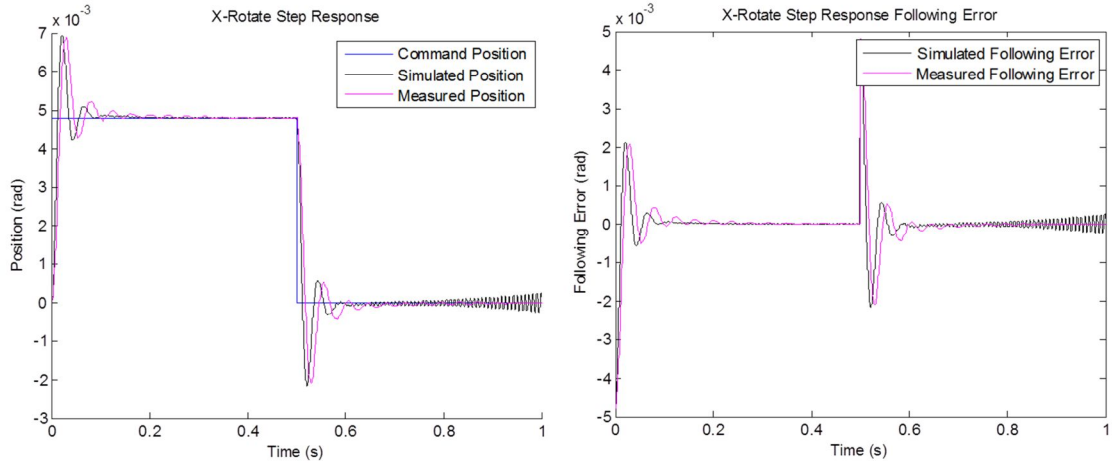
The flexible connection response shows smaller error in the first positive and negative peaks for each step direction. The simulated current response tracks the measured response well (Figure 6.6). Again the rigid connection simulation is overlaid for comparison. The current amplitudes are similar, but no oscillation is observed in the rigid-connection response.



**Figure 6.6. Simulated and Measured Y-Linear Step Response Current with Improved Damping Parameters.**

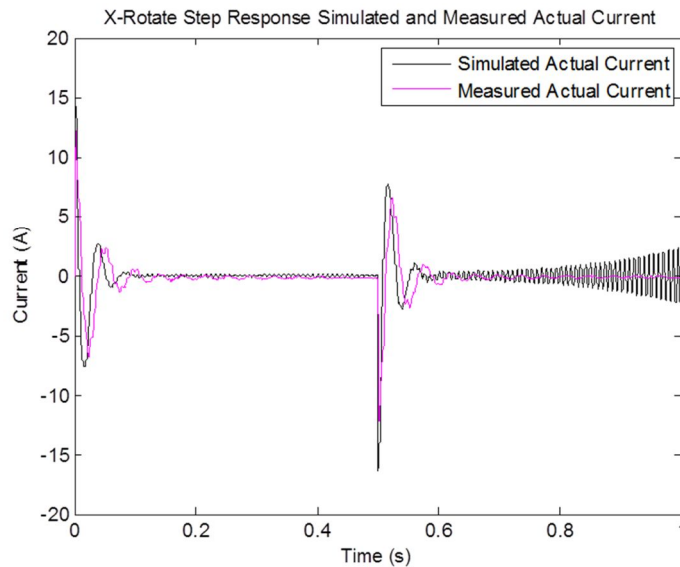
### **X-Rotate Axis**

A positional step is applied to the X-Rotate axis. The simulated response and following error with the calculated damping parameters are shown with the measured response in Figure 6.7.



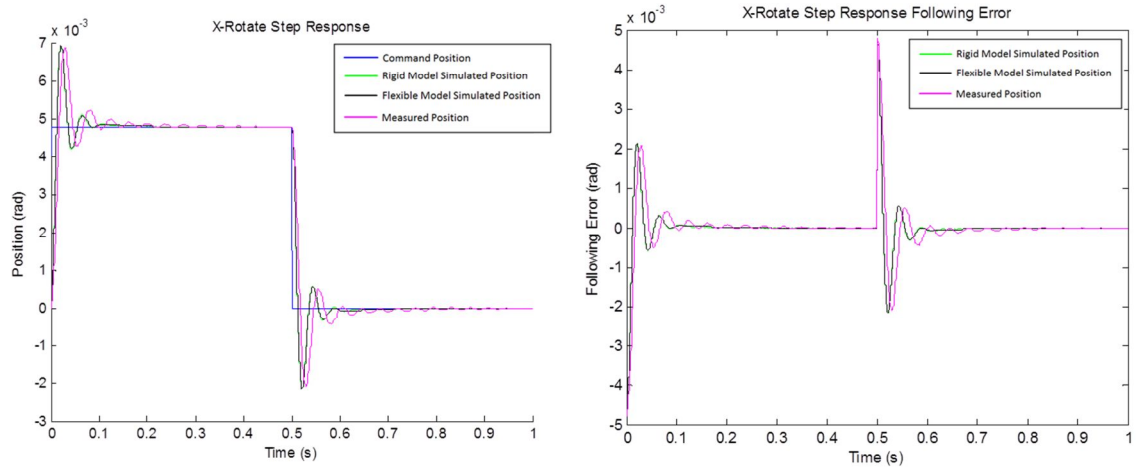
**Figure 6.7. Simulated and Measured X-Rotate Step Response with Calculated Damping Parameters.**

The simulated response is stable through the upward step, but after the downward step oscillations increase in amplitude. This is believed to be from instability in the Y-Rotate axis coupling through the bearings because when the flexible connection Y-Linear and X-Rotate axes are simulated with a rigid connection Y-Rotate axis these oscillations are not observed. Oscillations are also visible in the simulated current response (Figure 6.8).



**Figure 6.8. Simulated and Measured X-Rotate Step Response Current with Calculated Damping Parameters.**

The simulated positional step response and following error with increased bearing damping and Y-Rotate friction (as discussed previously) are shown in Figure 6.9. The rigid connection model is overlaid for comparison.



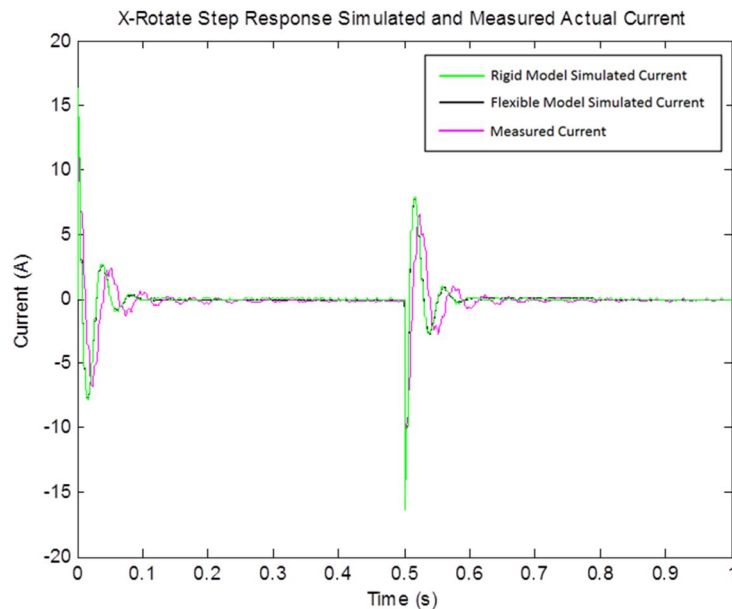
**Figure 6.9. Simulated and Measured X-Rotate Step Response with Improved Damping Parameters**

No instability is observed in the simulated response, and it tracks the measured response well, although as in the case of the rigid connection model the simulated response is faster than the measured. A quantitative comparison of the simulated and measured responses is shown in Table 6.3.

**Table 6.3. Comparison of Flexible Connection Simulation and Measured X-Rotate Step Response.**

Parameter	Simulated	Measured	Percent Difference
Delay Time (s)	0.0074	0.0111	32.95
Rise Time (s)	0.0120	0.0173	30.41
Peak Response (rad)	0.0069	0.0069	0.87
Peak Overshoot (%)	44.7500	43.5000	2.87
Peak Time (s)	0.0208	0.0297	30.02
Settling Time (s)	0.0714	0.0912	21.72

The simulated peak response amplitude and peak overshoot are within 3% of the measured values. The delay time, rise time and peak time lead the measured response by between 30% and 33%. The simulated settling time is nearly 22% faster than the measured. The RMS error is  $2.8e-4$ rad, which corresponds to 15.85% of the commanded step amplitude. The simulated and measured current responses are shown in Figure 6.10. The current response from the rigid connection model is overlaid and is nearly identical. As in the positional response, the simulated current leads the measured current, but the amplitude and shape of the simulated current response tracks well with measurements.



**Figure 6.10. Simulated and Measured X-Rotate Step Response Current with Improved Damping Parameters.**

The rigid and flexible connection models are compared for the X-Rotate axis in Table 6.4. The delay time, peak response magnitude and peak overshoot do not change from the rigid connection model, and only small increases in rise time and peak time prediction performance are observed. The settling time is slightly more accurate in the rigid connection model. Because the input torque and payload are direction coupled

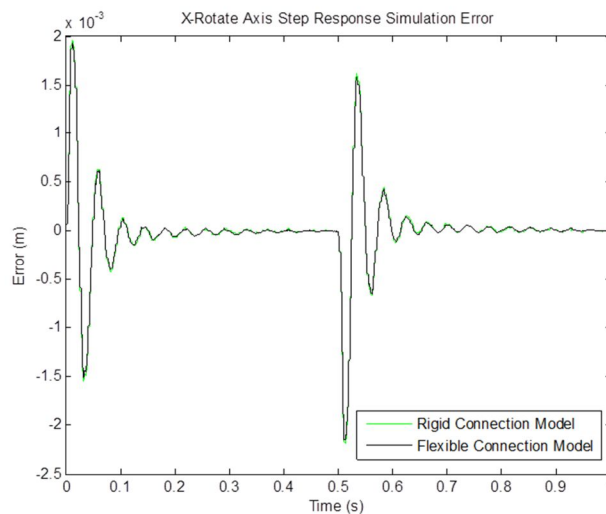
through the stiff ferrofluid seal drive shaft, it is expected that the addition of the shaft torsional stiffness has little effect on the positional response. The bearing stiffnesses

**Table 6.4. Performance Improvement from Flexible Connections in X-Rotate Step Response.**

Parameter	Rigid Connection (% Difference from Measured)	Flexible Connection (% Difference from Measured)	Performance Improvement (% Difference)
Delay Time	32.95	32.95	0
Rise Time	30.74	30.41	1.04
Peak Response	0.87	0.87	0
Peak Overshoot	2.87	2.87	0
Peak Time	30.20	30.02	0.62
Settling Time	21.11	21.72	-2.87

and shaft bending stiffness are not in the direction of motion, so they serve only to couple the X-Rotate axis to the Y-Linear and Y-Rotate axes. This coupling slightly increases the settling time. The RMS error is reduced 1.87% by the addition of flexible components.

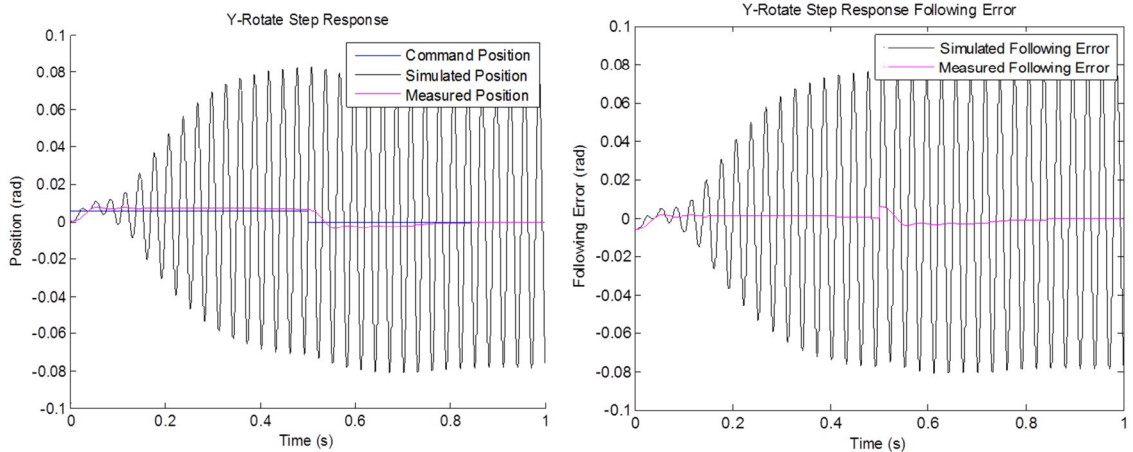
The simulation errors for the flexible and rigid connection models are overlaid in Figure 6.11. As expected, they are nearly identical.



**Figure 6.11. Error Between Rigid and Flexible Connection X-Rotate Models and the Measured Response.**

## Y-Rotate Axis

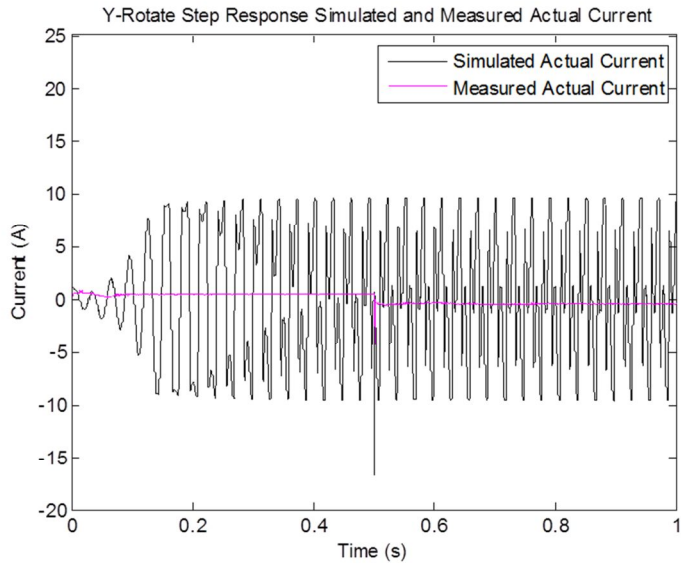
A positional step is applied to the Y-Rotate axis. The simulated response with calculated damping parameters is plotted with the measured response in Figure 6.12.



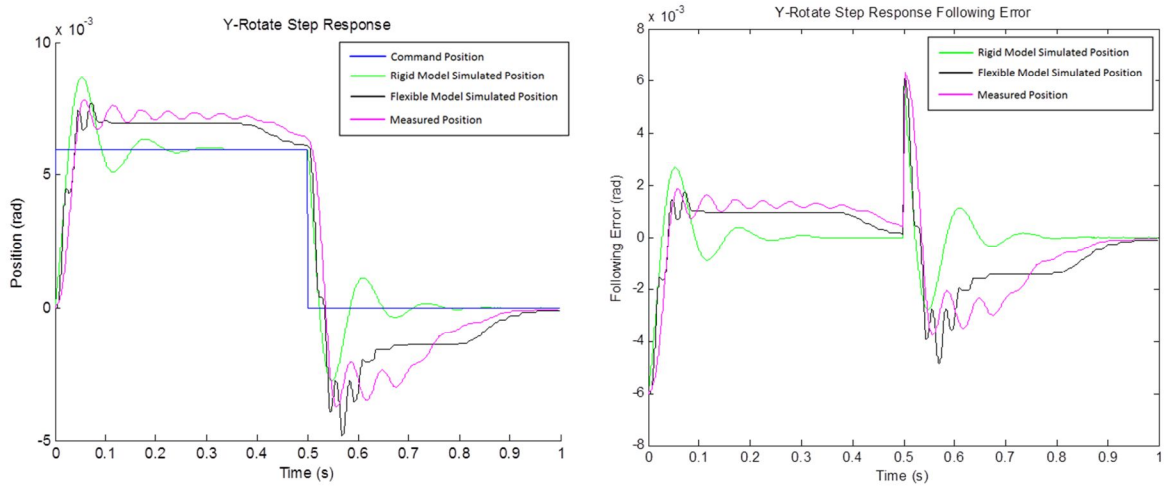
**Figure 6.12. Simulated and Measured Y-Rotate Step Response with Calculated Damping Parameters.**

The simulated response quickly becomes unstable. Oscillations increase in magnitude until reaching a constant amplitude after approximately 0.5s. The simulated and measured current responses are shown in Figure 6.13. As in the positional response, oscillations grow until reaching a large constant amplitude. However, in the current response the constant amplitude is reached at approximately 0.15s, much earlier than the positional response.

The simulated step response and following error with increased bearing damping and Y-Rotate friction are shown with the measured response in Figure 6.14. The rigid connection simulated response is also included for comparison.



**Figure 6.13. Simulated and Measured Y-Rotate Step Response Current with Calculated Damping Parameters.**



**Figure 6.14. Simulated and Measured Y-Rotate Step Response with Improved Damping Parameters.**

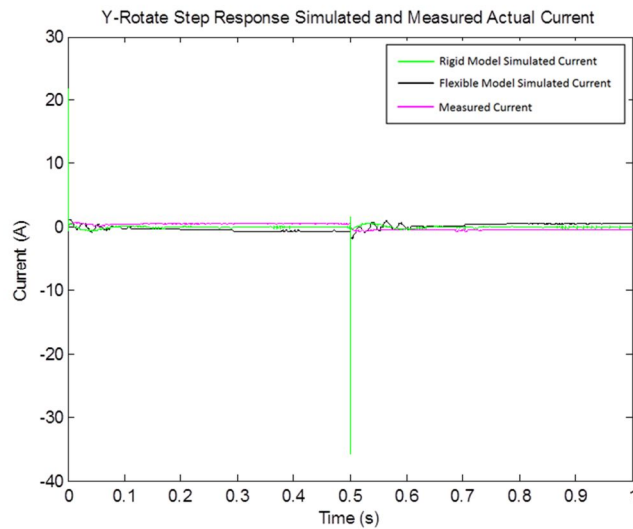
The simulated response has an initial peak that is not present in the measured data.

Additionally, the oscillatory behavior in the simulated response damps out much more quickly in the simulated response than in measurement. A quantitative comparison of the simulated and measured responses is shown in Table 6.5.

**Table 6.5. Comparison of Flexible Connection Simulation and Measured Y-Rotate Step Response.**

Parameter	Simulated	Measured	Percent Difference
Delay Time (s)	0.0178	0.0275	35.26
Rise Time (s)	0.0460	0.0403	14.16
Peak Response (rad)	0.0083	0.0078	6.42
Peak Overshoot (%)	39.2000	30.8000	27.27
Peak Time (s)	0.0566	0.0562	0.70
Settling Time (s)	0.5060	0.5038	0.42

The simulated peak and settling times are less than 1% larger than measured. The simulated peak response is less than 7% greater than the measured response, and the simulated rise time is just over 14% slower. The simulated peak overshoot is nearly 28% larger, and the delay time is slightly more than 35% faster than the measured response. The faster delay time and slower rise time make sense given the additional first peak in the simulated response. The RMS error between the simulated and measured responses is  $5.6e-4$ rad, which is 13.31% of the commanded step amplitude. The simulated and measured current responses are shown in Figure 6.15.



**Figure 6.15. Simulated and Measured Y-Rotate Step Response Current with Improved Damping Parameters.**



Overall the simulated response tracks the measured response well with sharp current increases at the step commands with quick returns to nearly zero current. The large magnitude of the simulated current response indicates that additional limits should be added to the model as such large current amplitudes are not possible in the physical system. The simulated current oscillates after the high-amplitude spikes. This oscillation is not seen in the measured response. The Y-Rotate axis rigid and flexible connection models are compared in Table 6.6.

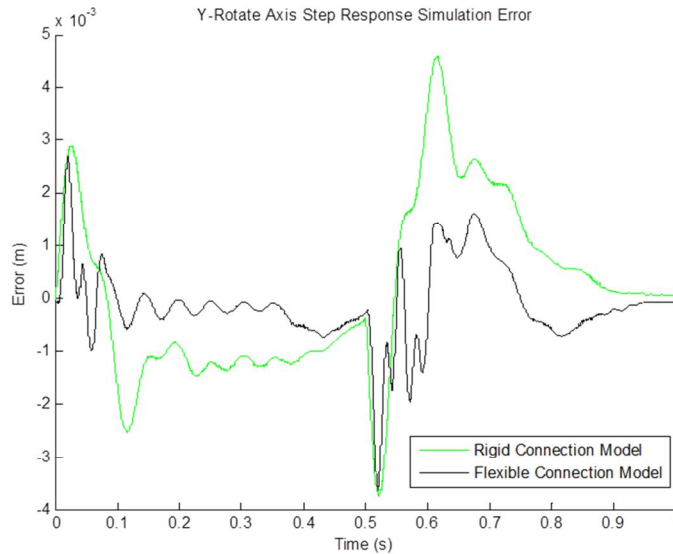
**Table 6.6. Performance Improvement from Flexible Connections in Y-Rotate Step Response.**

<b>Parameter</b>	<b>Rigid Connection (% Difference from Measured)</b>	<b>Flexible Connection (% Difference from Measured)</b>	<b>Performance Improvement (% Difference)</b>
Delay Time	46.55	35.26	24.25
Rise Time	31.44	14.02	55.40
Peak Response	10.70	6.73	37.14
Peak Overshoot	45.45	28.57	37.14
Peak Time	7.18	0.80	88.89
Settling Time	61.89	0.42	99.32

All response parameters are predicted more accurately by the model with flexible connections. The RMS error decreases 51.98% with the addition of the flexible connections. This makes sense as the Y-Rotate axis has the most flexible elements (as compared to the Y-Linear and X-Rotate axes), so the response is not adequately predicted by a rigid model.

The simulation error for the rigid and flexible connection models are plotted in Figure 6.16. The offset in the constant portion of the upward step observed in the rigid connection simulation error is not present in the flexible connection simulation error, but

in both cases the oscillatory behavior observed in the measured response is not present in the simulation. The rigid connection model has a large error peak after the first overshoot of the downward step that is not observed in the flexible connection model.



**Figure 6.16. Error Between Rigid and Flexible Connection Y-Rotate Models and the Measured Response.**

### **Simplified Flexible Connection Model**

A simplified version of the flexible connection model is developed to decrease simulation run time. The first frequency of each flexible element is determined (Table 6.7). Elements with first frequencies above 10x the controller bandwidth are assumed to have minimal impact on the closed-loop response, so they are treated as rigid. This simplifies the model by removing the Y-Linear payload bearing x-moment stiffness, the radial stiffness of the X-Rotate bearing, and the torsional and bending stiffnesses of the X-Rotate drive shaft.

The step responses are simulated for each of the three motion axes. The closed-loop model simulation time in Simulink is measured for the rigid connection, flexible

connection, and simplified flexible connection models. The simulation times are then normalized by dividing by the rigid connection simulation time. A comparison of the normalized simulation times is provided in Table 6.8.

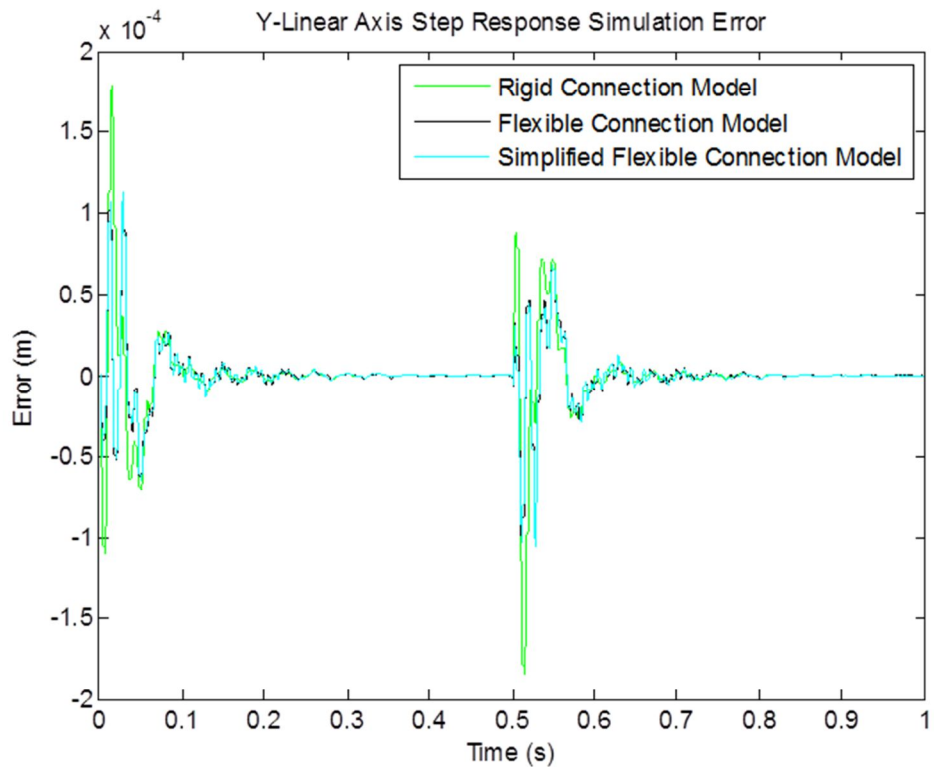
**Table 6.7. First Non-Rigid Frequency of Each Flexible Element. Elements with frequencies above 300Hz are assumed to be rigid in the simplified model.**

<b>Flexible Component</b>	<b>First Frequency (Hz)</b>
Y-Linear Cable	37.44
Y-Linear Counterweight Bearing X-Moment Stiffness	265.29
Y-Linear Counterweight Bearing Y-Moment Stiffness	197.25
Y-Linear Counterweight Bearing Z-Moment Stiffness	140.58
Y-Linear Payload Bearing X-Moment Stiffness	1179.05
Y-Linear Payload Bearing Y-Moment Stiffness	254.36
Y-Linear Payload Bearing Z-Moment Stiffness	235.70
X-Rotate Bearing Y-Radial Stiffness	433.18
X-Rotate Bearing Y-Moment Stiffness	77.35
X-Rotate Bearing Z-Moment Stiffness	76.28
X-Rotate Shaft Torsional Stiffness	1629.27
X-Rotate Shaft Bending Stiffness	349.86
Combined Y-Rotate Gearbox Torsional Rigidity and Axial Belt Stiffness	19.74

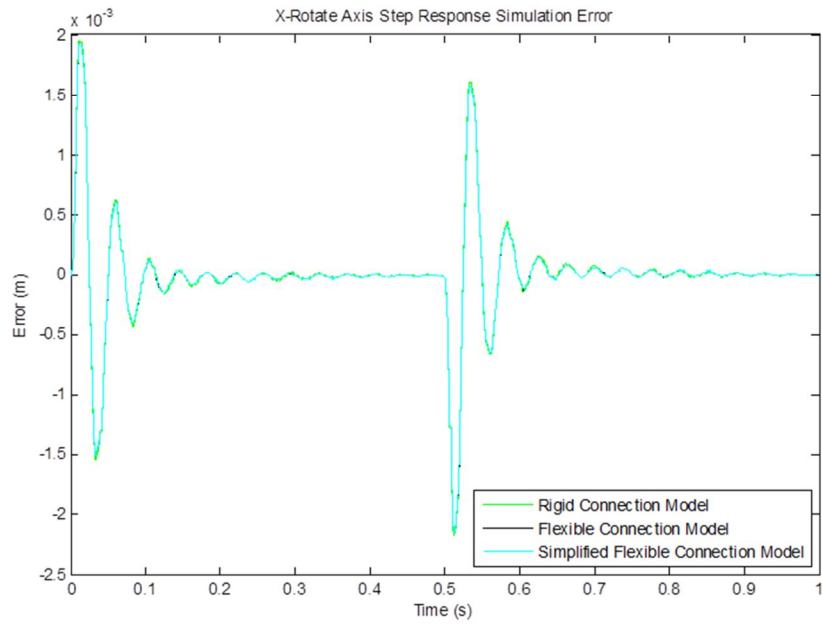
The simplified flexible connection model provides a run time performance improvement of 4.76x, 4.98x, and 6.89x for the Y-Linear, X-Rotate and Y-Rotate axes, respectively, as compared to the original flexible connection model. Figures 6.17 through 6.19 show the simulation error for each of the three simulation configurations for each of the three motion axes. The flexible connection and simplified model errors are nearly identical. This indicates that there is no significant loss in model predictive performance from the removal of the high-frequency stiffness elements. The differences in RMS error between the flexible and simplified models are 1.01%, 3.79e-3%, and 0.836% for the Y-Linear, X-Rotate and Y-Rotate axes, respectively.

**Table 6.8. Normalized Simulation Time for Rigid, Flexible and Simplified Flexible Connection Models.**

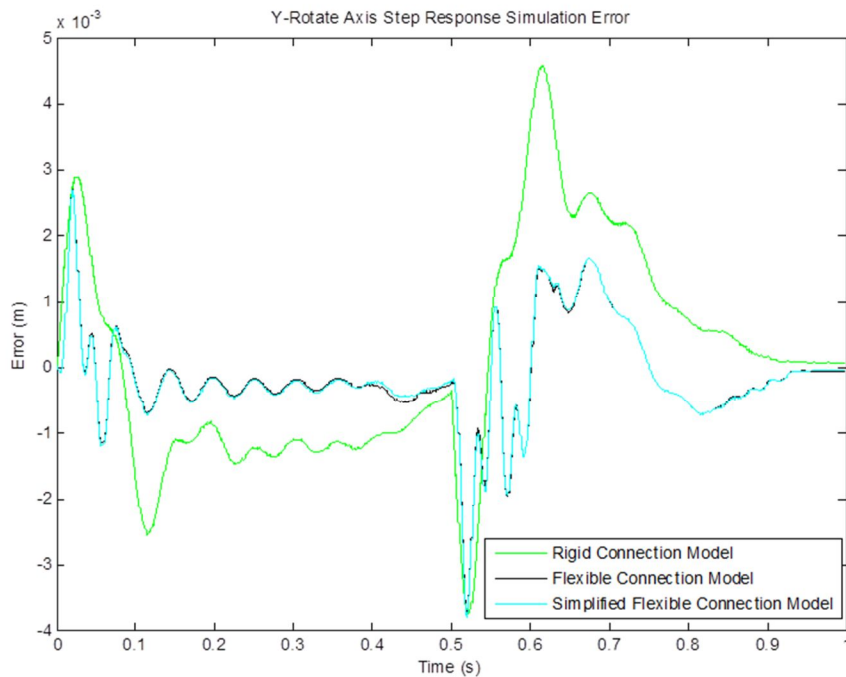
<b>Y-Linear Step</b>	<b>Normalized Simulation Time</b>
Rigid Connection Model	1
Flexible Connection Model	68.16
Simplified Flexible Connection Model	14.30
<b>X-Rotate Step</b>	
Rigid Connection Model	1
Flexible Connection Model	14.03
Simplified Flexible Connection Model	2.82
<b>Y-Rotate Step</b>	
Rigid Connection Model	1
Flexible Connection Model	23.37
Simplified Flexible Connection Model	3.39



**Figure 6.17. Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured Y-Linear Step Response.**



**Figure 6.18. Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured X-Rotate Step Response.**



**Figure 6.19. Simulation Error Between the Rigid, Flexible and Simplified Flexible Connection Models and the Measured Y-Rotate Step Response.**

## Conclusions

Correctly defining damping in predictive modeling applications is challenging since a prototype system is not available for measurement. In this model, the original simulation parameters significantly underestimated damping in some of the flexible elements which lead to closed-loop model instability. This model was stabilized by increasing bearing damping and the Y-Rotate output friction.

Overall the addition of flexible connections improved the model predictive performance as compared to the rigid connection simulation. The RMS error is decreased by 30.07%, 1.87%, and 51.98% for the Y-Linear, X-Rotate and Y-Rotate axes, respectively. The X-Rotate axis shows the least improvement, which is expected since the input and output are directly connected by a stiff shaft.

The addition of flexible components significantly increases closed-loop simulation time. This simulation time increase can be reduced by including only flexible components that are expected to have an impact on the closed-loop response. These elements are identified by comparing their first non-rigid natural frequency to 10x the controller bandwidth. Elements with higher frequencies are expected to have little effect on the closed-loop response and can be assumed to be rigid.

## **CHAPTER 7**

### **CONCLUSIONS**

A summary of the closed-loop mechanism modeling process is presented in a mechanism design framework. Modeling recommendations and rules of thumb are provided. Model limitations and areas for future work are discussed. Improving estimated model damping values, post-processing simulated and measured response data in the frequency domain, and the development of test procedures for model correlation are identified to be the most critical development areas for improvement of model performance.

#### **Mechanism Modeling in a Design Framework**

The goal of this study is to provide a mechanism modeling methodology that can be used during the design process to predict mechanism performance prior to the procurement of prototype parts. A methodology has been presented for generating reusable models of semiconductor mechanisms. Predictive performance was demonstrated for a three degrees-of-freedom wafer handling mechanism, with the simplified flexible connection simulation predicting the performance of all three axes with an RMS error less than 16%. The existing models can be used to guide the design process by identifying components and subsystems which dominate the mechanism response. Future applications of this modeling effort include design optimization and model-based control. The following sections describe the incremental construction of the closed-loop model.

## **Rigid Connection Mechanism Dynamics**

The first step in the mechanism modeling process is the generation of a rigidly connected mechanical plant model in Dymola. This model should include all masses and inertias in their proper orientations with a degree of freedom for each motion axis. All elements and connections are assumed to be rigid, and friction and damping are not initially considered. Motors are simplified to force or torque constants. Open loop simulation of a unit current applied to each axis should be performed, and the resulting acceleration should be compared to a mathematical model. If losses are expected to be a significant part of the mechanism response, they should be estimated and added to the model. The rigid connection model can then be used to size drive components such as motors and drive shafts. The rigid connection model is then brought into Simulink and integrated with a PMAC motion controller. The controller gains are tuned and should be used as a starting point for the controller tuning of more complex models.

## **Motor Electrical Dynamics**

Motor models should be generated in Dymola and verified by confirming the phase currents and force/torque output match a mathematical model for a common input. The motor model is then integrated with the rigid mechanism model in Dymola. This model may be used to predict motor temperature during operation using the heat ports in the phase resistances. The integrated rigid connection plant is then brought into Simulink and combined with a Copley motor controller for each motion axis. The motor controller gain parameters are tuned. Then the PMAC motion controller model (with initial parameters) is added, and the motion controller parameters are adjusted if necessary.



## **Flexible Connection Mechanism Dynamics**

Stiffness and damping are included for components with first frequencies lower than 10x the target controller bandwidth (a conservative estimate based on similar designs). The performance of the flexible connection model should approach the rigid connection model performance as the stiffnesses of the flexible elements become large. The flexible connection model is then combined with the motor models and integrated with the tuned motor and motion controllers in Simulink. Motion controller gains may be adjusted as necessary. This closed-loop model is then used to predict the performance of the mechanism. The design may be adjusted as necessary to achieve performance targets.

## **Model Limitations and Future Work**

Model limitations and areas for future work are identified. Development areas are categorized as system-level or component-specific depending on their scope. System-level development areas are the improved estimation of damping, frequency-domain response analysis, and the development of test procedures for parameter identification and black box modeling. These areas are expected to both improve model performance and help build both a better understanding of the system and better modeling intuition for systems with similar architectures. A list of component-specific development areas is also provided. These development areas are also expected to improve model performance, but they are more focused in scope and are expected to be smaller contributors to the closed-loop simulation error.

## **Estimation of Damping**

The flexible connection model generally predicts the performance of the test mechanism better than the rigid connection model. The RMS errors are decreased by

30.07%, 1.87%, and 51.98% for the Y-Linear, X-Rotate and Y-Rotate axes, respectively. However, reasonable estimates of friction and damping are required to achieve stable performance. This is challenging in predictive applications such as mechanism design where measured data is not yet available and supplier data may not be representative of the operating conditions of the mechanism under consideration. One approach to addressing this issue is to develop typical parameter ranges based on the measured response of similar components and drive mechanisms (e.g. determine a range of bearing loss factors based on bearing type). Additionally, identified parameters from validated models should be stored in a library for re-use. This is discussed further below.

### **Frequency-Domain Response Analysis**

A positional step move was selected because it is commonly used in the mechanism tuning process to characterize the system response. However, the step moves with small amplitudes used in this thesis may exhibit highly nonlinear behavior. Ideally the desired mechanism motion profile would be used to evaluate the model performance. A random or swept sine input can also be used, and these inputs would allow the generation of a mechanism frequency response function. Frequency peaks could be compared between the simulated and measured responses. Missing or additional peaks would help to provide insight into potential model improvement areas.

### **Parameter Identification and Black Box Modeling**

The third key development area is model correlation. In this thesis, the measured response was used to estimate bearing damping and Y-Rotate friction parameters since the predicted parameters were not sufficient to stabilize the model. However, this was only possible because hardware was available for characterization. This will not be the

case in a predictive application. In the future, models will be created from predicted parameters. Then, when prototype hardware is available all model parameters will be separately measured and identified based on measured response data to optimize the model performance. The model with identified parameters will be stored in a mechanism library for future reuse, and ranges of typical parameter values will be developed over time.

Similarly, black-box models fit to experimental data are an effective way to incorporate the dynamic behavior of proprietary components and subsystems. For the Y-Rotate axis, a black-box gearbox model determined experimentally may be a better alternative to the lumped-parameter gearbox model presented in this thesis since so little is known about the specific internal configuration of the gearbox due to its proprietary nature. The development of such a model is challenging because the losses are expected to be nonlinear and depend on both torque and speed. However, once a testing procedure is established it can be provided to suppliers allowing for all gearboxes to be characterized in a consistent way.

### **Component-Specific Development Areas**

Component-specific development areas are identified for each of the three motion axes. Component-specific development areas should be addressed in parallel with system development areas, but they are expected to have a smaller impact on the RMS simulation error. Of the component-specific development areas motor cogging and bearing stiffness are the highest priority because they are expected to have the largest relative effect on simulation accuracy.

## Y-Linear Axis

The model presented does not include cogging in any of the motors, including the linear motor in the Y-Linear axis. For the small amplitude step response investigated in this study cogging is not expected to have a significant impact. However, for larger amplitude responses cogging will create an additional position-dependent oscillatory response, which may impact servo performance and settling time. The addition of cogging to the motor model would ensure this oscillatory behavior is captured.

A second development area for the Y-Linear axis is the cable model, which currently considers only a single response mode of the cable. This mode corresponds to the fundamental frequency of the cable-mass system, so it is expected to be a dominant mode in the response. However, depending on the excitation applied to the system higher order modes may also contribute to the response. The cable model also assumes sufficient cable tension. With insufficient tension the cables may not bend completely around the pulleys adding a second compliance in series with the modeled axial compliance. A cable model that includes a cable tension check would ensure that minimum tension conditions are not violated, but determining an appropriate minimum tension value may be difficult. A more detailed cable model that includes higher order cable vibration modes and bending effects could be used in applications where low cable tension is suspected or where cable dynamics are of particular interest.

Finally, the effective moment stiffnesses and combined friction of the linear bearings are assumed to be constant. For small accelerations this is reasonable since the load only varies within a small range around the static loading conditions. However, to

model larger accelerations a lookup table may be used to determine the bearing stiffness based on the loading conditions since generally the bearing stiffness is not a linear function of load, and the normal force should be variable in the bearing friction calculation.

#### X-Rotate Axis

As in the Y-Linear axis the bearing stiffness is assumed to be independent of loading. An area for future development is the inclusion of the supplier provided force-displacement data in the model to enable the calculation of load-dependent bearing stiffness.

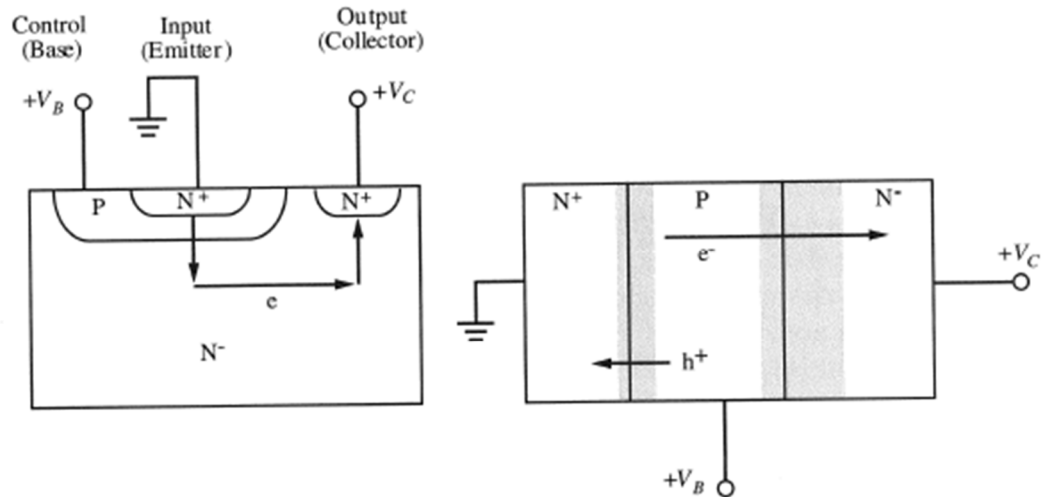
#### Y-Rotate Axis

The belt drive model assumes sufficient pretension. A pretension check would ensure this condition is met. Additional detail could also be added to the belt model to determine when slip would occur. This is especially important for mechanisms with flat or v-shaped belts.

## APPENDIX A

### TRANSISTORS

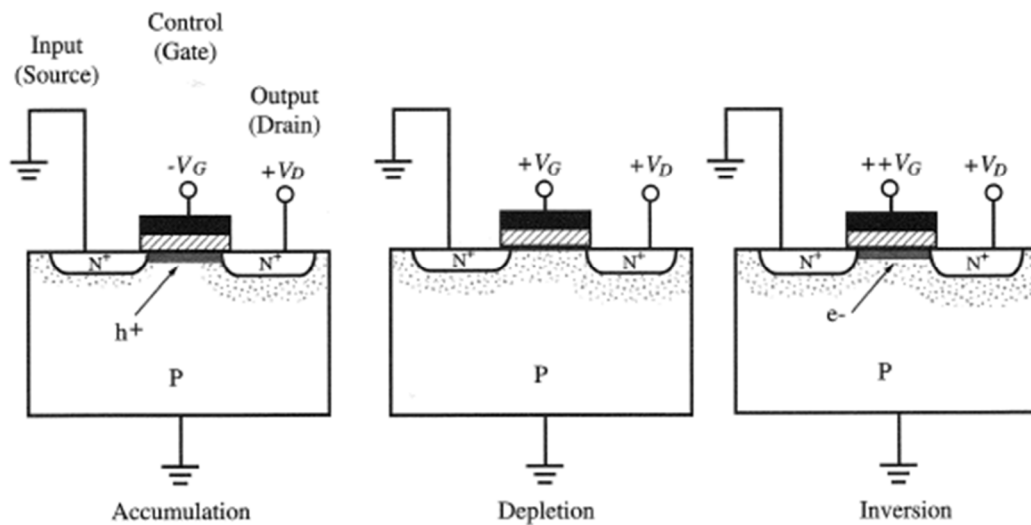
Transistors are used for switching or amplifying electrical signals. They typically have three terminals so the control terminal can be electrically isolated from the output [2]. Early ICs used bipolar junction transistors (BJTs). The BJT is a current-controlled device composed of three regions: the emitter (input), the collector (output), and the base (control). Structurally, the BJT is constructed from two *pn* junctions connected in series. Figure A.1 shows the physical structure of an *npn* device [27].



**Figure A.1: Simplified cross-section of BJT. (Left) Simplified cross section of BJT showing the emitter, collector and base terminals. Arrows show current flow from emitter to collector. (Right) 1D representation of BJT with arrows to indicate the flow of charge carriers. Image from Silicon VLSI Technology Fundamentals, Practice and Modeling [27].**

When a voltage is applied to the base it allows current to flow from the emitter to the collector. In analog applications, the proportionality between the applied base voltage and the collector current is utilized to create an amplifier. The BJT can also be used in digital applications. Typical switching times range from a few hundred nanoseconds to a few microseconds [2].

The field effect transistor (FET) also consists of three terminals: the source (input), the gate (control) and the drain (output). Some FETs have an additional fourth terminal known as the body. A voltage applied to the gate controls the conductivity between the source and the drain. The most common FET is the metal-oxide-semiconductor FET (MOSFET), named based on the material structure under the gate electrode—metal on top of an insulating oxide layer grown or deposited on the substrate (typically silicon). Figure A.2 shows the application of a voltage to the gate <sup>[27]</sup>.



**Figure A.2: Simplified cross-section of MOSFET. Left:** Simplified cross section of MOSFET showing the source, gate and drain regions. No voltage or a negative applied to the gate, so no electrons are able to move between the source and the drain. **Center:** A small positive voltage is applied to the gate which attracts electrons to the surface of the substrate under the gate. **Right:** A larger positive voltage is applied to the gate enabling electrons to flow between the source and the drain. Image from Silicon VLSI Technology Fundamentals, Practice and Modeling <sup>[27]</sup>.

MOSFETs have much faster switching times than BJTs, typically ranging from tens to hundreds of nanoseconds. Currently, more than 90% of ICs manufactured rely on MOSFETs as the primary switching element <sup>[27]</sup>.

# APPENDIX B

## SYMBOLIC GENERATION OF DYNAMIC EQUATIONS IN MATLAB

### Dynamic Equation Generation Script

```

%Rigid body Mathematical Model Equation Derivation for 3 DOF Vacuum
%Wafer Handling Mechanism

g= [0 -9.81 0];           %Gravity vector in robot base frame

syms a2 d1 d2 d4 t2 t3 t2dot t3dot dldot rg1_1 rg2_2 rg2_2x rg2_2y
rg2_2z rg3_3 rg3_3x rg3_3y rg3_3z rg1_b rg2_b rg3_b m1 m2 m3 I2_g2 I2
I2_11 I2_12 I2_13 I2_22 I2_23 I2_33 I3 I3_g3 I3_11 I3_12 I3_13 I3_22
I3_23 I3_33 rg1_1 rg1_1x rg1_1y rg1_1z T1 T2 T3 T_tot V1 V2 V3 V_tot w1
w2 w3 w4 L rg4_4 rg4_4x rg4_4y rg4_4z t4 t4dot I4 I4_g4 I4_11 I4_12
I4_13 I4_22 I4_23 I4_33 m4 r1_b r2_b r3_b r4_b dllddot t2ddot t3ddot
t4ddot f1 tau2 tau3 dcw mcw J3m w3m T3m

%Define Denavit-Hartenberg Structure-----
dh.a= [0 0 0 0];           %vector of a values
dh.f= [-pi/2 -pi/2 pi/2 0]; %vector of alpha values
dh.d= [d1 d2 0 d4];       %vector of d values
dh.t= [-pi/2 t2 t3 t4];   %vector of theta values

%Calculate coordinate transformations-----
T1_b= forwardKin(dh.a(1),dh.f(1),dh.d(1),dh.t(1));
%Coordinate transformation from frame 1 to base frame
T2_1= forwardKin(dh.a(2),dh.f(2),dh.d(2),dh.t(2));
%Coordinate transformation from frame 2 to frame 1
T3_2= forwardKin(dh.a(3),dh.f(3),dh.d(3),dh.t(3));
%Coordinate transformation from frame 3 to frame 2
T4_3= forwardKin(dh.a(4),dh.f(4),dh.d(4),dh.t(4));
%Coordinate transformation from frame 4 to frame 3

T2_b= T1_b*T2_1;           %Transformation from frame 2 to base frame
T3_b= T2_b*T3_2;           %Transformation from frame 3 to base frame
T4_b= T3_b*T4_3;           %Transformation from frame 4 to base frame

R1_b= T1_b(1:3,1:3);       %Rotation matrix from frame 1 to base frame
R2_b= T2_b(1:3,1:3);       %Rotation matrix from frame 2 to base frame
R3_b= T3_b(1:3,1:3);       %Rotation matrix from frame 3 to base frame
R4_b= T4_b(1:3,1:3);       %Rotation matrix from frame 4 to base frame

%Define position vectors-----
rg1_1= [rg1_1x; rg1_1y; rg1_1z];

```



```

%vector from coord. sys 1 to CG 1 in frame 1
rg2_2= [rg2_2x; rg2_2y; rg2_2z];
%vector from coord. sys 2 to CG 2 in frame 2
rg3_3= [rg3_3x; rg3_3y; rg3_3z];
%vector from coord. sys 3 to CG 3 in frame 3
rg4_4= [rg4_4x; rg4_4y; rg4_4z];
%vector from coord. sys 4 to CG 4 in frame 4

r1_b= T1_b(1:3,4);
%vector from coord sys 1 to base frame in base frame
r2_b= T2_b(1:3,4);
%vector from coord sys 2 to base frame in base frame
r3_b= T3_b(1:3,4);
%vector from coord sys 3 to base frame in base frame
r4_b= T4_b(1:3,4);
%vector from coord sys 4 to base frame in base frame

rg1_1b= R1_b*rg1_1;    %vector from coord. sys 1 to CG 1 in base frame
rg2_2b= R2_b*rg2_2;    %vector from coord. sys 2 to CG 2 in base frame
rg3_3b= R3_b*rg3_3;    %vector from coord. sys 3 to CG 3 in base frame
rg4_4b= R4_b*rg4_4;    %vector from coord. sys 4 to CG 4 in base frame

rg1_b= r1_b + rg1_1b;    %vector from base frame to CG 1 in base frame
rg2_b= r2_b + rg2_2b;    %vector from base frame to CG 2 in base frame
rg3_b= r3_b + rg3_3b;    %vector from base frame to CG 3 in base frame
rg4_b= r4_b + rg4_4b;    %vector from base frame to CG 4 in base frame
rcw_b= [0; dcw; 0];    %vector from base frame to CW CG in base frame

%Define inertia tensors-----
I2_g2= [I2_11 I2_12 I2_13; %link 2 inertia tensor in frame 2 coords.
        I2_12 I2_22 I2_23;
        I2_13 I2_23 I2_33];

I3_g3= [I3_11 I3_12 I3_13; %link 3 inertia tensor in frame 3 coords.
        I3_12 I3_22 I3_23;
        I3_13 I3_23 I3_33];

I4_g4= [I4_11 I4_12 I4_13; %link 4 inertia tensor in frame 4 coords.
        I4_12 I4_22 I4_23;
        I4_13 I4_23 I4_33];

I2= R2_b*I2_g2*R2_b';
%inertia tensor for link 2 rotation about CG 2 in base frame coords.

I3= R3_b*I3_g3*R3_b';
%inertia tensor for link 3 rotation about CG 3 in base frame coords.

I4= R4_b*I4_g4*R4_b';
%inertia tensor for link 4 rotation about CG 4 in base frame coords.

%Define angular velocity vectors-----

w1= [0 0 0].';          %Angular velocity of frame 1

```

```

w2= [t2dot 0 0].'; %Angular velocity of frame 2
w3= w2+R3_b*[0 0 t3dot].'; %Angular velocity of frame 3
w4= w3+R4_b*[0 0 t4dot].'; %Angular velocity of frame 4
w3m= r*r2/r1*t3dot; %Angular velocity of y-rotate motor

%Define linear velocity vectors-----

v1= [0 d1dot 0].'; %Velocity of frame 1
v2= v1; %Velocity of frame 2
v3= v2; %Velocity of frame 3
v4= v3 + cross(w3,R3_b*[0 d4 0].'); %Velocity of frame 4

vg2= v2 + cross(w2,rg2_2b); %Velocity of CG 2
vg3= v3 + cross(w3,rg3_3b); %Velocity of CG 3
vg4= v4 + cross(w4,rg4_4b); %Velocity of CG 4
vcw= -v1; %Velocity of CW CG

%Define kinetic energy of each link-----

Tcw= 1/2*mcw*(vcw.'*vcw); %Kinetic energy of CW
T1= 1/2*m1*(v1.'*v1); %Kinetic energy of link 1
T2= 1/2*m2*(vg2.'*vg2)+1/2*w2.'*I2*w2; %Kinetic energy of link 2
T3= 1/2*m3*(vg3.'*vg3)+1/2*w3.'*I3*w3; %Kinetic energy of link 3
T4= 1/2*m4*(vg4.'*vg4)+1/2*w4.'*I4*w4; %Kinetic energy of link 4
T3m= 1/2*J3m*w3m^2; %Y-rotate drive kinetic energy
T_tot= Tcw+T1+T2+T3+T4+T3m; %Total kinetic energy

%Define potential energy of each link-----

Vcw= mcw*g*(rcw_b-r1_b); %Gravitational potential energy of CW
V1= m1*g*rg1_b; %Gravitational potential energy of link 1
V2= m2*g*rg2_b; %Gravitational potential energy of link 2
V3= m3*g*rg3_b; %Gravitational potential energy of link 3
V4= m4*g*rg4_b; %Gravitational potential energy of link 4
V_tot= Vcw+V1+V2+V3+V4; %Total potential energy

%Calculate the Lagrangian-----

L= T_tot-V_tot; %Lagrangian

%Call Lagrange Function to determine dynamics equations-----

q= [d1 d1dot d1ddot t2 t2dot t2ddot t3 t3dot t3ddot t4 t4dot t4ddot];
%Vector of q variables for Lagrange Eqns
[M]= Lagrange(L,q); %Vector of dynamics equations

%Generate Motion Equations-----

eq1= M(1,1)==f1; %Generate dynamics equation for link 1
eq2= M(1,2)==tau2; %Generate dynamics equation for link 2
eq3= M(1,3)==tau3*r*r2/r1; %Generate dynamics equation for link 3
eq4= M(1,4)==0; %Generate dynamics equation for payload

v_d1ddot= solve(eq1,d1ddot); %Solve algebraically for d1ddot

```

```

v_t2ddot= solve(eq2,t2ddot); %Solve algebraically for t2ddot
v_t3ddot= solve(eq3,t3ddot); %Solve algebraically for t3ddot
v_t4ddot= solve(eq4,t4ddot); %Solve algebraically for t4ddot

```

### Transformation Matrix Calculation Function

```

function T = forwardKin(a,f,d,t)

st= s(t); %Sine t
sf= s(f); %Sine f
ct= c(t); %Cosine t
cf= c(f); %Cosine f

T= [ct    -st    0    a; %Coordinate transformation matrix
    st*cf  ct*cf  -sf  -sf*d;
    st*sf  ct*sf  cf   cf*d;
    0      0      0    1];

end

function out= c(in)
%Takes cosine of symbolic inputs and sets any values smaller than 1^-10
to zero
    this= cos(in);
        if (abs(this)<1^-10)==1
            this=0;
        end
    out=this;
end

function out= s(in)
%Takes sine of symbolic inputs and sets any values smaller than 1^-10
to zero
    this= sin(in);
        if (abs(this)<1^-10)==1
            this=0;
        end
    out=this;
end

```

### Lagrange Equation Calculation Function

```

function [M] = Lagrange(L,q)
%Determines equations of motion using the Lagrange Equation
%Inputs are the Lagrangian, L, and a vector of degrees-of-freedom
%Adapted from "Lagrange's Equations" function on Matlab Central
%http://www.mathworks.com/matlabcentral/fileexchange/23037-lagrange-s-
%equations posted Feb 19, 2009

syms t

Var= length(q)/3;

```

```

Vt= q;

%Create a vector with f1(t), diff(f1(t), t), diff(f1(t), t, t) for
each degree-of-freedom
for cont0=1:Var
    Vt(cont0*3-2)= strcat('f',num2str(cont0),'(t)');
    Vt(cont0*3-1)= diff(Vt((cont0*3)-2),t);
    Vt(cont0*3)= diff(Vt((cont0*3)-2),t,2);
end

for cont0=1:Var
    L1= simple(diff(L,q(cont0*3-1)));    %dL/dqdot
    L2= simple(diff(L,q(cont0*3-2)));    %dL/dq
    Dposx= L1;

    %Replace q(cont) with Vt(cont) in Dposx eqn
    for cont=1:Var*3
        Dposx= subs(Dposx,q(cont),Vt(cont));
    end

    L1= diff(Dposx,t);    %d/dt(dL/dqdot)

    %Replace Vt(cont) with q(cont) in L1 eqn
    for cont= Var*3:-1:1
        L1= subs(L1,Vt(cont),q(cont));
    end

    L1F= L1-L2;
    L1F= simple(expand(L1F));    %Expand terms then simplify
expression
    L1F= collect(L1F,Vt(cont0*3));    %Collect like terms
    M(cont0)= L1F;
end

end

```

## APPENDIX C

### MODELICA CODE FOR CABLE IN TENSION MODEL

The Modelica code for the CableInTension model is included in Figure C.1. This model is heavily based on the existing Elastogap component from the Modelica Standard Library. Modified sections are highlighted.

```
model CableInTension
  "1D translational spring damper combination that reacts only in tension"
  extends
    Modelica.Mechanics.Translational.Interfaces.PartialCompliantWithRelativeStates;
  parameter Real c(final unit="N/m", final min=0, start=1) "Spring constant";
  parameter Real d(final unit="N/ (m/s)", final min=0, start=1)
    "Damping constant";
  parameter Modelica.SIunits.Position s_rel0=0 "Unstretched spring length";
  parameter Real n(final min=1) = 1
    "Exponent of spring force ( f_c = -c*|s_rel-s_rel0|^n )";
  extends
    Modelica.Thermal.HeatTransfer.Interfaces.PartialElementaryConditionalHeatPortWithoutT;

  /*
  Please note that initialization might fail due to the nonlinear spring characteristic
  (spring force is zero for s_rel > s_rel0)
  if a positive force is acting on the element and no other force balances this force
  (e.g., when setting both initial velocity and acceleration to 0)
  */
  Boolean tension(start=true)
    "=true, if cable is in tension, otherwise cable is unloaded";
protected
  Modelica.SIunits.Force f_c "Spring force";
  Modelica.SIunits.Force f_d2 "Linear damping force";
  Modelica.SIunits.Force f_d
    "Linear damping force which is limited by spring force (|f_d| <= |f_c|)";
equation
  // Modify contact force, so that it is only "pushing" and not
  // "pulling/sticking" and that it is continuous
  tension = s_rel > s_rel0;
  f_c = smooth(1, noEvent( if tension then c*abs(s_rel - s_rel0)^n else 0));
  f_d2 = if tension then d*v_rel else 0;
  f_d = smooth(0, noEvent( if tension then (if f_d2 < -f_c then -f_c else
    if f_d2 > f_c then f_c else f_d2) else 0));
  f = f_c + f_d;
  lossPower = f_d*v_rel;
  a;
end CableInTension;
```

Figure C.1. Modelica Code for CableInTension Component. Modifications from Elastogap are highlighted.

## REFERENCES

- [1] AL-SHYAB, A. and KAHRAMAN, A. "A Non-Linear Dynamic Model for Planetary Gear Sets," Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics, vol. 221, pp.567-576, 2007.
  
- [2] ANDERSON, B.L. and ANDERSON, R.L. *Fundamentals of Semiconductor Devices*. McGraw-Hill, 2005.
  
- [3] CHI, R.M. and SHU, H.T. "Longitudinal Vibration of a Hoist Rope Coupled with the Vertical Vibration of an Elevator Car" Journal of Sound and Vibration, vol. 148, pp. 154-159, 1991.
  
- [4] CHO, Y.M. and RAJAMANI, R. "Identification and experimental validation of a scalable elevator vertical dynamic model," Journal of Control Engineering Practice, vol. 9, pp. 181-187, 2001.
  
- [5] Copley Xenus user guide (Copley Controls Corp. 2008).
  
- [6] CRAIG, J.J. *Introduction to Robotics Mechanics and Control*. Upper Saddle River: Pearson Education, Inc., 2005.
  
- [7] DANESHI-FAR, Z., CAPOLINO, G.A., and HENAO, H. "Modeling and Simulation of Planetary Gearbox Effects on a Wound Rotor Induction Machine," 2012 IEEE International Symposium on Industrial Electronics (ISIE), Hangzhou, China, May 2012.
  
- [8] DAS, T.M. and DÜLGER, L.C. "Mathematical modelling, simulation and experimental verification of a SCARA robot," Simulation Modelling Practice and Theory, vol. 6, pp. 257-271, 2005.
  
- [9] DREW, S.J. and STONE, B.J. "Torsional Damping Measurements for a Gearbox," Mechanical Systems and Signal Processing, vol. 19, pp. 1096-1106, 2005.
  
- [10] Dymola Dynamic Modeling Laboratory User Manual (Dassault Systemes AB, September 2012).

- [11] FERRETTI, G., MAGNANI, G., ROCCO, P, BONOMETTI, L., and MARAGLINO, M. “Simulating Permanent Magnet Brushless Motors in Dymola” 2<sup>nd</sup> International Modelica Conference Proceedings, Oberpfaffenhofn, Germany, March 2002.
- [12] FERRETTI, G., MAGNANI, G., and ROCCO, P. “Virtual prototyping of mechatronic systems,” *Annual Reviews in Control*, vol. 28, pp.193-206, 2004.
- [13] FLITNEY, R. K., *Seals and Sealing Handbook*. Burlington: Elsevier Ltd., 2007
- [14] Gates Mectrol Timing Belt Theory (Gates Mectrol, October 2006).
- [15] GENTA, G. and AMANTI, N. “On the equivalent viscous damping for systems with hysteresis,” *Atti Della Accademia Delle Scienze Di Torino, Classe Di Scienze Fisiche Matematiche e Naturali*, vol. 32, pp. 21-45, 2008.
- [16] JAZAR, R.N. *Theory of Applied Robotics: Kinematics, Dynamics, and Control*. Second Edition, Springer, 2010.
- [17] KIEKBUSCH, T., SAPPOK, D., SAUER, B., and HOWARD, I. “Calculation of the Combined Torsional Mesh Stiffness of Spur Gears with Two- and Three-Dimensional Parametrical PE Models,” *Journal of Mechanical Engineering*, vol. 57, pp. 810-818, 2011.
- [18] KIM, S. “Modeling and Fault Analysis of BLDC Motor Based Servo Actuators for Manipulators,” *IEEE International Conference on Robotics and Automation*, Pasadena, CA, May 2008.
- [19] LI, Y.H., MA, Y., LIU, S.T., LUO, Z.J., MEI, J.P., HUANG, T., and CHETWYND, D.G. “Integrated design of a 4-DOF high-speed pick-and-place parallel robot,” *CIRP Annals – Manufacturing Technology*, vol. 63, pp 185-188.
- [20] MARSH, J. *Motor Parameters Application Note* (Parker-Trilogy Linear Motors, November 14, 2003).
- [21] MATHIA, K. *Robotics for Electronics Manufacturing Principles and Applications in Cleanroom Automation*. Cambridge: Cambridge University Press, 2010.

- [22] MEROLI, S. “Two growth techniques for mono-crystalline silicon: Czochralski vs. Float Zone”, [http://meroli.web.cern.ch/meroli/Lecture\\_silicon\\_floatzone\\_czochralski.html](http://meroli.web.cern.ch/meroli/Lecture_silicon_floatzone_czochralski.html) (Accessed April 2<sup>nd</sup>, 2014).
- [23] MOHAN, N., UNDELAND, T.M., and ROBBINS, W.P. *Power Electronics Converters, Applications, and Design*. John Wiley & Sons, Inc., 2003.
- [24] PELCHEN, C., SCHWEIGER, C., and OTTER, M. “Modeling and Simulating the Efficiency of Gearboxes and Planetary Gearboxes,” 2<sup>nd</sup> International Modelica Conference Proceedings, Oberpfaffenhofn, Germany, March 2002.
- [25] PETERSEN, D., HOWARD, C., SAWALHI, N., and AHMADI, A.M. “Analysis of bearing stiffness variations, contact forces and vibrations in radially loaded double row rolling element bearings with raceway defects,” *Mechanical Systems and Signal Processing*, vol. 50-51, pp. 139-160, 2015.
- [26] PINHO, M., GENEVAUX, J.M., DAUCHEZ, N., BROUARD, B., COLLAS, P., and MEZIERE, H. “Damping Induced by Ferrofluid Seals in Ironless Loudspeaker,” *Journal of Magnetism and Magnetic Materials*, vol. 356, pp. 125-130, 2014.
- [27] PLUMMER, J.D., DEAL, M.D., and GRIFFIN, P.B. *Silicon VLSI Technology Fundamentals, Practice and Modeling*. Upper Saddle River: Prentice Hall, Inc., 2000.
- [28] PURNA CHANDRA RAO, A., OBULESH, Y.P., and SAI BABU, C. “Mathematical Modeling of BLDC Motor with Closed Loop Speed Control Using PID Controller Under Various Loading Conditions,” *ARPN Journal of Engineering and Applied Sciences*, vol. 7, pp.1321-1328, 2012.
- [29] RIVIN, E.I. *Handbook on Stiffness and Damping in Mechanical Design*. New York: ASME Press, 2010.
- [30] ROBERTS, R. “Control of high-rise/high-speed elevators,” Presented at the American Control Conference, Philadelphia, PA, June 1998.
- [31] SCHLEGEL, C., HOSL, A., and DIEHL, S. “Detailed Loss Modeling of Vehicle Gearboxes,” *Proceedings of the 7<sup>th</sup> Modelica Conference*, Como, Italy, September 2009.



- [32] SLOCUM, A.H. "Precision Machine Design," Dearborn, Michigan: Society of Manufacturing Engineers, 1992.
- [33] Turbo PMAC software manual (Delta Tau Systems Inc. 2008).
- [34] Ultra-tec Cable Railing Systems Derivation of Equations for Code Required Intermediate Railing Sphere Pass-Through Resistance (Ultra-Tec Corporation, 2003-2008)
- [35] VAN DER LINDEN, F.L.J., VAZQUES DE SOUZE SILVA, P.H. "Modelling and Simulating the Efficiency and Elasticity of Gearboxes," Proceedings of the 7<sup>th</sup> Modelica Conference, Como, Italy, September 2009.
- [36] YAMAMOTO, T., YASUDA, K., and KATO, M. "Vibrations of a String with Time-Variable Length," Bulletin of Japanese Society of Mechanical Engineers, vol. 21, pp. 1677-1683, 1978.
- [37] YOSHIKAWA, T. *Foundations of Robotics Analysis and Control*. Cambridge: The MIT Press, 1990.
- [38] ZEILLINGER, R. and KOTTRITSCH, H. "Damping in a Rolling Bearing Arrangement," Evolution Business and Technology Magazine from SKF, Feb. 15, 1996.