The Design and Control of Mechanical Switched Mode Drives

being a thesis submitted for the degree of

Doctor of Philosophy

in the University of Hull

by

R.S. Oldaker, BSc., MSc.

September 1997

Abstract

This thesis is concerned with the design, control and performance evaluation of a novel design for mechanical drives. This drive operates in a pulsed manner where energy is extracted from the input, stored and then released to the output. A spring acts as the energy store and brakes and clutches control the extraction and release of energy. By controlling the storage and release of this energy the device's output velocity can be controlled independently of the input velocity and since theoretically there is no energy loss the device operates in an analogous fashion to a variable ratio gearbox. Two design variations are presented. A step-up mechanism that is unidirectional and capable of output velocities greater than the input, and a step-up/step-down device that has bi-directional output velocity capabilities with no theoretical constraint on the value of output velocity. A prototype drive for each design is evaluated and detailed mathematical models are presented and compared to the prototypes. In addition a detailed design methodology is put forward for step-up/step-down devices.

Dedicated to the memory of my parents, Grace and Aubrey

Acknowledgements

The research described in this thesis was jointly funded by the EPSRC, Oxim Ltd and Fenner Plc, grant no. 20589.

I would like to thank my supervisor Dr Jim Gilbert for his invaluable guidance and encouragement throughout the period of this research. I would also like to thank Dave Joyce for his invaluable patience and technical help in the construction of the prototype.

Special thanks goes to Tony Mercer and Nathan Wood for their advice and encouragement during the difficult time of writing-up this thesis. I would also like to thank all my friends in the Department for their support and help over the last four years.

Table of contents

List of Figur	es	i
List of Table	S	V
Chapter 1	Introduction	1
1.1 Ine Sv	The Stee U. M. 1	2
1.1.1	The Step-Up Mechanism	3
1.1.2	The Step-Up/Step-Down Mechanism	4
1.1.3	Analogy to Electrical Switched Mode Power Supplies	5
1.2 Curren	t Variable Speed Drives	7
1.3 The Th	esis Aim	12
1.4 Layout	of Thesis	13
Chapter 2	Switching Algorithms and Feedback	
	Control Schemes	15
2.1 Step-U	p Mechanisms	16
2.1.1	PWM Switching	16
2.1.2	Resonant Switching	18
2.1.3	Clutch Matching	20
2.1.4	Feedback Control Loop	21
2.2 Step-up	b/Step-down Mechanisms	22
2.2.1	Resonant switching Algorithm	22
2.2.2	Feedback control loop	25
2.3 Conclu	sions	
Chapter 3	The Modelling and Simulation of Flexible	
	Drive Mechanisms	31
3.1 Mather	natical Models	31
3.1.1	Three-mass model for the step-up mechanism	34
3.1.2	Three-mass model for the step-up/step-down mechanism	35
3.2 The Nu	merical Solution	
3.3 Efficier		20
	icv Analysis	
3.4 Conclu	icy Analysis	
3.4 Conclu Chapter 4	sion The Design and Performance of a Prototype	
3.4 Conclu Chapter 4	The Design and Performance of a Prototype Step-up Drive System	
3.4 Conclu Chapter 4 4.1 Prototy	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation pe Results Open and closed loop tests	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation pe Results Open and closed loop tests Comparison of PWM and resonant switching	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation pe Results Open and closed loop tests Comparison of PWM and resonant switching Effect of clutch matching	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.3 4.2.4	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.4 4.3 Mathem	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation pe Results Open and closed loop tests Comparison of PWM and resonant switching Effect of clutch matching Increased Motor Inertia	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.4 4.3 Mathem 4.3.1	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.4 4.3 Mathem 4.3.1 4.3.2	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.4 4.3 Mathem 4.3.1 4.3.2 4.3.2 4.3.3	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	
3.4 Conclu Chapter 4 4.1 Prototy 4.2 Prototy 4.2.1 4.2.2 4.2.3 4.2.4 4.3 Mathem 4.3.1 4.3.2 4.3.3 4.3.4	The Design and Performance of a Prototype Step-up Drive System pe Construction and Instrumentation	

4.3.5 Comparison of Model with R	eal System61
4.4 Efficiency Analysis	64
4.5 Conclusions	
Chapter 5 The Computer Aideo	d Design of Step-
Up/Step-Down Mech	anisms 77
5.1 Component Selection	
5.1.1 Mathematical Analysis	
5.1.2 Sizing the spring	
5.1.3 Using the MATLAB Toolbox	k for Component Selection88
5.2 Simulation and Efficiency Analysis	
5.2.1 Mathematical Model	
5.2.2 Analysis of power losses	
5.2.3 Using the MATLAB toolbox	c for Simulation and Efficiency
Analysis	
5.3 Feedback Controller Design	
5.3.1 Discrete Controller Model	
5.3.2 Choosing the Controller Gair	18
5.4 Example Design	
Chapter 6 The Design & Perfor	mance of a Prototype
Step-Up/Step-Down	Mechanism 117
6.1 Introduction	
6.2 Computer Aided Design	
6.3 Initial Test Results	
6.4 Mathematical Model of Prototype	
6.4.1 Bearing Friction Coefficients	
6.4.2 Spring stiffness and spring sy	/stem inertia
6.4.3 Comparison of Model and Re	eal System139
6.5 Efficiency Analysis	
6.6 Closed Loop Results	
6.7 Conclusion	
Chapter 7 Conclusion	153
7.1 Further work	
Appendix $A = Calculating the time$	ng of the clutch for the
ston un mochonism	157
step-up mechanism	107
Appendix B - Calculating the timin	ng of the clutches for
the step-up/step-dow	n mechanism 159
Amondia C Model file used for t	he simulation of sten-
Appendix C - Wioder file used for t	ne simulation of step-
up mechamisms	105
Appendix D - Model file used for t	he simulation of step-
up/step-down mecha	inisms 169

Appendix E - Controller function used in step-up prototype and simulation	173
Appendix F - Controller function used in step-up/step- down prototype and simulation	179
Appendix G - Component data sheets for step-up prototype	186
Appendix H - Photograph of step-up prototype	191
Appendix I - Reference Guide to the MATLAB CAD toolbox of step-up/step-down mechanisms	193
Appendix J - Parameter model file used in simulation of 500 watt example design	202
Appendix K - Engineering drawing of step-up/step-down mechanism	204
Appendix L - Photograph of step-up/step-down prototype	206
References	208

List of Figures

Figure 1.1 - Schematic diagrams of a) the step-up and b) the step-up/step-	
down switched mode mechanical drive mechanisms	3
Figure 1.2 - Circuit diagrams of a) the "boost" and b) the "buck-boost"	
SMPS circuits	6
Figure 1.3 - Different designs of mechanical variable ratio drives: (a) belt	
drives, (b) mechanical couplings	9
Figure 2.1 - Energy transfer cycles for PWM switching	17
Figure 2.2 - Flow diagram for PWM switching	
Figure 2.3 - Energy transfer cycles for resonant switching	
Figure 2.4 - Flow diagram for resonant switching	20
Figure 2.5 - Feedback control loop for the step-up mechanism	22
Figure 2.6(a) - Energy transfer cycles producing positive load velocity	24
Figure 2.6(b) - Energy transfer cycles producing negative load velocity	25
Figure 2.7 - Schematic diagram of control loop for the step-up/step-down	
mechanism	25
Figure 2.8 - PI controller feedback scheme with velocity squared feedback	28
Figure 2.9 - Modified PI feedback scheme for bi-directional output	29
Figure 3.1 - Force vs. velocity characteristic used to model bearings	32
Figure 3.2 - Torque vs. velocity characteristic for clutch and brake model in	
the "on-state"	33
Figure 3.3 - Three mass model representation of step-up mechanism	34
Figure 3.4 -Three mass model representation of the step-up/step-down	
mechanism	35
Figure 3.5 - Overview of software to perform numerical simulation	
Figure 4.1 - Schematic diagram of prototype with instrumentation	43
Figure 4.2 - Accelerating a mass from rest using PWM switching	
Figure 4.3 - Accelerating a mass from rest using duty cycles of 0.05 to 0.5	45
Figure 4.4 - Step-up prototype under closed-loop control with a reference	1.6
velocity of 287 rad/s	46
Figure 4.5 - A comparison of resonant and PWM switching accelerating a	47
mass from rest	47
Figure 4.6(a) - Comparison of resonant switching with and without clutch	40
matching	
Figure 4.6(b) - Snapshot of comparison of resonant switching with and	40
without clutch matching	
Figure 4.7 - Power/velocity characteristic of motor driven with a constant 12	40
Figure 4.8 - Comparison of device with and without added motor inertia	
Figure 4.9 - Motor current vs. steady-state unloaded motor velocity	
Figure 4.10 - Comparison of motor model at four different input voltages	
Figure 4.11 - Schematic diagram of clutch and spring inertia experiment	
Figure 4.12 - Experimental and fitted data for spring "D" and load mass "A"	

Figure 4.13 - Grouping of velocity points: representative acceleration taken as	
gradient of line of best fit and representative velocity as the mean	
Figure 4.14 - Set of 25 frictional force/velocity points plotted with line of best	
fit	59
Figure 4.15 - Estimation of brake engagement delay and clutch	
disengagement delay	60
Figure 4.16 - Voltage and current supplied to clutch/brake unit through a	
single engagement/disengagement cycle	61
Figure 4.17(a) - Comparison of Model and Real System using PWM	
Switching	63
Switching	<i></i>
Figure 4.18 Step up mechanism with output norman means in	63
Figure 4.19 - Measured input power, output power measuring equipment	64
sten-up mechanism	(5
Figure 4.20 - Calculated losses attributed to the various components of step	05
up mechanism	66
Figure 4.21 - Discrepancy between model and real system for 320 rad/s	00
efficiency test	68
Figure 4.22 - Torque vs. displacement for spring used in efficiency tests	
Figure 4.23 - Maximum spring extension for the 25 efficiency runs	70
Figure 4.24(a)- Part of spring used for energy transfer in the 115 rad/s test	71
Figure 4.24(b) - Part of spring used for energy transfer in the 320 rad/s test	72
Figure 4.25 - Model and real system using a larger spring friction term	72
Figure 4.26 - Calculated losses attributed to various components using a	
larger spring friction term	73
Figure 4.27 - Simulated efficiency assuming a spring working completely	
within its linear region	74
Figure 4.28 - Simulated losses assuming spring working completely within its	
linear region	74
Figure 5.1 - Schematic diagram of the design process	
Figure 5.2 - Definition of steady-state velocity ripple	80
Figure 5.3 - Approximation for the time to complete one cycle	81
Figure 5.4 - Key dimensions of torsional herical spring	85 00
Figure 5.6 Example of graphical output from routine CADIS	00
Figure 5.7 - Using CADIS with multiple valued input parameters	90
Figure 5.8 - Example of graphical output from simulation routine SIMSUSD	91 95
Figure 5.9 - Example of graphical output from routine EFFSUSD	
Figure 5.10 - Graphical representation of point of marginal stability for the	
open-loop discrete model of the mechanism	100
Figure 5.11- Discrete feedback model	100
Figure 5.12 - Root locus trajectories for four different locations of the pole	
and the zero	101
Figure 5.13 - Closed loop system as a function of K_p and K_I	104
Figure 5.14 - Families of parameter values for four different switching rates	
(5,10,15 and 20 energy cycles/s)	106

 m²	109 109 112 113
 Figure 5.16 - Half second snap shot of open-loop test	109 109 112 113
 Figure 5.17 - Closed-loop test with a 400 rad/s step input Figure 5.18 - Closed loop performance with mixed step and ramp reference signals for positive and negative output directions Figure 5.19(a) - 400 rad/s step test with four different values of load torque (all figures quoted in Nm)	112
 Figure 5.18 - Closed loop performance with mixed step and ramp reference signals for positive and negative output directions	112
 signals for positive and negative output directions	113
 Figure 5.19(a) - 400 rad/s step test with four different values of load torque (all figures quoted in Nm) Figure 5.19(b) - 400 rad/s step test with four different values of load inertia (all values quoted in kg m²) Figure 6.1 - Power vs. velocity characteristic for motor driven at a constant 21 volts Figure 6.2 - Graphical output from routine CADJS Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions	115
 (all figures quoted in Nm)	
 Figure 5.19(b) - 400 rad/s step test with four different values of load inertia (all values quoted in kg m²) Figure 6.1 - Power vs. velocity characteristic for motor driven at a constant 21 volts Figure 6.2 - Graphical output from routine CADJS Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers. Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.10 - Velocity vs. friction data for the output bearing	114
 (all values quoted in kg m²)	1 1 4
 Figure 6.1 - Power vs. velocity characteristic for motor driven at a constant 21 volts Figure 6.2 - Graphical output from routine CADJS Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.10 - Velocity vs. friction data for the output bearing	114
 Figure 6.1 Polyter for volterly enducted site for motor driven at a constant 21 volts Figure 6.2 - Graphical output from routine CADJS Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.10 - Velocity vs. friction data for the output bearing	1 1 4
 Figure 6.2 - Graphical output from routine CADJS Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	110
 Figure 6.2 Oraphical output from fourne CADAS	. 110
 Figure 6.5 Open-loop sinulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings	119
 Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	
 Figure 6.4 - One second snap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	101
 Figure 6.4 - One second shap-shot at maximum load velocity Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	122
 Figure 6.5 - Simulation of closed-loop response for a mixed step and famp reference trajectory Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	1 2 2
 Figure 6.6 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	104
 Figure 6.0 - Output voltage waveform for specially designed clutch drivers Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	124
 Figure 6.7 - Experimental set-up to measure clutch delay Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	.120
 Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring 	.127
Figure 6.9 - Initial 100 watt tests for forward and reverse load directions Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	100
Figure 6.10 - Velocity vs. friction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	120
Figure 6.10 - Velocity vs. Inction data for the output bearing Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	129
Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	120
Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	122
Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring	.133
and in the star and star derive restations	124
Eigen (12(h) Displacement of tensor data (negative direction) for any in a	.134
Figure 6.12(b) - Displacement vs. torque data (negative direction) for spring	125
Used in the step-up/step-down prototype	.135
Figure 6.13 - Measured free vibration response of the spring with the fitted	100
model	.136
Figure 6.14 - Three hysteresis curves for the spring	137
Figure 6.15 - Log of peak amplitude vs. cycle no for free vibration test	138
Figure 6.16 - Viscous fit for the first cycle of the free vibration test	139
Figure 6.17(a) - Comparison of model and real system with an input power of	
50 watts and Bs = 67.6 x 10^{-3} kg m ²	140
Figure 6.17(b) - Snap-shot of above showing spring extension as well	141
Figure 6.18(a) - Comparison of model and real system with an input power of	
100 watts and Bs = $107 \times 10^{-5} \text{ kg m}^2$	142
Figure 6.18(b) - Snap-shot of above showing spring extension as well	142
Figure 6.19(a) - Controller design model and real system for an input power	
of 50 watts	144
Figure 6.19(b) - Controller design model and real system for input power of	
100 watts	144
Figure 6.20 - Percentage efficiency for step-up/step-down prototype operating	
with an input power = 100 watts	146

Figure 6.21 - Typical variator efficiency	147
Figure 6.22 - Losses attributed to the individual components of the step-	
up/step-down prototype	148
Figure 6.23(a) - Closed loop tests for the step-up/step-down prototype	150
Figure 6.24(b) - Error between reference and load velocities for closed loop	
test	150
Figure B.1 - Transfer and extraction delay times for positive load velocity	159
Figure B.2 - Quadrants containing correct solution for extraction delay	162
Figure B.3 - Quadrants containing correct solution for transfer delay	163

List of Tables

Table 1.1 - Comparison of a switched mode drive with other available	
variable ratio drives	12
Table 4.1 - List of Components for the Step-up Prototype	43
Table 4.2 - Summary of coefficients used in the model of the step-up	
prototype	62
Table 5.1 - Example specification	
Table 5.2 - Design parameters with clutch torque = 20 Nm	107
Table 6.1 - Example specification for step-up/step-down prototype	
Table 6.2 - Calculated parameter choices for clutch torque of 5 Nm	
Table 6.3 - Design parameters for spring used in step-up/step-down prototype	125
Table 6.4 - Energy efficiency for the different hysteresis curves	
Table B.1 - Choice of solution for energy extraction as a function of the sign	
of ω_{a} and ω_{t}	163
Table B.2 - Choice of solution for energy transfer as a function of the sign of	
ω_a and ω_t	164

Chapter 1 Introduction

Motors and other actuators will typically have a limited range of speeds where they produce significant levels of power. This is particularly true for cheap or light units which are optimised to produce high power only over a narrow velocity range. If it is required that high levels of power are supplied over a larger range of output speeds then it is necessary to use a gearbox with a set of different gears which allow a different ratio of motor and output speeds for each gear. For optimum performance a variable ratio drive should be used which will allow the motor (or *prime mover*) to rotate at a constant speed at or around maximum power and the output speed). This allows maximum power, minus losses in the drive, to be output over the range of speeds that can be achieved from the range of ratios available. A number of different variable ratio drives are available [1], which all have their own distinct advantages and disadvantages for use in any particular application. This thesis puts forward a novel design of variable ratio drive that possesses some distinct features and so adds to the design options available to engineers.

Part of the motivation for this device has come from the design of electrical switched mode power supplies [6,7] and works on the principle of pulsing packets of energy from the input to the output. In the electrical domain energy packets from the input supply are stored in an inductor and then released to the output. The extraction and release of these packets is controlled by switches and diodes. In the mechanical design the energy storage device is a spring and the energy packets are extracted and released by clutches and brakes.

The concept of a pulsing mechanical drive is not new. Constantinesco, Ljungstrom and Hobbs [8] suggested a pulsating mechanical drive and their original idea was 'reinvented' by Williams and Tipping [9] and successfully used as a drive system in a car. In this design a so called pulsator unit converted the constant rotation of the engine into an alternating level of torque which was rectified and transferred to the output shaft through two specially adapted sprag clutches. This system had draw backs in that the pulsing frequency was fixed to the rotational speed of the engine, which required the addition of springs to boost torque at low speeds, and the use of sprag clutches meant that engine braking could not be utilised. The switched mode mechanism outlined in this thesis differs from this mechanism in that the pulsing frequency is independent of the rotational speed of the prime mover and for the case of the step-up/step-down configuration, energy can be transferred from the prime mover to the load and vice versa.

1.1 The Switched-mode Concept and Operation

Fig. 1.1 shows schematic diagrams of the two design variations of switched mode mechanical drives discussed and analysed in this thesis. Fig. 1.1(a) shows the step-up drive mechanism and Fig. 1.1(b) the step-up/step-down mechanism. As can be seen both mechanisms consist only of a small number of basic components: a brake, a spring and a clutch or ratchet for the step-up device, and a spring and two clutches for the step-up/step-down device.

The basic idea behind both of these mechanisms is to keep the motor running at a constant speed, at or around the speed for maximum power, and to repeatedly transfer energy from the motor to the spring and then from the spring to the load. These *energy transfer cycles* should be repeated many times a second and are caused by the appropriate switching of the brake and clutch (step-up device) or both clutches (step-up/step-down device). The energy stored in the spring during a cycle is known as an *energy packet* and it can easily be seen that if the size of this packet is varied whilst keeping the rate at which the packets are transferred constant, then the power transferred will vary in a proportional manor. The operation of each design will now be considered in turn.



Figure 1.1 - Schematic diagrams of a) the step-up and b) the step-up/step-down switched mode mechanical drive mechanisms

1.1.1 The Step-Up Mechanism

In this design the motor is rigidly attached to one end of the spring and a brake is attached to the other and either a clutch or a ratchet used to connect the output from the brake to the load. With the brake engaged the motor will twist the spring and so store energy. Releasing the brake will allow this energy to be transferred to the load. The simplest arrangement is to let this energy be transferred to the load by the use of a ratchet, but alternatively, a clutch can be used to emulate the operation of a ratchet if the spring extension is monitored and the clutch released when all of the energy from the spring has been transferred. With this arrangement when all of the energy has been extracted from the spring the load must be going faster than the motor. This means that the mechanism will only produce load velocities that are greater than the motor velocity and is the reason for its name. It should also be noted that the velocity of the load will always be in the same direction to that of the motor.

1.1.2 The Step-Up/Step-Down Mechanism

In this design a spring has one end attached to ground and the other attached to a shaft. One end of the shaft is connected via a clutch to the motor and the other end connected via a clutch to the load. When clutch 1 is engaged and clutch 2 disengaged the motor will wind up the spring and so store energy. By disengaging clutch 1 and engaging clutch 2 this energy can be transferred to the load. As with the step-up device clutch 2 needs to be operated to perform like a ratchet by monitoring the extension of the spring and disengaging the clutch when all of the energy has been transferred. Operating in this way the load will rotate in an opposite direction to that of the motor.

This design has some interesting features that distinguish it from the step-up mechanism previously described. Firstly, since one end of the spring is held stationary, the output velocity can theoretically drop to zero. Secondly, if the spring is allowed to freely rotate through a half cycle before engaging clutch 2, the load will be able to rotate in the same direction as the motor, thus allowing the output velocity to be totally bi-directional. Thirdly, due to the input/output symmetry of the device, in the same way that energy can be transferred from the motor to the load, energy can also be transferred from the load to the motor. This third point allows the load to decelerate at the same rate as it can accelerate with no theoretical energy loss. If the second and third points were deemed unnecessary for a particular application then a ratchet could be used in place of clutch 2. This design has clearly more potential than the step-up device and so for this reason the thesis puts greater emphasis on the design and analysis of this mechanism.

1.1.3 Analogy to Electrical Switched Mode Power Supplies

The inspiration for the design of switched-mode mechanical drives has come, in part, from the design of electrical switched-mode power supplies. Switched-mode power supplies (SMPS) have been successfully used for many years as energy efficient DC-DC voltage converters. The step-up design can be thought of to be analogous to a SMPS with a so called "boost" circuit, and the step-up-step-down design (with a ratchet replacing clutch 2) analogous to a SMPS with "buck-boost" circuit. These two circuits are shown in **Fig. 1.2(a)** ("boost") and **Fig. 1.2(b)** ("buck-boost").

Consider first the operation of the "boost" circuit. When the switch is closed the current through the inductor will ramp upwards storing energy as it does so. When the switch is open this stored energy is delivered to the output circuit through the diode (which stops it from being reflected back). It is obvious that the output voltage must be greater than the input voltage otherwise the inductor would not discharge into the output circuit. This operation is analogous to that of the step-up mechanism previously described with the switch being an analogous component to the brake and the diode an analogous component to the ratchet.



Figure 1.2 - Circuit diagrams of a) the "boost" and b) the "buck-boost" SMPS circuits

A similar argument can be used for the operation of the "buck-boost" circuit with the switch being analogous to clutch 1 and the diode being analogous to clutch 2 (assuming it's operation mimics the behaviour of a ratchet). The bi-directional output and bi-directional energy capabilities of the step-up/step-down mechanism are not features of the "buck-boost" SMPS due to the presence of the diode.

This analogy can be taken further if we look at the differential equations governing the operation of the "boost" SMPS circuit shown in Fig. 1.2(a) and the "step-up" mechanical drive shown in Fig. 1.1(a). For this analysis we assume we have a constant motor velocity and that the load energy losses are viscous.

	Switch "closed"/ brake "on"	Switch "open"/ brake "off"
Electrical "boost" circuit	$V_{in} = L \frac{di_l}{dt}$	$V_{in} = LC\dot{V}_{out} + \frac{L}{R}\dot{V}_{out} + V_{out}$
Mechanical "step-up" mechanism	$\omega_m = \frac{1}{K} \frac{d\tau}{dt}$	$\omega_m = \frac{J}{K} \ddot{\omega}_l + \frac{B}{K} \dot{\omega}_l + \omega_l$

where,	ω"	=	Motor angular velocity	V_{in}	=	Input voltage
	ω_l	=	Load angular velocity	Vout	=	Output voltage
	τ	=	Spring torque	i_l	=	Current through inductor
	J	=	Load inertia	Ċ	=	Capacitance
	K	=	Spring stiffness	L	=	Inductance
	B	=	Load viscous loss coefficient	R	=	Resistance

We see that the forms of the differential equations are identical and by using the analogous variables of voltage for angular velocity and current for torque we can make the following equivalence relations,

$$L \equiv 1/K$$
$$J \equiv C$$
$$R \equiv 1/B$$

This suggests that an inductor is an analogous component to a spring, a capacitor is analogous to an inertial mass and viscous losses are analogous to electrical resistance.

Making analogies between mechanical and electrical systems is not new [10] and is perhaps best exemplified through the theory of "Bond Graphs" [11,12]. These allow a systematic and transparent way of modelling mixed electrical/mechanical/fluid systems through the systematic application of "flow" variables (velocity or voltage in our analogy) and "effort" variables (torque and current in our analogy).

1.2 Current Variable Speed Drives

This section attempts to summarise currently available variable speed drives and compares their characteristics to those of the switched-mode design. The variable speed drives of interest are the ones where the output velocity can vary whilst keeping the output power constant. This relationship is given below,

$$P_I = \tau_o \times \omega_o$$

where

 P_I = Input power τ_o = Output torque ω_o = Output angular velocity

Some devices which are called variable speed drives keep the output torque constant and hence are intrinsically inefficient since there will be a difference between input and output power. Examples of these are hydroviscous drives and magnetic couplings and for this reason these devices are not included in the review. Also not included are stepped ratio devices since these devices are only capable of producing a finite number of output speeds; a continuously variable output speed can only be achieved by changing the velocity of the prime mover. A manual multi-speed gearbox is an example of such a device.

Many different designs of variable speed drives have emerged and the remarkable growth of interest in this technology can be traced to the increasing cost of energy and the need to conserve resources. It is in this particular area that effective speed control can make a considerable contribution. Devices of interest can be broadly divided into three main categories: mechanical, electrical and hydraulic based systems and they are discussed separately below.

Mechanical Devices

These devices can be divided into two groups: belt drives with coned or split pulleys and variable speed couplings, which are also called variators. Examples of belt drives are shown in **Fig. 1.3** (a) and variable speed couplings in **Fig. 1.3** (b). The first design uses a belt which sits on either a cone or V-shaped split pulley. To achieve different drive ratios either the belt is moved (for cone designs) or the gap between either sides of the pulley is varied (for split pulley designs). Variable speed couplings use a freely rotating ball or cone with the input shaft in contact with one part and the output shaft in contact with another. Ratios are varied by either physically moving the ball or cone or changing its axis of rotation.



Figure 1.3 - Different designs of mechanical variable ratio drives: (a) belt drives, (b) mechanical couplings

Both of these devices rely on friction to transfer power and as a result inefficiencies are unavoidable. Friction also limits the ratios available which is typically 8:1 for belt drives and usually 9:1 (1:3 to 3:1) for mechanical couplings even though the so called Kopp design mechanical coupling can achieve 12:1 [1]. Ratio change is usually only achievable by manual means and is slow and the output speed is always unidirectional. The belt drive principle has been successfully used in the design of so called CVTs (continuously variable transmissions) for use as car transmissions systems [2, 3].

Hydraulic Drives

These can be divided into two main groups [4]: so called *hydrostatic* devices in which power transfer is accomplished by fluid pressure, with no change in fluid momentum, and *hydrokinetic* devices where power is transmitted by a transfer of the momentum of the fluid.

Hydrostatic devices consist of a motor and pump combination where either one must be of a variable displacement design. Hydrokinetic devices typically consist of a centrifugal type pump or impeller driving a turbine in close proximity. To be truly variable speed devices they need so called scoop control which varies the amount of active fluid between the impeller and turbine.

As with the mechanical devices previously described the change in ratio between input and output is a manual operation with some form of actuation device required for automatic control. Hydrostatic systems tend to be more expensive but are more efficient than hydrokinetic systems. One of the major advantages of using hydrostatic systems is that the pump and motor do not need to be adjacent, they only need to be connected via flexible hydraulic hosing. Since the motors have a very high power to weight ratio this design of variable speed drive can be very compact and light. Hydrokinetic drives require the input and output to be physically adjacent and the output rotation direction the same as the input. This last point also applies to hydrostatic drives but they can be made bi-directional by the introduction of a flow control valve into the system.

Electrical Drives

Electrical systems can be used to transfer mechanical power in an analogous fashion to the hydrostatic system previously described. In this system the prime mover drives a variable voltage DC generator which then directly drives a DC motor and is commonly called a Ward Leonard-Ilgner drive [5]. As with hydraulic systems the prime mover and motor do not need to be physically adjacent, they only need to be linked via wires carrying the current to the motor. The direction of rotation will always be that of the input but can be quite simply changed by the inclusion of a switch.

The switched-mode mechanism, which is the subject of this thesis, is really a *mechatronic* device since it needs control electronics as an integral component (to provide the correct switching signals). The features that distinguish it from the devices just described are it's inherent bi-directional output capabilities (without the need for a reverse gear (mechanical devices) or flow control valve (hydraulic systems)) and that closed loop velocity control is an integral feature for the mechanism. The effective ratio of the device is automatically varied as part of the velocity control loop and there is no need for manual adjustment. The other devices have manual ratio adjustment but require extra actuator equipment to achieve automatic ratio adjustment.

The choice of the most suitable drive will be influenced by the field of application. An example of the sorts of questions that need to be addressed when choosing the most appropriate drive system is given below,

- Type of machine to be driven.
- Maximum and minimum output speeds required and direction of rotation.
- Maximum power and torque to be transmitted.
- Nature of load, i.e., steady or fluctuating.
- Operational conditions i.e., fire hazard, ambient temperature, space available etc.
- Requirement for automatic or manual speed control.
- Requirement for flexible coupling between prime mover and drive motor.
- Minimum acceptable efficiency over a range of output speeds.
- Minimum acceptable capital and running costs of drive.

It is best from the designers perspective to have a wide range of possible drive systems so that a drive system can be found that closely matches the requirements of the application. As a summary to this section **Table 1.1** compares the important features of the drive systems mentioned and the switched mode mechanical drive which is the subject of this thesis. The switched mode drive in question is one using the step-up/step-down configuration and the efficiency and range of ratio figures are those taken from experimental results on the 100 watt prototype.

Type of drive	Range of ratios	Method of ratio change	Output velocity relative to input	Peak efficiency	Coupling between input and output
Belt	9:1	manual	uni- directional	85-90%	fixed
Mechanical coupling	up to 12:1	manual	uni- directional	85-90%	fixed
Electric	variable	manual	bi- directional	<80%	flexible
Hydrostatic	variable	manual	bi- directional	<80%	flexible
Hydro- kinetic	infinite up to speed of input	manual	uni- directional	<80%	fixed
Switched mode	7:1	auto	bi- directional	70-80%	fixed

 Table 1.1 - Comparison of a switched mode drive with other available variable ratio

 drives

1.3 The Thesis Aim

The analysis of Section 1.1 assumed the system components to be perfect. This however is not the case in reality. Clutches, for example, will always produce limited levels of torque, have an engagement and disengagement delay and posses inertia; springs will only have a finite energy capacity and will also have inertia. These imperfections in system components can easily cause serious degradation in system performance which is usually made apparent through serious energy losses. The relative significance and characteristics of these imperfections can be quite different from those found in the equivalent electrical components and present the designer

with new and challenging problems. The key to making a successful mechanical device is to minimise the effect of these imperfections on the overall system performance. This thesis is devoted to characterising these imperfections and coming up with methods to minimise their effect and so construct working prototypes that realise the theoretical ideas put forward in Section 1.1. This is achieved through novel switching algorithms (Chapter 2) and also the construction of a detailed design methodology that is capable of generating compatible sets of components that will maximise the performance of the overall system (Chapter 5).

1.4 Layout of Thesis

Chapter 2 deals with the switching algorithms that have been successfully used to control these devices and also explains the feedback structure that allows them to perform tracking of the output velocity. Crucial to the analysis and design of these devices is an accurate internal model. This is the subject of **Chapter 3** which presents models for both the step-up and step-up/step-down mechanisms. **Chapter 4** deals with the analysis and construction of a step-up prototype. Particular attention is paid to fitting the model, developed in **Chapter 3**, to this device so that the original model structure can be verified. Overall efficiency analysis is performed and the mathematical model is used to attribute losses to particular constituent components. This analysis then points the way for future system improvements.

Chapter 5 describes a design methodology and set of design tools for the construction of step-up/step-down mechanisms. Important in this is the development of a model which expresses the device's input/output behaviour in terms of some key design parameters. This model can be used to generate complete families of parameter values that will generate mechanisms with identical input/output characteristics. This gives the designer the widest possible design choice at an early point in the design process. This chapter then proceeds to develop an input/output model that can be used to choose the feedback gain values and finishes with an example design.

13

Chapter 6 describes the construction and analysis of the step-up/step-down prototype. This chapter starts by describing how the design tools mentioned in **Chapter 5** were used to design this prototype and then analyses its performance and compares it to that predicted by the design process. The rest of the chapter performs energy efficiency analysis and also assesses its closed-loop performance. The last chapter summarises and draws conclusions from the work described in this thesis and points the way for further research.

Chapter 2 Switching Algorithms and Feedback Control Schemes

This chapter describes the switching algorithms and feedback control schemes that have been used with each of the two mechanisms. The switching algorithms have been designed to take account of the imperfections in system components which are capable of seriously degrading system performance.

Consider these mechanisms with perfect components. This would be a motor with an infinitely large inertia, spring and clutches that have no inertia and clutches that switch infinitely fast and produce infinite torque. In this case the energy packet transfer rate (or *switching rate*) could be made arbitrarily large and so make the power throughput smooth. In addition all the energy in the spring will be transferred to the load since the spring will have no inertia and therefore no kinetic energy after the transfer has completed. However, a spring does have inertia (it also includes other components rigidly attached) and so has residual kinetic energy after the energy transfer stage has completed, and unless care is taken not to waste this residual energy significant energy losses can result. Since clutches have inertia care must be taken when engaging them since any velocity difference across either side will cause a loss in kinetic energy as the clutch forcible matches the velocity of both plates. Clutches also have engagement and disengagement delays which can limit the rate at which energy packets can be transferred.

The first section describes two switching algorithms that have been successfully applied to the step-up prototype. The first discussed, called PWM switching, is commonly used in electrical switched mode power supplies and is simple to implement but has drawbacks in terms of energy efficiency since it does not take into account the residual kinetic energy remaining in the spring. The second switching strategy, called resonant switching, was developed with the mechanical device in mind to overcome the problems associated with PWM switching. Also discussed is the method of matching the velocities of the clutch faces and the effect this has on the performance of the device.

The second section deals with the switching algorithm and feedback control scheme for the step-up/step-down device. This switching algorithm was developed from that used for the step-up mechanism incorporating both the clutch matching and resonant switching ideas. However, it also allows bi-directional energy flow and bi-directional output velocity, which are features associated only with the step-up/step-down device, and as a consequence this algorithm is more complicated in nature. Also discussed is the feedback control scheme for the device.

2.1 Step-Up Mechanisms

The two switching algorithms that have been tested on the step-up prototype mechanism will be the topic of the next two sub-sections. These switching algorithms are PWM switching, which is simple to implement but has deficiencies in terms of energy efficiency, and resonant switching which is an attempt to overcome some of the deficiencies of the PWM algorithm. Also discussed is clutch matching which attempts to prevent losses in the output clutch caused if it is engaged with different clutch plate velocities. Lastly the feedback loop for these devices is presented.

2.1.1 PWM Switching

This type of switching uses a fixed length of time between energy transfers and will perform energy storage for some proportion of this time. Fig. 2.1 shows a diagram of a few energy transfer cycles along with the signals sent to the clutch and brake. The length of time for the energy storage stage, t_k , will determine the energy packet size and since the transfer rate is fixed (due to t_l , the length of time for the cycle, being fixed) the proportion t_k is of t_l will ultimately determine the power output of the mechanism (within the limitations of the motor). The time taken to transfer this

energy will vary and depends on the velocity of the load and the spring extension. To guarantee that enough time is allowed for this transfer of energy the maximum proportion of the cycle allowed for energy storage will be set at 50% (which is also used in practice in the equivalent electrical circuit [6]). Justification for this approximation can be seen when it is considered that, apart from an initial acceleration phase, the load will normally be going faster than the motor so the time taken to unwind the spring will always be shorter than the time to wind up the spring. The remaining time of the energy cycle, known as *dead time*, is unused.



Figure 2.1 - Energy transfer cycles for PWM switching

To determine the point at which all the energy has been extracted from the spring (i.e. the point at which the energy transfer stage has been completed) the spring extension is measured and when this is zero the clutch is turned off. This process of switching the clutch mimics the operation of a ratchet. It can be seen from **Fig. 2.1** that the velocity of the spring when the brake is applied depends on the load velocity and the duration of t_k and in general will be non-zero. Engaging the brake with significant spring velocity can mean quite high power losses since physical constraints will

mean that the spring inertia is not negligible (it will consist of the moving parts of the spring as well as flanges connecting it to the brake and the moving brake components themselves). **Fig. 2.2** shows a flow diagram of the algorithm used by the controller routine when performing PWM switching.



Figure 2.2 - Flow diagram for PWM switching

This type of switching is easy to implement and, if a ratchet is used instead of a clutch, the switching algorithm needs only minimal instrumentation since no velocities or displacements need to be measured.

2.1.2 Resonant Switching

This type of switching is similar to PWM switching, described in the previous section, however, it attempts to minimise the energy losses associated with engaging the brake when the spring velocity is significant. In this algorithm the length of dead-time is fixed instead of the complete cycle. By tuning the dead-time length, t_n , appropriately, the brake will be applied when the spring velocity is close to zero.

Even though the time taken for zero velocity will not be in general fixed (it will be a function of the load and motor velocities) the approximation of using a fixed deadtime produces a workable compromise which is superior to PWM switching. A consequence of using this method of switching is that the overall energy transfer time will not be constant. A few energy transfer cycles for this type of switching is given in **Fig. 2.3** and a flow diagram for the resonant switching algorithm is given in **Fig. 2.4**.



Figure 2.3 - Energy transfer cycles for resonant switching

PAGE NUMBERING AS IN THE ORIGINAL THESIS

calculated if the resonant frequency of the spring (and attached fixings to the brake) and the motor velocity (assumed fixed during this period) are known. The equation defining this delay is given below and its derivation is contained in **Appendix A**,

$$t_{d} = 2\sqrt{\frac{J_{s}}{K}} \tan^{-1} \left[\frac{\sqrt{\frac{K}{J_{s}}}\theta_{e} - \sqrt{\frac{K}{J_{s}}}\theta_{e}^{2} - \omega_{l}(V_{l} - 2\omega_{m})}{\omega_{l} - 2\omega_{m}} \right]$$
(2.1)

where,

 t_d = Time delay J_s = Spring inertia K = Spring rate θ_e = Initial extension of spring ω_l = Load angular velocity ω_m = Motor angular velocity

It will be noticed that the spring rate and inertia do not need to be known separately but can be found by measuring the resonant frequency of the spring (assuming negligible friction) since this will be approximately $\frac{1}{2\pi}\sqrt{\frac{K}{J_s}}$.

Using this predictive scheme the clutch actuation delay can be taken into account by simply taking this delay away from the figure for t_d previously calculated. This is preferable to the alternative of monitoring the velocities on either side of the clutch and engaging it when they were found to match.

A comparison of resonant switching with and without clutch matching for the prototype step-up mechanism is given in Section 4.2.3.

2.1.4 Feedback Control Loop

This section describes the operation of the feedback control loop used for the step-up mechanism. The controller will modify the length of the energy storage stage, t_k (which is proportional to the duty-cycle for PWM switching) which will then vary the output velocity of the device. The output velocity is fed back to the controller and simple PI compensation is used to generate the control signal (the time t_k) given to

the switching algorithm (which can be either PWM or resonant switching). This arrangement is shown in Fig. 2.5.



Figure 2.5 - Feedback control loop for the step-up mechanism

An example of the prototype step-up mechanism operating under closed-loop control is given in **Section 4.2.1**.

2.2 Step-up/Step-down Mechanisms

These mechanisms require a more sophisticated switching algorithm than step-up devices due to their design allowing bi-directional output velocity and bi-directional energy transfer capabilities. This switching algorithm incorporates both the clutch matching and resonant switching ideas developed for the step-up mechanism.

2.2.1 Resonant switching Algorithm

Since the spring and its rigidly attached components (including connecting rod and clutch rotors etc.) will have finite inertia, energy will always be left over after the spring has transferred energy to the load. Allowances for this residual energy must be made otherwise significant energy losses can result. For the step-up mechanism the resonant switching algorithm introduced a fixed delay between the disengagement of the clutch (the completion of energy transfer) and the engagement of the brake (the start of energy storage). This allowed the spring velocity to return to zero and so not waste its kinetic energy when the brake was applied. For the step-up/step-down mechanism this delay must be such that the spring velocity matches the motor velocity and, in fact, can be correctly calculated if the resonant frequency of the

spring system is known. This is the same as the clutch matching algorithm described for the step-up mechanism but operated on the input clutch instead of the output clutch. The equation used to calculate the delay is slightly simpler since one end of the spring is held stationary, and this equation can also be used for the velocity matching of the output clutch due to the symmetry of the device. This equation is given below,

$$t_{d} = 2\sqrt{\frac{J_{s}}{K}} \tan^{-1} \left[\frac{-\sqrt{\frac{K}{J_{s}}}\theta_{e} \pm \sqrt{\frac{K}{J_{s}}}\theta_{e}^{2} - (\dot{\theta}_{s} + \omega_{e})(\dot{\theta}_{s} - \omega_{e})}{(\dot{\theta}_{s} + \omega_{e})} \pm n\pi \right]$$
(2.2)

where,

 t_d = Time taken for spring to reach target velocity K = Spring rate J_s = Spring inertia θ_e = Initial spring extension ω_e = Initial spring angular velocity $\dot{\theta}_s$ = Target spring angular velocity (load or motor)

This equation will find the time taken for an oscillating spring, with finite inertia and a generally non-zero initial velocity and spring extension, to achieve a target velocity. The target velocity will be either the motor velocity, to calculate the correct delay prior to energy storage, or the load velocity, to calculate the correct delay prior to energy transfer. What is more this equation also allows the bi-directional capabilities of the device to be realised since this equation will generate the correct clutch delays when the load and motor are going in the same and opposite directions. It can also handle bi-directional energy transfer by reversing the interpretation of the load and motor. The derivation of this equation is given in **Appendix B**. Also explained in this appendix is how to efficiently choose the correct solution (**Eq. (2.2)** has multiple solutions) for a particular operational mode of the device (direction of energy transfer and direction of rotation of motor and load). As with the step-up device it will be noticed that neither the spring rate, K, or the spring inertia, J_s , need to be known separately, only the square root of their ratio and is simply found as being 2π times the natural frequency of the spring. It is important that the time taken for the

velocities to match can be calculated in advance since it means that the clutch actuation delay can be taken into account by simply taking it away from the result found in Eq. (2.2). Typical energy transfer cycles are shown in Fig. 2.6(a) (positive load velocity) and Fig. 2.6(b) (negative load velocity).



Figure 2.6(a) - Energy transfer cycles producing positive load velocity


Figure 2.6(b) - Energy transfer cycles producing negative load velocity

2.2.2 Feedback control loop

So that the mechanism can follow a given velocity trajectory a feedback control loop is required. This loop controls the size of the energy packet used by the switching algorithm and is different to the feedback loop for the step-up mechanism which simply controlled the energy storage time. The feedback loop for step-up/step-down mechanisms is shown in **Fig. 2.7**.



Figure 2.7 - Schematic diagram of control loop for the step-up/step-down mechanism

To decide on an appropriate form of controller let us first consider the open loop model for this system. If we assume the switching rate to be fast compared to the output dynamics (defined by the output load inertia and load torque) then the



following continuous differential equation can be used to approximate the open-loop system,

$$\tau_m = J_l \hat{\Theta}_l + B_{total} \Theta_l + \tau_l$$
(2.3)

where,

 τ_m = Average torque produced by mechanism J_l = Load inertia B_{total} = Term representing viscous losses τ_l = Load torque

If we multiply through by $\dot{\theta}_i$ we have,

$$\tau_{m}\dot{\theta}_{l} = J_{l}\ddot{\theta}_{l}\dot{\theta}_{l} + B_{total}\dot{\theta}_{l}^{2} + \tau_{l}\dot{\theta}_{l}$$
(2.4)

Noting that the left-hand side is the output power of the device and substituting x for $\dot{\theta}_l^2$ into this equation then we have,

$$P = \frac{J_{l}}{2} \dot{x} + B_{total} x + \tau_{l} x^{\frac{1}{2}}$$
(2.5)

The output power (P) can be thought of as the energy packet size (ε_p) divided by the energy switching time for the device and so we have,

$$\varepsilon_p = \frac{J_l t_s}{2} \dot{x} + B_{total} t_s x + \tau_l t_s x^{\gamma_2}$$
(2.6)

which describes the dynamics of the device in terms of the control variable ε_p . This equation is non-linear due to the square-root term and to be able to analyse it, it is best to linearise it about an operating point. Hence applying this equation around an operating point $\varepsilon_p = \varepsilon_{p0}$ and $x = x_0$ and considering incremental changes $\Delta \varepsilon_p$ and Δx we have,

$$\varepsilon_{p0} + \Delta \varepsilon_{p} = \frac{J_{l}t_{s}}{2} (\dot{x}_{0} + \Delta \dot{x}) + B_{total}t_{s} (x_{0} + \Delta x) + \tau_{l}t_{s} (x_{0} + \Delta x)^{\frac{1}{2}}$$
(2.7)

and using the Taylor expansion,

$$(1+\delta)^{\frac{1}{2}} \approx 1+\frac{1}{2}\delta$$
 for $0 < \delta << 1$

we have finally,

$$\Delta \varepsilon_{p} = \frac{J_{l}t_{s}}{2} \Delta \dot{x} + \left(B_{total}t_{s} + \frac{\tau_{l}t_{s}}{2x_{0}^{\gamma_{2}}} \right) \Delta x$$
(2.8)

This is a linear model of the system which shows that the open-loop dynamics of the mechanism approximates to that of a first-order system.

Taking Laplace transforms of both sides we have the open loop transfer function,

$$G(s) = \frac{A}{s+B}$$
(2.9)

where,

$$A = \frac{2}{J_l t_s} \qquad \text{and} \qquad B = \frac{1}{J_l} \left(2B_{total} + \frac{\tau_l}{x_0^{\frac{1}{2}}} \right)$$

Let us consider this system being controlled by a proportional feedback control scheme. In this case the steady-state errors to an input of the form $\frac{t^k}{k!}$ are, using the

final value theorem,

$$e_{ss} = \lim_{s \to 0} \frac{1}{s^k} (1 - T(s))$$

where T(s) is the closed loop transfer function. This shows that,

$$e_{ss} = \frac{B}{B + KA}$$
 for $k=0$

= infinity for
$$k > 0$$

This shows the system to be of type 0 and would require a feedback gain (K) such that B+K>>B to achieve low steady-state errors to step inputs, with errors to ramp inputs being unbounded. However changing the system type to 1 by adding integral control is the preferred control strategy since it will have zero errors to step inputs and constant errors to ramp inputs. Following the same analysis as that just outlined for proportional control it is straightforward to see that errors to ramp inputs for this system are given by,

$$e_{ss} = \frac{B}{K_I A} \tag{2.10}$$

where K_1 is the proportional gain.

The analysis so far suggests using a PI (proportional plus integral) feedback controller with the velocity squared as the feedback variable should produce a feedback system which is of type 1. This feedback scheme is shown below,





It is interesting to note that using the velocity squared as the control variable really means that the device is controlling a quantity that is proportional to the kinetic energy of the load. Since the device is capable of driving the load in both forward and reverse directions the controller must be such that the sign information of the reference signal is not lost. The switching controller must be aware of the sign of the reference signal so that the output direction of the load can be chosen correctly. In addition a sign needs to be associated with the level of energy so that the controller can distinguish the case where the signs of the reference and output velocities are different. An example of what would happen if the sign was ignored would be where the reference velocity suddenly switched sign whilst maintaining the same magnitude, the controller would in fact do nothing since the kinetic energies matched. To overcome this problem the feedback error is defined as,

feedback error = $sign(r) \times (|r|r - |\omega|\omega)$

where,

$$r$$
 = Reference angular velocity
 ω = Output angular velocity

Using this modification the feedback structure now becomes,



Figure 2.9 - Modified PI feedback scheme for bi-directional output

The choice of the controller gains (proportional and integral) and stability issues arising from the inherent switching action of the device are discussed in Section 5.3 since the CAD tools developed are used in this selection.

The feedback scheme just described is not the only one that could be used with step-up/step-down mechanisms, however, it is the one that has been successfully used for the prototype (see Section 6.6).

2.3 Conclusions

This chapter has dealt with the switching algorithms and feedback control schemes used for each mechanism. For the step-up mechanism a simple PWM switching algorithm is first explained which is a simple scheme to implement and has its direct counter part in the electrical domain. This algorithm has a drawback, however, in that it creates energy losses by switching on the brake when, generally, the spring has significant kinetic energy. A second algorithm, called resonant switching, attempts to minimise these energy losses by switching the brake when the spring velocity is close to its minimum value. Also discussed in this chapter is an enhancement that can be made to both algorithms that significantly improves the overall performance. This enhancement involves delaying the actuation of the clutch until the velocity of the spring matches that of the load and means that less energy would be wasted in the clutch and also reduces its wear characteristics. A simple feedback structure for the step-up mechanism has also been introduced.

The switching algorithm for the step-up/step-down mechanism encompasses the ideas of clutch matching and resonant switching first introduced for the step-up mechanism, but is more complicated due to the step-up/step-down mechanism's bidirectional output velocity and bi-directional energy transfer capabilities. The feedback structure for the step-up/step-down mechanism is also more complicated for similar reasons.

30

Chapter 3 The Modelling and Simulation of Flexible Drive Mechanisms

This chapter attempts to develop mathematical models for both step-up and stepup/step-down mechanisms. These models must be as accurate as possible whilst still remaining tractable and be physically intuitive. Ideally they should take into account the engagement and disengagement delay inherent in mechanical clutches and brakes, and the finite amount of torque they can provide. In addition the spring's inertia should not be ignored since this has a significant effect on the overall system behaviour. These simulations would then help in further system development, help to pinpoint areas that cause significant energy losses and help controller design and parameter selection. The first section describes the model itself, gives the coupled second order equations that can be used to model step-up mechanisms, and the modified versions that can be used to model step-up/step-down mechanisms. The next section describes the method for the model's numerical solution and also discusses a common problem with simulating switching systems such as these and how this problem can be overcome. The final section describes how the models can be used for efficiency analysis and how energy losses can be attributed to particular components of the device.

3.1 Mathematical Models

An appropriate method of modelling these systems is to divide them into three main components: the *motor*, *spring* and *load*. Each component will have a second order equation associated with it and these equations will be coupled through the operation of the clutches and/or brake. Since each of the main components has bearings associated with them the model for bearing friction will be considered first. This is a simple viscous and coulomb fit [13] with separate viscous and coulomb parameter pairs used for each bearing. This friction model is shown in **Fig. 3.1**.



Figure 3.1 - Force vs. velocity characteristic used to model bearings

Mathematically this function can be expressed as,

$$F = B\omega + Csign(\omega)$$

Where,

F = Frictional force of bearing

- ω = Relative angular velocity of inner and outer races
- B =Viscous friction coefficient
- C =Coulomb friction coefficient

and the sign function is defined as,

$$sign(x) = \begin{cases} +1 & x \ge 0\\ -1 & x < 0 \end{cases}$$

The clutches and brakes are simply modelled, in their "on-state", as coulomb friction devices with the level of friction being the rated dynamic torque, and, in their "off-

state", as devices with no friction. The "on-state" force vs. velocity characteristic is shown in Fig. 3.2.





Mathematically this can be expressed as follows,

$$F(t) = u(t - C_d) \tau_s sign(\omega)$$

Where,

F = Force applied by clutch or brake $\tau_s = Rated dynamic torque of clutch or brake$ $\omega = Relative angular velocity between faces of clutch or brake$ u = State of clutch or brake (1 for "on" and 0 for "off") $C_d = Delay of clutch or brake$

It should be noted how the clutch or brake delay is achieved by simply delaying the operation of the state variable u by the delay of the clutch or brake (C_d) .

The model of the motor is simply defined as an inertia providing a source of torque as a function of velocity. This is a purposefully general model since the mechanism can work with any design of motor.

3.1.1 Three-mass model for the step-up mechanism

Fig. 3.3 shows a schematic diagram of a step-up mechanism and how it is divided into its three components; the *motor*, *spring* and *load*. The *motor* component contains the motor plus flanges etc. up to the start of the spring, the *spring* component the spring and all parts of the clutch and brake that are rigidly attached, and the *load* component contains the load mass plus parts of the clutch that are rigidly attached.



Figure 3.3 - Three mass model representation of step-up mechanism

Combining the equations of motion of each component with the mathematical model of the clutch and brake previously defined we can construct the complete model for the step-up mechanism which is given in **Eqs.** (3.1, 3.2 and 3.3),

$$\tau_m(\dot{\theta}_m) = J_m \ddot{\theta}_m + B_m \dot{\theta}_m + C_m sign(\dot{\theta}_m) + K(\theta_m - \theta_s)$$
(3.1)

$$0 = J_{s}\dot{\theta}_{s} + B_{s}\dot{\theta}_{s} + C_{s}sign(\dot{\theta}_{s}) + K(\theta_{s} - \theta_{m}) + u\tau_{b}sign(\dot{\theta}_{s}) + v\tau_{c}sign(\dot{\theta}_{s} - \dot{\theta}_{l})$$
(3.2)

$$0 = J_{l} \ddot{\theta}_{l} + B_{l} \dot{\theta}_{l} + (C_{l} + \tau_{l}) sign(\dot{\theta}_{l}) + v \tau_{c} sign(\dot{\theta}_{l} - \dot{\theta}_{s})$$
(3.3)

= Motor, load and spring positions $\theta_m, \theta_s, \theta_s$ Where, Motor, load and spring inertias J_m, J_s, J_l = = Viscous friction coefficients B_m, B_s, B_l Coulomb friction coefficients C_m, C_s, C_l = Spring stiffness K = = Torque generated by motor τ_m = Load torque τ_{I} Rated dynamic brake torque = τ_{b}

- τ_c = Rated dynamic clutch torque
- u = Brake state (1=on, 0=off)
- v =Clutch state (1=on, 0=off)

3.1.2 Three-mass model for the step-up/step-down mechanism

A schematic diagram of the three mass model for step-up/step-down mechanisms is shown in **Fig. 3.4**. This, as with the step-up mechanism, consists of three components, the *motor*, *spring* and *load*. The *motor* contains the motor plus components of clutch 1 rigidly attached, the *spring* term contains the spring, connecting rod and all parts of both clutches that are rigidly attached, and the *load* component contains the load mass and parts of clutch 2 rigidly attached.



Figure 3.4 -Three mass model representation of the step-up/step-down mechanism Combining the equations of motion of each component with the mathematical model of the clutch previously defined, we can construct the complete model for the stepup/step-down mechanism which is given in **Eqs.** (3.4, 3.5 and 3.6),

$$\tau_m(\dot{\theta}_m) = J_m \ddot{\theta}_m + B_m \dot{\theta}_m + C_m sign(\dot{\theta}_m) + u_1 \tau_{c1} sign(\dot{\theta}_m - \dot{\theta}_s)$$
(3.4)

$$0 = J_s \ddot{\theta}_s + B_s \dot{\theta}_s + C_s sign(\dot{\theta}_s) + K \theta_s + u_1 \tau_{c1} sign(\dot{\theta}_s - \dot{\theta}_m) + u_2 \tau_{c2} sign(\dot{\theta}_s - \dot{\theta}_l)$$
(3.5)

$$0 = J_{l}\ddot{\theta}_{l} + B_{l}\dot{\theta}_{l} + (C_{l} + \tau_{l})sign(\dot{\theta}_{l}) + u_{2}\tau_{c2}sign(\dot{\theta}_{l} - \dot{\theta}_{s})$$
(3.6)

Where, $\theta_m, \theta_s, \theta_l = Motor$, load and spring positions

J_{m}, J_{s}, J_{l}	=	Motor, load and spring inertias
B_{m}, B_{s}, B_{l}	=	Viscous friction coefficients
C_m, C_s, C_l	Ξ	Coulomb friction coefficients
K	=	Spring rate
τ_m	=	Torque generated by motor (as a function of motor velocity)
τ_l	=	Load torque
$ au_{cI}$	=	Rated dynamic torque of clutch 1
τ_{c2}	=	Rated dynamic torque of clutch 2
u_{I}	=	Clutch 1 state $(1 = on, 0 = off)$
<i>u</i> ₂	=	Clutch 2 state $(1 = on, 0 = off)$

3.2 The Numerical Solution

To find the solution to these equations numerical methods must be used and this is most conveniently done if they are converted to state-space form. The obvious states to choose are the motor position and velocity, the spring position and velocity, and the load position and velocity. The state-space equations for the step-up mechanism are shown in Eqs. (3.7 to 3.13) and for the step-up/step-down mechanism in Eqs. (3.14 to 3.20).

State-space model for the step-up mechanism

$$x_1 = \theta_m, x_2 = \dot{\theta}_m, x_3 = \theta_s, x_4 = \dot{\theta}_s, x_5 = \theta_l, x_6 = \dot{\theta}_l$$
 (3.7)

$$\dot{x}_1 = x_2 \tag{3.8}$$

$$\dot{x}_{2} = \frac{1}{J_{m}} \left[\tau_{m} - B_{m} x_{2} - C_{m} sign(x_{2}) - K(x_{1} - x_{3}) \right]$$
(3.9)

$$\dot{x}_3 = x_4 \tag{3.10}$$

$$\dot{x}_{4} = \frac{1}{J_{s}} \Big[-B_{s} x_{4} - C_{s} sign(x_{4}) + K(x_{1} - x_{3}) - u\tau_{b} sign(x_{4}) - v\tau_{c} sign(x_{4} - x_{6}) \Big]$$

$$\dot{x}_5 = x_6 \tag{3.12}$$

$$\dot{x}_{6} = \frac{1}{J_{l}} \Big[-B_{l} x_{6} - (C_{l} + \tau_{l}) sign(x_{6}) - v \tau_{c} sign(x_{6} - x_{4}) \Big]$$
(3.13)

(3.18)

State space model for the step-up/step-down mechanism

•

$$x_1 = \theta_m, x_2 = \dot{\theta}_m, x_3 = \theta_s, x_4 = \dot{\theta}_s, x_5 = \theta_l, x_6 = \dot{\theta}_l$$
(3.14)

$$\dot{x}_3 = x_4 \tag{3.15}$$

$$\dot{x}_{2} = \frac{1}{J_{m}} \left[\tau_{m} - B_{m} x_{2} - C_{m} sign(x_{2}) - u_{1} \tau_{c1} sign(x_{2} - x_{4}) \right]$$
(3.16)

$$\dot{x}_3 = x_4 \tag{3.17}$$

$$\dot{x}_{4} = \frac{1}{J_{s}} \Big[-B_{s} x_{4} - C_{s} sign(x_{4}) - K x_{3} - u_{1} \tau_{c1} sign(x_{4} - x_{6}) - u_{2} \tau_{c2} sign(x_{6} - x_{4}) \Big]$$

$$\dot{x}_5 = x_6 \tag{3.19}$$

$$\dot{x}_{6} = \frac{1}{J_{l}} \Big[-B_{l} x_{6} - (C_{l} + \tau_{l}) sign(x_{6}) - u_{2} \tau_{c2} sign(x_{6} - x_{4}) \Big]$$
(3.20)

There exists a problem when integrating these systems of equations due to the singularities associated with the clutch, brake and bearing models. These singularities are caused by the *sign* function (see Figs. 3.1 and 3.2). If the equations are solved, as is, with a variable step method the integration algorithm will fall into an infinite loop trying to force the time step to zero in an attempt to overcome this singularity. To solve this problem a tolerance is put on the width of the singularity. If the variable associated with the sign function goes inside this tolerance, and the terms forcing the change in this variable are less then the magnitude of the sign term, then the variable should be "locked" at its current value (by putting the variable's derivative to zero). The variable should only be "unlocked" from this state when the sum of terms affecting its change are greater than the coefficient of its sign function.

Consider Eq. (3.9), this defines the derivative of state x_2 (the motor velocity) and this equation has a singularity caused by the model of motor bearing. To overcome the numerical problems found when the value of x_2 approaches this singularity, simply test for the value of x_2 being less than a small value, around the singularity, and then test the sum of the other terms are less than the magnitude of the

sign function. If this is the case then "lock" its value by setting its derivative to zero, i.e.,

IF $(x_2 < \text{velocity tolerance})$ AND ABS $(\tau_m - B_m x_2 - K(x_1 - x_3)) < C_m)$ $\dot{x}_2 = 0$ /* locked */ ELSE $\dot{x}_2 = \frac{1}{J_m} [\tau_m - B_m x_2 - C_m sign(x_2) - K(x_1 - x_3)]$ /* unlocked */ END

Obviously the same procedure needs to be carried out for all the other sign terms in the set of state-space equations.

The numerical method employed was a 2nd/3rd order variable step Runge-Kutta method [16] and was chosen since it was relatively easy to program. This algorithm was implemented using the "C" programming language and interfaced with the popular mathematical analysis tool MATLAB [17] for plotting and further analysis work. An overview of the simulation software is shown in Fig. 3.5.



Figure 3.5 - Overview of software to perform numerical simulation

The software is divided into three parts; the first part (called the SIMULATOR) implements the Runge-Kutta algorithm, the second, a function called *model()* implements the above state-space equations, and the third a controller function (called *controller()*) performs the appropriate switching algorithm and feedback control if required. The *model()* subroutine is called at every integration time step by

the simulator with the current states (x) and defines the state-space derivatives (\dot{x}) . At the sample frequency for the system the *controller()* subroutine is called with the current states and provides the inputs to the model (these are the clutch and/or brake signals). Appendices C and D show print-outs of the "C" code used to define the physical models for the step-up and step-up/step-down systems and Appendices E and F show the "C" code used to define the controllers for the step-up and step-up/step-down mechanisms.

3.3 Efficiency Analysis

The mathematical models can be used to perform efficiency and power loss analysis using the data from open loop simulations over a range of different output load torques. These simulations will produce a range of different output steady-state velocities and the results of these simulations can be used to estimate power losses in key components of the device as a function of output velocity. A list of these components is shown below,

- 1. Motor bearing
- 2. Spring bearing
- 3. Load bearing
- 4. Input clutch (step-up mechanism) or brake (stepup/step-down mechanism)
- 5. Output clutch

The mathematical models assume the spring to be a 100% efficient device and so for this analysis this will be the case as well. The following equations can be used to estimate power losses,

Input power

input power =
$$\int \tau_m \times \omega_m dt$$
 (3.21)

where, $\tau_m = Motor torque$ $\omega_m = Motor angular velocity$

Output power

output power =
$$\int \tau_l \times \omega_l dt$$
 (3.22)

where, $\tau_l = \text{Load torque}$ $\omega_l = \text{Load angular velocity}$

Motor, spring and load bearing power

bearing power =
$$\int (B\omega + sign(\omega)C)\omega dt$$
 (3.23)

where,

B =Viscous friction coefficient C =Coulomb friction coefficient $\omega =$ Load, spring or motor angular velocity

Clutch or brake power

$$power = \int \tau_c \times u(t - C_d) \times (\dot{\theta}_i - \dot{\theta}_o) dt$$
(3.24)

where,	$ au_{c}$	= Clutch or brake torque	
	и	=	Clutch or brake signal (0 for off, 1 for on)
	C_d	=	Clutch or brake delay
	$\dot{\boldsymbol{ heta}}_i$	=	Clutch or brake input angular velocity
	$\dot{\theta}_{o}$	=	Clutch or brake output angular velocity

The equations outlined above can also be used with experimental data and form the basis of the efficiency analysis performed on both of the test prototypes.

3.4 Conclusion

This chapter has outlined a three mass mathematical model for each of the step-up and step-up/step-down mechanisms. The method of its numerical solution and how this model can be used to predict the amount of energy loss in particular system components has also been discussed. These models proved an invaluable tool in the analysis and development of the two prototypes and the model of the step-up/step-down mechanism and the simulation software form a crucial part of the CAD software discussed in **Chapter 5**.

Chapter 4 The Design and Performance of a Prototype Step-up Drive System

In order to prove that the concept of the step-up mechanism works in practice a prototype was constructed. This chapter describes its construction and analyses its performance. Practical comparisons between the different types of switching algorithms (PWM and resonant) introduced in **Section 2.1** are made and the effects of clutch matching and increasing the motor inertia are assessed. The general mathematical model described in **Chapter 3** is further refined through inclusion of a model for the motor. Methods for the estimation of some key parameters such as the clutch delay and spring inertia etc. are also described and the results presented. The model is verified by comparing the results of open loop tests performed on the prototype with identical tests performed using the model. Overall power efficiency tests are performed on the device and, by using the mathematical model, these losses are attributed to the various components of the mechanism.

4.1 Prototype Construction and Instrumentation

A schematic diagram of the test prototype with its associated instrumentation is shown in Fig. 4.1,



Figure 4.1 - Schematic diagram of prototype with instrumentation

There was no rigorous design procedure undertaken to choose the mechanisms constituent components, they were chosen on the basis of engineering judgement alone. The prototype included the following off-the-shelf components,

Component	Model	Specification
Motor	Printed Motors Ltd,	torque constant = 0.11 Nm/amp
	Model G12M4	
Clutch & brake	Clarke, Model CB 175	static torque = 1.1 Nm
Spring	Lewis spring products	torsional spring rate = 0.22
	LS 800 38	Nm/rad

Table 4.1 - List of Components for the Step-up Prototype

The data sheets for these components are given in **Appendix G**. The motor amplifier, clutch/brake driver and encoder counting electronics were all produced in-house at the university. A photograph of the completed prototype is shown in **Appendix H**.

A PC was used to control the mechanism and log results. This sensed the positions and velocities of the motor, spring and load and produced signals to control the clutch and brake. The motor was always run with a constant input voltage. The PC operated at a sample rate of 500 Hz and implemented a simple Euler differentiation technique to provide velocity signals from the displacement signals provided by the encoders. Using a PC to produce the control signals gave flexibility when testing and implementing various switching algorithms. The controller function

used to control the prototype is shown in **Appendix E** and is in fact identical to the one used in the three-mass mathematical model. This routine was called once at every time step and could provide either PWM or resonant switching and also produced the PI feedback control.

4.2 Prototype Results

This section presents test results from the prototype, compares the two switching algorithms and shows the effect of matching the clutch plate velocities prior to engagement and increasing the motor inertia.

4.2.1 Open and closed loop tests

Results of an open-loop test which accelerated the load mass from rest is given in **Fig. 4.2**. This test used the PWM switching algorithm with a fixed duty cycle of 0.5 and a cycle time of 0.2 seconds. Various cycle times (t_1) were tried and 0.2 seconds was found to produce the highest load velocity. A constant motor voltage of 12 volts was used, much more than this and it was found that either the spring failed or the brake and/or clutch slipped. It can be seen that the load velocity reaches a value almost three times that of the mean motor speed. This result shows that the theoretical idea behind the device works in practice.

The next test performed was varying the duty cycle of the switching algorithm. Fig. 4.3 shows the load mass being accelerated with six different duty cycles ranging from 0.05 to 0.5.

44



Figure 4.2 - Accelerating a mass from rest using PWM switching



Figure 4.3 - Accelerating a mass from rest using duty cycles of 0.05 to 0.5

As theoretically predicted the steady-state output velocity increases with the duty cycle and the output velocity can be controlled by manipulation of this parameter. **Fig. 4.4** shows the mechanism working in closed loop with a reference velocity of

287 rad/s. The feedback gains were tuned manually and the final values chosen were $K_p=0.12 \text{ s}^2/\text{rad}$ and $K_f=4.4\text{e}-5 \text{ s}/\text{rad}$. As can be seen the reference velocity was attained with no overshoot and no detectable steady-state error.



Figure 4.4 - Step-up prototype under closed-loop control with a reference velocity of 287 rad/s

4.2.2 Comparison of PWM and resonant switching

A comparison of between resonant and PWM switching algorithms is given in Fig. 4.5. It will be seen that the terminal velocity is higher using resonant switching inferring that this type of switching produces higher levels of output power and reduces the losses in the device.



Figure 4.5 - A comparison of resonant and PWM switching accelerating a mass from rest

4.2.3 Effect of clutch matching

A comparison of resonant switching with and without clutch matching is given in **Fig. 4.6(a)**, a snapshot of this test showing a few cycles is given in **Fig. 4.6(b)**. As can be seen the dips in velocity just prior to energy transfer have been removed and a significant increase in output velocity has been achieved.



Figure 4.6(a) - Comparison of resonant switching with and without clutch matching



Figure 4.6(b) - Snapshot of comparison of resonant switching with and without clutch matching

4.2.4 Increased Motor Inertia

Fig. 4.2 showed the motor and load velocities for the step-up device with no added motor inertia. As can be seen, the motor velocity varies considerably, from about 140 rad/s to almost 0. Since the motor is being driven by a constant voltage (12 volts) its output power, as a function of velocity, can easily be found by using the motors mathematical model (see Section 4.3.1). This power vs. velocity characteristic is given in **Fig. 4.7**. From this graph we can see that when the motor velocity varies from 0-110 rad/s the power output from the motor is seriously compromised. If the motor velocity can be kept constant, at or around the velocity which provides peak power (about 55 rad/s), the input power to the device will be increased considerably. An obvious way to achieve this is to increase the motor inertia. This will keep the motor velocity steadier and so allow the motor to store energy even when it is not transferring energy directly to the spring. The inertia added took the form of a steel disc 90 mm in diameter and 30 mm in length representing an added inertia of 1.42×10^{-3} kg m². This resulted in a 10 fold increase in the effective motor inertia and the results of using this increased motor inertia can be seen in **Fig. 4.8**.



Figure 4.7 - Power/velocity characteristic of motor driven with a constant 12 volts



Figure 4.8 - Comparison of device with and without added motor inertia

It is apparent that the motor velocity ripple has been reduced significantly and that its average velocity is at or around the velocity of peak power (55 rad/s). This increased input power has resulted in a considerable increase in load velocity (about 50%) even though there is a slight reduction in initial acceleration because the motor is accelerated from rest at time zero and this takes longer with the increased inertia. This result demonstrates that, when the motor has only a narrow velocity band where it produces significant power, it is essential that it possess a large enough inertia to keep its velocity within these limits.

4.3 Mathematical Model

This section develops the model outlined in Section 3.1.1 specifically for this prototype. It defines the motor model used with this prototype and also describes how the values of the various parameters were arrived at.

The motor used with this prototype is a printed armature DC permanent magnet electric motor (see Appendix G for data sheet) and the classical model for

this type of motor is given below. Note that the armature inductance is neglected since it is very small.

$$\tau_m = \frac{K_t \left(V_s - K_e \theta_m \right)}{R} \tag{4.1}$$

where,

$$V_s$$
 = Motor supply voltage
 K_e = Motor emf constant
 K_t = Motor torque constant
 R = Motor resistance

This model (Eq. (4.1)) combined with the three-mass model defined in Section 3.1.1 represents the complete model of the system. Before this model can be simulated values for the model parameters must be obtained. Some of them can be found from manufacturer's data but others must be found using experimentation. Parameters found from manufacture's data include K_e , K_i , R, J_m , τ_b and τ_c , with all other parameters requiring experimentation. The following sub-sections describe these experiments in detail.

4.3.1 The Motor Bearing Friction and Motor Model

This experiment attempts to find the motor bearing friction coefficients and also performs some simple tests to verify the motor model. The experiment simply measured the motor current at different unloaded steady-state motor velocities. At this steady-state the motor acceleration is zero and so the model (**Eq. (4.1**)) becomes,

$$\tau_m = K_I I_m = B_m \dot{\theta}_m + C_m sign(\theta_m)$$
(4.2)

where, $I_m =$ Motor current

Since K_t is a constant Eq. (4.2) should produce the friction characteristic shown in Fig. 3.1. A graph of the measured current vs. velocity data is show in Fig. 4.9. Along with the measured points a line of best fit has also been plotted.



Figure 4.9 - Motor current vs. steady-state unloaded motor velocity

As can be seen the results justify the original choice of bearing model and from this graph the friction coefficients can easily be found. Using an average of the gradients in the positive and negative directions the value of viscous coefficient is,

$$B_m = 7.95e-5$$
 Nm s/rad

and using an average of the y axis intercepts the coulomb friction can easily be found as,

$$C_m = 0.0246 \text{ Nm}$$

These values were found using the manufacturers supplied value for K_t of 0.1101 Nm/amp.

To verify the complete motor model different open-loop step tests were performed using four different input voltages of 12, 15, 18 and 21 volts and the inertia of

52



1.42x10⁻³ kg m² attached. The comparison between the model and experiment is shown in Fig. 4.10 below,

Figure 4.10 - Comparison of motor model at four different input voltages

As can be seen the fit between the model and real system is very good both in terms of its steady-state velocity and transient behaviour.

4.3.2 Spring Stiffness and the Clutch and Spring Inertia

These experiments attempt to find the spring stiffness, the inertia of the spring and the clutch/brake components that are rigidly attached to it (J_s) , and in addition, the inertia of the clutch components (J_c) that are attached to the load. The motor end of the spring was rigidly held and with the clutch engaged and the brake disengaged the load mass was allowed to freely oscillate. Tests were performed with two different load masses (called "A" and "B") with inertia's J_A and J_B and two different springs (called "C" and "D") which had different spring rates K_C and K_D . Another test was performed with the clutch disengaged so that only J_s would oscillate. These tests together would produce five values of the ratio of spring stiffness to effective inertia and from these equations the required parameters could be found.

A schematic diagram of this set-up is shown in Fig. 4.11,



Figure 4.11 - Schematic diagram of clutch and spring inertia experiment

With this arrangement the system will obey the following differential equations,

$$J\ddot{\theta} + B\dot{\theta} + K\theta + Csign(\dot{\theta}) = 0$$
(4.3)

where,

 θ = Displacement of oscillating mass J = Inertia of oscillating mass B = Total viscous friction term for bearings K = Spring rate C = Total coulomb friction term for bearings

Re-arranging this in terms of the acceleration we have,

$$\ddot{\theta} = \frac{B}{J}\dot{\theta} + \frac{K}{J}\theta + \frac{C}{J}sign(\dot{\theta})$$
(4.4)

By measuring the displacement of the oscillating mass, and differentiating to achieve velocity and acceleration signals, the parameters B/J, K/J and C/J can all be found by using a simple least-squared estimator [14]. However before this can be achieved something needs to be done about the non-linear function *sign* appearing in Eq. (4.4). This can be accomplished by separating the data between those points for which $\dot{\theta} \ge 0$ and those for which $\dot{\theta} < 0$. In this case we will have two equations with each

equation having its own set of experimental points. These equations are shown below,

$$\dot{\theta}_{+} = \frac{B}{J}\Big|_{+} \dot{\theta}_{+} + \frac{K}{J}\Big|_{+} \theta_{+} + \frac{C}{J}\Big|_{+}$$
(4.5)

$$\dot{\theta}_{-} = \frac{B}{J} \bigg|_{-} \dot{\theta}_{-} + \frac{K}{J} \bigg|_{-} \theta_{-} - \frac{C}{J} \bigg|_{-}$$
(4.6)

where, $\ddot{\theta}_{+}, \dot{\theta}_{+}, \theta_{+} = \text{Set of experimental points where } \dot{\theta} \ge 0$ $\ddot{\theta}_{-}, \dot{\theta}_{-}, \theta_{-} = \text{Set of experimental points where } \dot{\theta} < 0$

We will now achieve two estimates of B/J, K/J and C/J, one for the θ_+ data and another for the θ_- data. If the original model (Eq. (4.3)) was correct then these estimates should be identical. In practice this was not the case but the deviation was very small and as a consequence the mean of the two parameter estimates was used as the "best" estimate. For the purposes of this analysis the values of B/J and C/J are superfluous. However they are required when comparing the model (Eq. (4.4)) to the experimental data. This is achieved by performing an analytical time series solution to Eq. (4.4) and comparing the displacements, velocities and accelerations with the original data.

Five oscillation tests were performed. The first four had the brake disengaged and the clutch engaged and used all combinations of both springs and both load masses, and the last one had the clutch disengaged, so only the brake inertia was oscillating, and used spring "D". Fig. 4.12 compares the experimental data with that of the fitted model using the displacement, velocity and acceleration data from the test using spring "D" and load mass "A". As can be seen the quality of fit is very good and was equally as good for all the other test performed.



Figure 4.12 - Experimental and fitted data for spring "D" and load mass "A"

For the five tests the following values of K/J were found,

$$\frac{\hat{K_c}}{J_s + J_c + J_A} = 837.9$$
(4.7)

$$\frac{K_C}{J_s + J_c + J_B} = 1961$$
 (4.8)

$$\frac{K_D}{J_s + J_c + J_A} = 340.1$$
 (4.9)

$$\frac{K_D}{J_s + J_c + J_B} = 796.0$$
 (4.10)

$$\frac{K_D}{J_s} = 4252$$
 (4.11)

Since J_A and J_B are known (J_A is a metal disk 90 mm in diameter and 13.5 mm thick giving an inertia of 6.76×10^{-4} kg m² and J_B an aluminium disk of the same size having an inertia of 2.44×10^{-4} kg m²) these five equations have five unknowns and hence are solvable. In fact **Eqs.** (4.7, 4.8 and 4.11) provide three equations in three unknowns but having the extra equations (provided by using a different spring) will provide extra confidence that the final result is correct. Solving for $J_s + J_c$ gave the results,

$$J_s + J_c = 7.854 \times 10^{-5} \text{ kg m}^2$$
 (using Eqs. (4.7, 4.8))

and

$$J_s + J_c = 7.851 \times 10^{-5} \text{ kg m}^2$$
 (using Eqs. (4.9, 4.10))

The closeness of these two results is encouraging and gives confidence to the result as a whole. Using the mean of these two values and solving Eqs. (4.9, 4.10 and 4.11) yields these values for J_c , J_s , K_c and K_D ,

$$J_c = 1.81 \text{ x } 10^{-5} \text{ kg m}^2$$
 $J_s = 6.04 \text{ x } 10^{-5} \text{ kg m}^2$
 $K_C = 0.63 \text{ Nm/rad}$ $K_D = 0.26 \text{ Nm/rad}$

which will be those used in the mathematical model.

4.3.3 Clutch and Brake Bearing Friction Coefficients

The clutch/brake unit used in this mechanism has identical bearings for both its clutch and brake sides. Thus only one experiment was performed, which found the clutch bearing coefficients, and these coefficients used for the brake. To find these coefficients a load mass of known inertia was connected to the output shaft and was accelerated to a high speed and allowed to coast to a halt (the clutch and brake were both disengaged). An optical encoder measured the output displacement and this signal was differentiated twice to achieve acceleration data. Since the inertia of the output mass was known the frictional force applied could easily be found. Due to the problems of discretisation noise from the optical encoder data, the following procedure was followed to estimate the acceleration,

i) Obtain velocity data from the displacement data using a normal Euler approximation.

 ii) Group together sets of velocity points and choose a representative acceleration point as the gradient of the line of best fit and a representative velocity point as the mean velocity.

This approach gives a reasonable approximation since the velocity is only changing slowly. The test had approximately 1000 velocity points and sets of forty points were grouped together. This test is shown in **Fig. 4.13**.



Figure 4.13 - Grouping of velocity points: representative acceleration taken as gradient of line of best fit and representative velocity as the mean

This grouping of points produced a set of 25 acceleration/velocity values. Multiplying the acceleration values by the load inertia produced the graph shown in **Fig. 4.14**. Also shown on this graph is the line of best fit and shows that the original choice of a viscous and coulomb fit to be a good one. The viscous coefficient was taken to be the gradient of the line of best fit and the coulomb coefficient as the intercept with the velocity = 0 line. These are quoted below,

> Coulomb coefficient = 0.0075 Nm Viscous coefficient = 3.32×10^{-5} Nm s/rad

The data shown in **Fig. 4.14** is particularly interesting since it also shows the effect of so called Stribeck friction [15] a well known phenomenon found in lubricated metal surfaces as a rise in the level of friction at low speeds.



Figure 4.14 - Set of 25 frictional force/velocity points plotted with line of best fit

4.3.4 Clutch and Brake Delay

To estimate the clutch and brake delay steady-state data from the actual operation of the device was used. The engagement delay was measured by comparing the signal sent to the brake and velocity trajectory of the spring, and the disengagement delay by looking at the signal sent to the clutch and the velocities of the load and spring. These signals are shown in **Fig. 4.15**. It is clear that no noticeable effect in terms of a reduction in spring velocity is apparent until approximately 6 ms after the brake engagement signal was applied. In a similar way it can be seen that the load and spring velocities were found not to differ significantly until at least 6 ms after the clutch disengagement signal was sent.

Since identical electrical drives and almost identical components are used for both the clutch and brake then it is quite easy to justify using the engagement delay,

59

calculated for the brake, for the clutch also, and the disengagement delay, calculated for the clutch, for the brake as well.

Fig. 4.16 shows the voltage and current signals emanating from the clutch/brake drivers during a single engagement/disengagement cycle. These drivers have current feedback and a maximum supply voltage of ± 50 volts. Since the current rise takes at least 5 ms, due to the coil inductance and limited supply voltage, the delay until full torque is applied will be expected to take at least this long. Also note the voltage signal to the driver. Initially it shoots up to almost 50 volts (the supply voltage) and stays between 45-50 volts until the correct supply current is reached (and shoots to -50 volts when the clutch/brake is released). It is obvious that the drivers are doing the best they can, within the confines of the supply voltage, to achieve the fastest engagement and disengagement times for this clutch/brake unit.



Figure 4.15 - Estimation of brake engagement delay and clutch disengagement delay


Figure 4.16 - Voltage and current supplied to clutch/brake unit through a single engagement/disengagement cycle

4.3.5 Comparison of Model with Real System

Table 4.2 summarises the results of the previous four sub-sections and lists the values of all the coefficients and their sources as used in the model of the step-up mechanism. In the simulations, the same controller "C" code was used as that in the real system. This ensured that errors between the model and the real system were purely in the physical system model and not in the controller implementation.

Parameter	Description	Source M-manufacturers data E-Experiment	Value
K_t Torque constant		Μ	0.1101 Nm/amp
K_e EMF constant		Μ	0.1043 Vs/rad
J_m	Motor inertia	M	1.4×10 ⁻⁴ kg m ²
R	Motor resistance	Μ	1.1 ohms
K _{cm}	Motor amplifier gain	E	2.94
C_m	Motor friction coeff.	E	0.0246 Nm
B _m	Motor friction coeff.	E	7.95×10 ⁻⁵ kg m ²
K	Spring rate	E	0.26
J_s	Spring inertia	Е	6.04×10 ⁻⁵ kg m ²
C_s	Spring friction coeff.	E	0.0075
B _S	Spring friction coeff.	E	3.33×10 ⁻⁵ kg m ²
J_{c}	Clutch inertia	E	1.81×10 ⁻⁵ kg m ²
J_1	Load inertia	E	6.76×10 ⁻⁵ kg m ²
	Load friction coeff.	Е	0.0075
B_l	Load friction coeff.	E	3.33×10 ⁻⁵ kg m ²
τ_{L}	Brake torque	Μ	1.1 Nm
B	Brake delay	Е	0.006 ms
τ_{h}	Clutch torque	Μ	1.1 Nm
C_d	Clutch delay	E	0.006 ms

Table 4.2 - Summary of coefficients used in the model of the step-up prototype

Fig. 4.17(a) shows a comparison between the model and the real system for an open loop test using PWM switching, and Fig. 4.17(b) shows a comparison between them using resonant switching.



Figure 4.17(a) - Comparison of Model and Real System using PWM Switching



Figure 4.17(b) - Comparison of Model and Real System using Resonant Switching

As can be seen for the conditions given they both show a remarkable level of agreement. It will be noticed that the exact timing of the energy transfer cycles varies slightly in the resonant switching example. This is hardly surprising since the overall

cycle time is dependent upon the time taken to transfer the energy from the spring to the load, and any small errors in the modelling of this will have a cumulative effect. Having said this the simulation and real system are less than one cycle different after 35 cycles.

4.4 Efficiency Analysis

This section describes the efficiency tests performed on the device. For these experiments a torque transducer was attached to the output shaft of the device and a variable friction mechanism attached to the output shaft of the torque transducer. This set up is shown in **Fig. 4.18**.



Figure 4.18 - Step-up mechanism with output power measuring equipment

Repeated open loop step tests, similar to those shown in **Fig. 4.2**, were performed with the variable friction mechanism being adjusted for each run. The motor had the extra inertia attached and the device was operating using the resonant switching algorithm with clutch matching enabled. These runs created a different load torque and hence a different steady-state load velocity for each run. Twenty-five runs were performed with steady-state load velocities ranging from about 115 to 320 rad/s.

From this data the output power could easily be calculated as the following,

$$power = \int \tau \times \omega \, dt \tag{4.12}$$

where,

 τ = Output torque ω = Load angular velocity

The output power of the motor (and hence input power to the step-up mechanism) was also measured in a similar way. In this case the friction device and torque transducer were connected directly to the output shaft of the motor and a set of steady-state torque vs. velocity readings were measured. The data produced from this test was used to generate a table from where the torque produced by the motor, at a given motor velocity, could be linearly interpolated.

Fig. 4.19 shows the level of input and output power and also shown is the percentage efficiency (100×(input power/output power)) for the mechanism.



Figure 4.19 - Measured input power, output power and overall efficiency for step-up mechanism

As can be seen surprisingly high efficiencies are apparent at velocities below about 160 rad/s (>60%) and from then on trail off linearly to zero at about 320 rad/s. It would be now very beneficial to know what component or components are the cause

of this apparent power loss. The step-up mechanism can be divided into four main components that each cause known power losses and these are listed below,

- i) The spring bearing (due to friction)
- ii) The brake (due to slipping)
- iii) The clutch (due to slipping)
- iv) The load bearing (due to friction)

The next stage in the analysis is to use the mathematical model developed in Section 3.1 to find out what amount of the power loss can be apportioned to these components. The previous section found accurate physical parameters for this device and when used in the three-mass model produced surprisingly comparable results to the prototype. Using these parameters the power losses can be apportioned to the listed components as outlined in Section 3.3. Results of using this procedure on the 25 efficiency runs given in Fig. 4.19 are shown in Fig. 4.20. Also shown is the so called unmodelled loss which is the difference between the total power loss and the total losses attributed to the four components listed.



Figure 4.20 - Calculated losses attributed to the various components of step-up mechanism

It can be seen that the losses associated with the clutch and brake are relatively low (under 3 watts) also roughly constant or at least not dependent on the output velocity. This low value is the result of using the clutch matching algorithm, which delays clutch actuation until the velocity of the mating faces match, and the resonant switching algorithm, which delays brake actuation until the spring velocity is close to zero.

As can be seen the unmodelled loss constitutes a very high proportion of the total losses from about 160 rad/s onwards, and almost matches the level of power loss caused by the load bearing. This discrepancy between the model and the real system needs to be investigated since **Figs. 4.17(a)** and **4.17(b)** show excellent agreement. The first efficiency run (with a 320 rad/s output velocity) had no friction and a large value of unmodelled power loss and so is a good candidate to compare in more detail with the model.

When this comparison was first done it was obvious that serious brake slippage occurred in the model which was not present in the real system. For this reason the static torque of the brake was measured using a calibrated torque transducer and found to be in the region of 1.7 to 2 Nm (dependent on the relative orientation of the faces) which was significantly larger than the torque quoted in the manufacturer's data (1.1 Nm as used in the model). The torque of the clutch was also measured and found to be between 1 and 1.1 Nm and so roughly agrees with the manufacturer's data. For this reason the model's value of brake torque was set to 1.7 and the simulation repeated. The results of this are shown in **Fig. 4.21** which shows that there is a significant difference in output velocity (about 70 rad/s) between the simulation and the real system. In addition, there is also a large discrepancy in both the spring extension and the motor velocity.

67



Figure 4.21 - Discrepancy between model and real system for 320 rad/s efficiency test

This discrepancy suggests that some unmodelled behaviour has crept into the system between the comparisons shown in **Figs. 4.17(a)** and **4.17(b)** and that shown in the figure above. This must mean that either the model parameters or the model structure or both need changing. The most obvious difference between these comparisons is the addition of the extra inertia to the motor. This has the effect of keeping the motor velocity more constant and as a consequence allow more power to be extracted. As a result of this the maximum spring extension increases from about 4.7 radians in **Figs. 4.17(a)** and **4.17(b)** to about 6.7 radians in **Fig. 4.21**. The spring is modelled as a device having no energy loss whatsoever which in practice is not the case and as the extension increases the losses can also be expected to increase. **Fig. 4.22** shows the spring vs. displacement curve for the spring used in these tests. A torque transducer was used to measure torque and an optical encoder used for the displacement.



Figure 4.22 - Torque vs. displacement for spring used in efficiency tests

This figure shows a very good linear fit up to about five radians at which point the measured torque tails off. This is an important result since it shows that the spring must be entering its plastic deformation region where energy is lost to heat and the break down of its molecular structure. **Fig. 4.23** shows the maximum spring extension for each of the efficiency runs,



Figure 4.23 - Maximum spring extension for the 25 efficiency runs

This graph clearly shows that where the step-up mechanism showed acceptable efficiencies and a good fit to the model (approximately 160 rad/s and below) the spring was always within its linear region but when it entered its plastic deformation region significant and unmodelled power losses were introduced. It is also apparent that the unmodelled loss is roughly proportional to the maximum spring extension. It is important that as the speed of the load increases the part of the spring used to transfer energy shifts from about 1.3 to 4.8 radians, for the 115 rad/s test, to about 4.8 to 6.7 radians for the 320 rad/s test. An example of this is shown in **Figs. 4.24(a)** and **4.24(b)**. This result is significant since for the higher speed test 1.8 out of the 2.9 radians of spring extension, used to transfer energy (about 62%), is inside the plastic deformation region. As the load velocity increased from 160 to 320 rad/s the percentage of the spring extension used to transfer energy increases from 0 to about 62% and so this would clearly account for the linear relationship between unmodelled loss and steady-state load velocity.

To try and simulate losses in the spring the viscous friction term was increased. This will have a more pronounced effect at higher spring velocities which

70

also means higher spring extensions. A comparison of the model and the real system using a viscous friction term, B_s , of 3.5e-4 Nm s/rad is shown in Fig. 4.25. Even though the load velocity is slightly higher it is still noticeable that agreement is shown in both the motor velocity and spring extension.



Figure 4.24(a)- Part of spring used for energy transfer in the 115 rad/s test



Figure 4.24(b) - Part of spring used for energy transfer in the 320 rad/s test



Figure 4.25 - Model and real system using a larger spring friction term

Using the enlarged figure for spring viscous friction the power losses can now be recalculated and these are shown in **Fig. 4.26**. The spring power loss has now significantly increased but the unmodelled power loss has reduced by roughly the same amount. Even though we cannot be sure that the spring is a major contributor to power losses when operated outside its linear region, the model agreement shown in **Fig. 4.25** strongly suggests this. Assuming this to be the case we can use the mathematical model to predict the performance of the device assuming that a larger capacity spring was used which did not enter its plastic deformation region. The predicted efficiency plot is shown in **Fig. 4.27** and the calculated losses attributed to the step-up mechanism's various components are shown in **Fig. 4.28**. It is obvious now that although the overall efficiency has improved the power losses caused by the brake have increased dramatically. This is because the brake is slipping due to the larger spring extensions, found at velocities greater than 320 rad/s, and means the mechanism needs a more powerful brake if it is to run efficiently at these speeds and powers.



Figure 4.26 - Calculated losses attributed to various components using a larger spring friction term



Figure 4.27 - Simulated efficiency assuming a spring working completely within its linear region



Figure 4.28 - Simulated losses assuming spring working completely within its linear region

4.5 Conclusions

The operation of a prototype switched mode step up mechanism has been described and it has been shown that the concept can be made to work in practice. Output velocities greater than the input can be achieved and closed loop control of output velocity is possible. By taking account of the imperfections of the system components and modifying the controller appropriately, it is possible to greatly improve the performance. In particular, resonant switching and clutch matching largely eliminate clutch and brake slip and the associated losses, whilst increasing the motor inertia ensures that energy is extracted from the drive motor in an efficient manner.

The three mass mathematical model developed in Section 3.1 showed surprisingly good agreement with the actual device when operating under both PWM and resonant switching strategies. Increasing the motor inertia and performing efficiency tests showed that the device had a disappointing overall efficiency curve even though there was a narrow band of output velocities (115 to 160 rad/s) where the efficiency was an acceptable 75-80%. However by using the mathematical model it was possible to assign the losses to the various components of the device and in particular show that the device was loosing a lot of power which was not accounted for in the model. It was also shown that the most likely candidate for this was the spring since it was being used well within its the plastic deformation region when the losses were greatest.

Using the mathematical model and assuming the mechanism had a larger capacity spring, that always stayed within its linear region, a revised efficiency curve was put forward. This had an improved range of acceptable output velocities (115 to 240 rad/s) but also showed that these improvements could only be realised if the torque of the brake was increased.

The construction of this prototype has highlighted the need for some form of design methodology to aid the design process. It is vital, for a given system

75

performance, (max. power, max. velocity etc.) to know what capacity and rate of spring is required and also what size the brake and clutch need to be. The interaction between component parameters is not straightforward e.g. increasing the spring capacity will then change its inertia which might mean a larger brake is needed, which will also change the effective spring inertia, and affect the energy packet transfer rate etc. etc. The designer can never be sure that a given configuration is optimum or even whether a required system performance is attainable from the components available. Even though a design methodology does not exist for step-up devices it does exist for step-up/step-down devices and is the subject of **Chapter 5**.

Chapter 5 The Computer Aided Design of Step-Up/Step-Down Mechanisms

The component parameters in switched mode mechanisms have complex and interacting relationships with regard to system performance. It is the purpose of computer aided design to come up with combinations of components that give the required system performance without having to perform the painstaking task of building prototypes. The design process, as outlined in this chapter, consists of three main elements: component selection, feedback gain selection and system simulation. A schematic diagram of how these elements are combined to produce the overall design process is given in **Fig. 5.1**.

The starting point is to decide on the specification required of the device, usually in terms of maximum power and output speed etc. The next stage is to select system components, this is achieved by determining families of component parameters that can produce the same system performance and to select the most appropriate set. It might well be that at this stage no components exist that possess the required properties in which case the specification will have to be amended if further progress is to be made. Assuming a set of parameters has been chosen then the open-loop simulation can take place. This simulation is based on the three-mass model developed in Section 3.1 and allows the designer to take account of clutch delay and sample rate on the performance of the system. If appropriate data is available then the effect of bearing friction and spring losses can be assessed on the overall efficiency of the device. If the designer is happy with the open-loop performance then the feedback gains can be selected and the simulation software used to assess the closed-loop performance of the system. When the designer is happy with the closed-loop performance, construction of the prototype can take place.

A MATLAB [17] toolbox has been developed to perform many of the tasks in the design of these devices. This chapter describes how to use the relevant routines in the toolbox in a sub-section at the end of each section describing a separate stage in the design process. Section 5.1 deals with component selection, Section 5.2 system simulation and efficiency analysis, Section 5.3 controller design and feedback gain selection and Section 5.4 runs through an example design to summarise the ideas put forward in this chapter. Appendix I gives a functional description of all the routines contained in the MATLAB toolbox.



Figure 5.1 - Schematic diagram of the design process

5.1 Component Selection

The three most important parameters that determine system performance are: the maximum clutch torque, the rate of the spring and the inertia of the spring plus all components that are rigidly attached. The spring plus components that are rigidly

attached (i.e. the clutch rotors and connecting rod etc.) will frequently be referred to as the *spring system*. The mathematical procedure outlined is capable of constructing *families* of parameter values that all produce the same system performance. In this way the designer can have the maximum choice in component selection at an early stage in the design process. This analysis also allows the feasibility of the design to be assessed given the spring and clutch technology that is currently available. The analysis assumes the device to be controlled by the resonant switching algorithm as described in **Section 2.2**.

As with most design processes the starting point is the desired specification for the device and for the purposes of this analysis must be defined using the following quantities,

- a maximum motor power (P) and mean motor speed (V_m) ,
- a maximum output speed (ω_{max}) , and either,
- a nominal output velocity ripple, ω_{rip} defined at a nominal minimum velocity, ω_{min} and load inertia, J_l ,

or,

• a minimum energy switching rate, Sr_{min} for the device

It should be noted that this specification is only one way to express the performance of the device. It may be that an alternative specification will be of more use to a particular designer, and that an alternative procedure can be found to generate suitable parameters which may be better or worse in different circumstances.

5.1.1 Mathematical Analysis

To formulate a mathematical model suitable for this analysis the following assumptions are made.

• All bearings are frictionless

- Clutch switching times are negligible
- Motor velocity is constant
- Output power = input power (i.e. the device is 100% efficient)

One of the most important design specifications for these devices is the amount of steady-state velocity ripple produced. This is defined in **Fig. 5.2**. The factors affecting steady-state ripple are the energy switching rate for the device (that is the rate at which energy packets are transferred from the motor to the load), the average output velocity, the output inertia and the total power transferred.



Figure 5.2 - Definition of steady-state velocity ripple

The mean deceleration over one energy transfer cycle is,

deceleration = $\omega_{rip} \times Sr_{min}$

and from *torque* = *inertia*×*angular acceleration* we have,

$$\frac{load \ torque}{J_{i}} = \omega_{rip} \times Sr_{min}$$

The output power can be approximated by *load torque* ×average output velocity and so we have that the switching rate at the velocity ω_{min} is defined by the equation,

$$Sr_{\min} = \frac{P}{\omega_{\min}\omega_{rip}J_{l}}$$
(5.1)

where,

 $Sr_{min} =$ Minimum switching rate P = Output power of device (= power of motor assuming 100% efficiency) $\omega_{min} =$ Nominal minimum output velocity $\omega_{rip} =$ Output velocity ripple $J_1 =$ Output inertia

It must be realised however that this equation shows that the higher the output velocity the smaller the amount of ripple. Hence it is better to define ω_{min} as small as practicable and then ω_{rip} will be, in effect, a maximum for operating range of the device.

Let us now consider the time taken for the device to complete one energy transfer cycle. This will consist of an energy storage stage plus an energy transfer stage and is shown in **Fig. 5.3**.



Figure 5.3 - Approximation for the time to complete one cycle

The time taken for the energy storage stage is the displacement moved by the spring divided by the velocity of the motor and so we have,

$$t_{es} = \frac{(x_f - x_s)}{V_m}$$
(5.2)

and the time taken for the energy transfer stage can be approximated by the time taken for one complete free oscillation of the spring and so,

$$t_{et} = 2\pi \sqrt{\frac{J_s}{K}}$$
(5.3)

Combining these two results we have an approximation for the time taken to complete one cycle which is given by Eq. (5.4) below,

$$t = t_{es} + t_{et} = 2\pi \sqrt{\frac{J_s}{K}} + \frac{(x_f - x_s)}{V_m}$$
(5.4)

where,

$$t =$$
 Time taken to complete one cycle
 $K =$ Spring rate
 $J_s =$ Spring system inertia
 $x_f, x_s =$ Start and finish extension of spring during energy
storage phase
 $V_m =$ Velocity of motor

The switching rate will simply be one over the time taken for one cycle which from Eq. (5.4) is the following,

$$Sr = \frac{\sqrt{K}V_m}{V_m 2\pi\sqrt{J_s} + \sqrt{K}(x_f - x_s)}$$
(5.5)

Now let us consider the power that is actually transferred. This can be approximated as the *energy packet size* \times *switching rate*. The energy packet size will simply be the energy stored in the spring immediately after the energy storage phase has completed. These ideas are expressed mathematically in Eq. (5.6) below,

$$P = \varepsilon_p Sr = \frac{1}{2} K (x_f^2 - x_s^2) Sr$$
(5.6)

where

$$\varepsilon_p = \text{Energy packet size}$$

An approximation for x_s (the extension in the spring at the start of the energy storage phase) can be found if we consider the total energy in the spring at the finish of the energy transfer phase. Assuming zero losses in the spring and bearings this energy can be equated to the total energy just prior to the start of the next energy storage phase and so we have,

$$\frac{1}{2}J_{s}\omega_{l}^{2} = \frac{1}{2}Kx_{s}^{2} + \frac{1}{2}J_{s}V_{m}^{2}$$
(5.7)

where, $\omega_l = \text{Load velocity}$

Re-arranging we have,

$$x_{s}^{2} = \frac{J_{s}}{K} (\omega_{l}^{2} - V_{m}^{2})$$
(5.8)

and substituting Eq. (5.8) into Eq. (5.6) we have,

$$x_{f} = \sqrt{\frac{2P}{Sr\,K} + \frac{J_{s}(\omega_{l}^{2} - V_{m}^{2})}{K}}$$
(5.9)

Now x_f will represent the maximum extension of the spring during a cycle and hence the maximum torque seen by the input clutch for this cycle is,

$$\tau_{c} = K x_{f} = \sqrt{\frac{2 P K}{S r} + K J_{s} (\omega_{l}^{2} - V_{m}^{2})}$$
(5.10)

We now require an expression defining the switching rate, Sr, in terms of K and J_s . Substituting for x_f (from Eq. (5.9)) and x_s (from Eq. (5.8)) into Eq. (5.5) and after much re-arrangement we have,

$$K = \frac{\left[Sr\sqrt{J_{s}}\left(V_{m}2\pi - \sqrt{\omega_{l}^{2} - V_{m}^{2}}\right) + \sqrt{2PSr + J_{s}\omega_{l}^{2}Sr^{2}}\right]^{2}}{V_{m}^{2}}$$
(5.11)

This equation is capable of generating values of K and J_s that will produce a chosen value of switching rate only needing the known variables P, V_m and ω_l . This equation can be re-arranged so that J_s and Sr are the explicit variables, i.e. J_s is the solution to the quadratic equation,

 $aJ_s^2 + bJ_s + c = 0$

where,

$$a = Sr^{4} (\omega_{l}^{2} - V_{m}^{2} - A^{2})^{2}$$

$$b = 2Sr^{2} (\omega_{l}^{2} - V_{m}^{2} - A^{2})B - 4A^{2}V_{m}^{2}K$$

$$c = B^{2}$$
where $A = V_{m} 2\pi - \sqrt{\omega_{l}^{2} - V_{m}^{2}}$ and $B = 2PSr - V_{m}^{2}K$

$$(5.12)$$

and Sr is the solution to the quadratic equation,

$$aSr^2 + bSr + c = 0$$

where,

$$a = J_{s} (\omega_{l}^{2} - V_{m}^{2} - A^{2})$$

$$b = V_{m} \sqrt{KJ_{s}} A + 2P$$

$$c = -KV_{m}^{2}$$

where $A = (V_{m} 2\pi - \sqrt{\omega_{l}^{2} - V_{m}^{2}})$
(5.13)

Eqs. (5.11 and 5.12) are very important since they generate two of the key design parameters (K and J_s) that will generate a guaranteed minimum switching rate ($Sr = Sr_{min}$) at the given load velocity ($\omega_l = \omega_{min}$), motor power P and motor velocity V_m .

The maximum spring extension can be found using Eq. (5.9). From this equation it can be seen that the spring extension will have a maximum value when the load velocity is a maximum ($\omega_l = \omega_{max}$). The switching rate at this velocity will increase but can be determined using the values of K and J_s already found and Eq. (5.13). Once the maximum spring extension has been determined the input clutch torque can easily be found from Eq. (5.10). One last thing to note about component selection is that from symmetry arguments the torque of the output clutch should be the same as that found for the input clutch.

5.1.2 Sizing the spring

The mathematical analysis so far described only characterises the spring in terms of its spring rate and maximum extension. Nothing has been said about the size of the spring, that is, the dimensions it must be to efficiently hold the required energy. The following analysis assumes the designer wishes to use a torsional helical spring of circular cross-section and constant coil diameter. The key dimensions of this type of spring are shown in **Fig. 5.4**.



Figure 5.4 - Key dimensions of torsional helical spring

For these springs the torsional spring rate is defined as,

$$K = \frac{d^4 E}{64 DN} \tag{5.14}$$

and the volume of the spring is,

$$V = \pi^2 DNd^2 \tag{5.15}$$

where,

d = Wire diameter D = Coil diameter K = Torsional spring rate N = Number of active coils V = Spring volume

A torsional helical spring is subject to bending stresses and so a standard design equation relating the ratio of maximum energy to volume [18] can be used and is given below,

$$\frac{\varepsilon}{V} = U \frac{\delta_f^2}{E}$$
(5.16)

where, $\varepsilon = \text{Energy capacity of spring}$ $\delta_f = \text{Yield stress (maximum stress without permanent set)}$ U = 1/8 (for round cross sections in uniform circular bending)

This formula indicates that the spring capacity increases with the square of the yield stress for the material. Re-arranging this formula in terms of the energy capacity we have,

$$\varepsilon = VU \, \frac{\delta_f^2}{E} \tag{5.17}$$

The energy capacity of the spring can be defined in terms of the design parameters K and x_{max} (maximum spring extension) as,

$$\varepsilon = \frac{1}{2} K x_{\max}^2$$
 (5.18)

Substituting Eq. (5.18) into Eq. (5.17) we have,

$$V = \frac{E}{U\delta_f^2} \left[\frac{\gamma_2}{K} K x_{\max}^2 \right]$$
(5.19)

Equating the l.h.s of this equation with that of Eq. (5.15) we have,

$$DN = \frac{E}{U\delta_{f}^{2}\pi^{2}d^{2}} \left[\frac{1}{2} Kx_{\max}^{2} \right]$$
 (5.20)

Substituting this result into Eq. (5.14) we have, finally,

$$d = \sqrt[6]{\frac{32K^2 x_{\max}^2}{U\delta_f^2 \pi^2}}$$
(5.21)

This equation will give a wire diameter for a given torsional spring rate and maximum spring extension. Using the wire diameter and Eq. (5.14) families of values of D (coil diameter) and N (no. of coils) can be generated.

Eq. (5.16) defines the volume of a spring capable of taking the yield stress for the material. However this equation does not take into account the repeated nature of the stresses applied to the spring in this application. Failure of a spring under these conditions is known as *fatigue*. Much work has been done in testing springs under these conditions [19] and a typical plot of the number of cycles to failure vs. endurance factor (level of stress divided by tensile stress) for a cyclically loaded spring is shown in Fig. 5.5,

This graph also shows the effect on the fatigue life of springs of shot peening. Shot peening is the process of subjecting the spring to a stream of shot moving at high velocity. The peening action of the shot sets up beneficial biaxial compressive stresses in a thin surface layer which, by preventing surface cracks from propagating into the material, help greatly in extending the spring's fatigue life.

From this graph it can be seen that the maximum cyclical stress must be approximately 1/4 of the tensile strength for the material to produce an infinite life

87

for a shot peened spring. This means that the value for yield stress used in Eq. (5.16) must be reduced by this amount to allow for the fatigue life of the spring. Due to the squared relationship between stress and volume this means that making allowances for infinite fatigue life of a spring means increasing its volume by 16.



5.1.3 Using the MATLAB Toolbox for Component Selection

Three "m-file" routines are available for the component selection stage of the design and they perform the following functions,

1.	SRMIN	Generates a minimum switching rate based on motor power, a
2	CADIS	inertia (implements Eq. (5.1)). Generates values of clutch torques, spring rates and maximum
2.	CADJS	spring extensions given a range of spring system inertias, motor power, minimum switching rate and a minimum and maximum
3.	CADK	output velocity (implements Eq. (5.11)). As CADJS but generates values of spring system inertia given spring rates instead of spring rates given values of spring system
		inertia (implements Eq. (5.12)).

4. SPRINGSZ Determines the wire diameter (d) and families of coil diameter (D) and no. of coils (N) for a helical torsion spring (implements Eq. (5.21))

If the designer has no idea of the required energy switching rate then it can be determined based on the maximum motor power (*P*), the maximum acceptable ripple (ω_{rip}) at a minimum velocity (ω_{min}) for a load inertia (*J_i*) using the routine SRMIN. Then the designer has two choices he can either define an initial search range of spring rates and use routine CADK, or define an initial search range of spring inertias and use routine CADJS. In either case the result will be a family of compatible spring rates, spring system inertias and minimum clutch torques that should all generate systems giving the required performance. When the routines CADK and CADJS are used without left arguments it will force the generation of a plot showing spring rate and spring system inertia as a function of clutch torque. In addition the plot will also show the maximum spring extension and maximum switching rates for the system which are also important design parameters. Using left-hand side arguments extracts the results of the routines into MATLAB matrices for further analysis and suppresses the display of the plot.

An example of using CADJS is given below,

>> cadjs(100,18,100,500,100,[1e-5:1e-5:1e-4]);

where the parameters are, in the order that they are given:

Motor power= 100WMaximum switching rate= 18 HzMinimum output velocity= 100 rad/sMaximum output velocity= 500 rad/sMotor velocity= 100 rad/sSpring system inertia between 10^{-5} and 10^{-4} kg m² in steps of 10^{-5} kgm²

This will generate the plot given in Fig. 5.6,



Figure 5.6 - Example of graphical output from routine CADJS

The designer now has many combinations of the key design components which will all generate similar system performance. In addition, if more than one value is given for any one of the parameters Sr_{min} , P, ω_{min} , ω_{max} or V_m then multiple graphs will be generated showing how the families of spring rates, spring system inertias etc. vary with this parameter. An example of this is given below,

>>cadjs(100,18,100,[300 400 500 600],100,[1e-5:1e-5:1e-4]);

This command will generate separate curves for $\omega_{max} = 300, 400, 500$ and 600, as shown in Fig. 5.7,



Figure 5.7 - Using CADJS with multiple valued input parameters

Values can be read off and displayed on the screen using the mouse (or crosshair); the left-hand mouse button (or spacebar) giving the current cursor position and the right-hand mouse button (or insert key) giving the closest point on the curve to the current cursor position.

From these graphs the designer is able to select suitable clutch and spring parameters. The curves for maximum switching rate and spring system inertia against clutch torque are compared with clutch manufacturers' data to determine a clutch with the appropriate actuation speed and inertia. It should be noted that the spring system inertia is made up of the inertias of the components of the two clutches rigidly connected to the spring, the spring itself and the various mounting components including the connecting rod.

It may be found at this stage that a suitable clutch and/or spring cannot be found that will satisfy the specification, in which case the specification must be modified. Once an acceptable combination of spring rate, spring system inertia and

91

clutch torque has been chosen the designer will be ready to simulate the system's performance in open loop, which will be the topic of the next section.

5.2 Simulation and Efficiency Analysis

The simulation stage of the CAD process is important as it allows the designer to examine the effects of bearing friction, clutch delays and motor dynamics; all these factors having not been considered at the component selection stage. It may also be the case that these simulations show that component parameters must be altered to meet the specification. The same simulation routines can also used when applying closed loop control. Efficiency analysis can also be performed by using multiple simulation runs at different loads, generating efficiency data for a range of output velocities.

5.2.1 Mathematical Model

The mathematical model used in the simulation software is that presented in Section 3.1.2 and the numerical solution implemented is the same as that discussed in Section 3.2. The same controller as that discussed in Section 2.2.1 is used for the simulation.

5.2.2 Analysis of power losses

To perform efficiency and power loss analysis the data from open loop simulations using a range of different load torques is used. These simulations will produce a range of different steady-state output velocities and the results of these simulations can be used to estimate power losses, as a function of output velocity, in key components of the device. A list of these components is shown below,

- 1. Motor bearing
- 2. Spring bearing
- 3. Load bearing
- 4. Input clutch
- 5. Output clutch

The simulation assumes the spring to be a 100% efficient device and so for this analysis this will be the case as well. The equations implemented in the software are those given in Section 3.3.

5.2.3 Using the MATLAB toolbox for Simulation and Efficiency Analysis

The simulation part of the design process uses the following three routines from the MATLAB toolbox,

1.	NEWSUSD	Creates a new simulation model
2.	SIMSUSD	Performs simulation
3.	EFFSUSD	Performs efficiency and power loss analysis

Before the simulation part of the design process can start a simulation model must be created. A simulation model will consist of a MATLAB m-file that defines the parameters used in the subsequent simulation. The name of the model can have up to six letters identifying it and if this is "xxxxxx" then the name for this parameter definition file will be "xxxxxx.m",

This file will contain three distinct sets of parameters: "physical" parameters of the model such as inertias, friction coefficients and the spring rate etc., "controller" parameters that consist of an open loop flag and feedback gains and reference signals etc. (if required), and lastly "simulation" parameters which include parameters such as the minimum and maximum time steps and integration tolerance etc.

The function NEWSUSD will automatically create this file given the model name. Where appropriate, default values will be given for parameters but otherwise the parameters will have blank values which will need to be filled in by the user. An example of using NEWSUSD is given below,

```
>>newsusd('susd');
```

This will create the following file,

```
susd.m
```

and will have to be completed with appropriate values filled in by the user using a suitable text editor.

To simulate the system the function SIMSUSD should be used with the model name as an input parameter. If no left hand side arguments are given then the results of the simulation will be plotted. An example of using SIMSUSD is given below,

>>simsusd('susd');

This will generate a plot similar to that given in Fig. 5.8.



Figure 5.8 - Example of graphical output from simulation routine SIMSUSD

The user can make changes to any parameters at "run-time" without having to edit the parameter file by simply adding the parameter names and values as extra right-hand side arguments to the simulator. This is shown below,

This will perform the simulation with spring rate set to 0.9 Nm and motor inertia set to 0.01 kg m^2 .

To find out the efficiency of the system for a range of different output velocities the function EFFSUSD can be used which will perform multiple runs of the simulator over a given range of output load torques. This will calculate not only the input/output efficiency but also the power lost to the bearings and the power lost to both of the clutches. An example of using EFFSUSD is given below,

>>effsusd('susd',[0:0.05:5]);

This will simulate the model "susd" using equally spaced load torques from 0 to 5 Nm.

If no left hand side arguments are given then a plot will be produced which will give the overall input/output efficiency as a function of output velocity. Included in this plot is the power lost to the three bearings in the system and that lost to the switching of the clutches. An example of this graphical output is given in **Fig. 5.9**.



Figure 5.9 - Example of graphical output from routine EFFSUSD

A typical simulation run can be accomplished in real-time (using a 66 MHz PC) and an efficiency run will take the time for a single simulation run multiplied by the number of load torque values used. All of the software is designed to be used interactively and all plots can be "zoomed" in so, for example, single energy cycles can be examined.
5.3 Feedback Controller Design

In Section 2.2.2 a feedback control law was constructed by starting with an approximate continuous open-loop model. This model was first order and by using a PI feedback loop the analysis suggested that the overall system should be of type 1. This would mean having zero steady-state error to step velocity inputs and constant errors to constant acceleration inputs. This section suggests how to choose the controller parameters, K_p and K_p , and in addition takes into account stability issues raised by the mechanism's inherent switching characteristics. This is achieved by introducing an equivalent discrete model of the system based on the energy transfer of the mechanism.

5.3.1 Discrete Controller Model

Consider the energy change over one energy transfer cycle. Some of the energy packet goes to an increase in the kinetic energy of the load, some gets dissipated through the viscous term (representing the bearings etc.) and the rest to drive the load torque. This is represented in the following equation,

$$\varepsilon_{k} = \frac{1}{2} J_{l} \omega_{k+1}^{2} - \frac{1}{2} J_{l} \omega_{k}^{2} + t_{s} B_{total} \omega_{k}^{2} + t_{s} \tau_{l} \omega_{k}$$
(5.22)

where,

 ε_k = Energy packet for step k ω_k = Average load velocity for step k t_s = Time to complete one energy transfer cycle (1/switching rate)

 B_{total} = Term representing viscous losses τ_{l} = Load torque

Let $x = \omega_k^2$ and so we have,

$$\varepsilon_{k} = \frac{1}{2} J_{l} x_{k+1} - \frac{1}{2} J_{l} x_{k} + t_{s} B_{total} x_{k} + t_{s} \tau_{l} x_{k}^{2}$$
(5.23)

The form of this equation is almost identical to that given in Eq. (2.6) for the continuous model. Eq. (5.23) can be linearised by following a similar procedure as that used for the continuous model. This leads to the following linear model about an operating point, x_0 ,

$$\varepsilon_{k} = \frac{1}{2} J_{l} x_{k+1} - \frac{1}{2} J_{l} x_{k} + \left(t_{s} B_{total} + \frac{t_{s} \tau_{l}}{2 x_{0}^{\frac{1}{2}}} \right) x_{k}$$
(5.24)

So that we can perform control systems analysis it is necessary to take Z transforms of both sides. It should be noted, however, that if the resonant type switching algorithm is used, the switching rate will not be constant over the full range of output velocities and so that taking Z transforms will be an approximation. After taking Z transforms on both sides we have the following equation,

$$E(Z) = \left[\frac{1}{2} J_l(Z-1) + \left(t_s B_{total} + \frac{t_s \tau_l}{2 x_0^{\frac{1}{2}}} \right) \right] X(Z)$$

and re-arranging we have,

$$\frac{X(Z)}{E(Z)} = \frac{1}{\frac{J_{l}}{2}(Z-1) + t_{s}\left(B_{total} + \frac{\tau_{l}}{2x_{0}^{\frac{1}{2}}}\right)}$$

which means the discrete transfer function, G(Z) is given by,

$$G(Z) = \frac{X(Z)}{E(Z)} = \frac{\frac{2}{J_l}}{Z - 1 + \frac{t_s}{J_l} \left(2B_{total} + \frac{\tau_l}{x_0^{\frac{1}{2}}} \right)}$$
(5.25)

Since we are using foward differences it is worth verifying that this transfer function is stable and hence we have, for poles of G(Z) to be outside the unit circle,

This equation states that the dissipative energy cannot be greater than the current kinetic energy of the load plus the energy provided by the energy packet, i.e. Eq. (5.26) implies that,

$$t_{s}B_{total} + \frac{t_{s}\tau_{s}}{2x_{0}^{\frac{1}{2}}} < J_{l}$$
 (5.27)

and multiplying Eq. (5.24) by $2/J_l$ we have,

$$\frac{2\varepsilon_k}{J_l} = x_{k+1} - x_k + \frac{2\left(t_s B_{total} + \frac{t_s \tau_l}{2x_0^{\frac{1}{2}}}\right)}{J_l} x_k$$

and assuming equality for Eq. (5.27) (the point of marginal stability) implies,

$$\frac{2\varepsilon_k}{J_l} = x_{k+1} + x_k \tag{5.28}$$

This difference equation is shown graphically in **Fig. 5.10** (with zero initial conditions). The situation in this figure clearly shows that all the energy provided by the energy packet is dissipated before the next energy cycle can begin. However a system using components which gave such a behaviour would not have been developed at this stage as such a high level of ripple would not be acceptable. Hence we will assume the system is designed correctly and therefore assume the open-loop linearised system (**Eq. (5.25**)) to be unconditionally stable. Note however that this does not necessarily mean that the non-linear model (**Eq. (5.23**)) is unconditionally stable.



Figure 5.10 - Graphical representation of point of marginal stability for the openloop discrete model of the mechanism

Let us now look at the discrete closed-loop model of the system including the controller. This is shown in Fig. 5.11 below,



Figure 5.11- Discrete feedback model

Section 2.2.2 showed that using a PI controller for the continuous model created a type 1 system. Using this controller for the discrete model we can easily derive the Z transform (using an Euler approximation) for the controller as,

$$G_{c}(Z) = \frac{K_{p}(Z-C)}{C(Z-1)}$$
 where, $C = \frac{K_{p}}{K_{p} + K_{i}t_{s}}$ (5.29)

5.3.2 Choosing the Controller Gains

We can combine **Eqs.** (5.25) and (5.29) to find the open-loop transfer function for the system (controller plus mechanism),

$$G_{c}(Z)G(Z) = \frac{K_{p}(Z-C)}{C(Z-1)} \frac{A}{(Z-1+B)}$$
where, $A = \frac{2}{J_{l}}$

$$B = \frac{t_{s}}{J_{l}} \left(2B_{total} + \frac{\tau_{l}}{x_{0}^{\frac{1}{2}}} \right)$$

$$C = \frac{K_{p}}{K_{p} + K_{l}t_{s}}$$
(5.30)

The closed-loop response of the system can be considered by looking at the rootlocus of G_cG to changes in the controller gain K_p . The root locus is dependent on the position of the zero at C, the pole at 1-B and the pole at (1,0). Four pertinent cases, (a) to (d) are shown in **Fig. 5.12** below,



Figure 5.12 - Root locus trajectories for four different locations of the pole and the zero

Case a) has the zero (at C) to the right of the pole (at 1-B) where cases b) to d) have the zero to the left of the pole. The loci never leave the real axis in case a), re-join the real axis to the right of the origin in case b), re-join to the left of the origin in case c), and directly at the origin for case d). In case a) the maximum speed of response is dominated by the zero at Z = C, in case b) increasing the gain will cause overshoot but if this gain is increased further both of the poles can still be positioned on the positive real-axis and so produce no overshoot. In the third case, c), it is not possible to position the poles on the real axis after they have left it. Case d) is the most advantageous case since the point where the trajectories re-join the real axis is at the origin and so the system will have, theoretically, an infinite speed of response with no overshoot. However it must be realised that in practice the speed of response will be finite due to the finite power output of the device.

The object of the gain selection will be to position the open-loop zero (C) as close as possible to the origin such that case c) or d) exists i.e. one pole can be positioned at the origin and the other can be positioned arbitrarily close on the positive real axis. Considering the closed-loop poles we have,

$$G_{cl}(Z) = \frac{G_c(Z)G(Z)}{1 + G_c(Z)G(Z)}$$

which from Eq. (5.30) means,

$$G_{cl}(Z) = \frac{K_p A(Z - C)}{K_p A(Z - C) + C(Z - 1)(Z - 1 + B)}$$

Therefore for the closed-loop poles, we have, after manipulation of the denominator,

$$Z^{2} + Z\left(B - 2 + \frac{K_{p}A}{C}\right) + 1 - B - K_{p}A = 0$$
(5.31)

For one pole at the origin we have,

$$1 - B - K_p A = 0$$

$$\Rightarrow \qquad K_p = \frac{1 - B}{A}$$
(5.32)

and the other on positive the real axis,

$$B - 2 + \frac{K_p A}{C} \le 0 \tag{5.33}$$

Equality of this equation would mean the other root being on the origin. Substituting for A, B and C from Eq. (5.30) would mean,

$$K_{p} = \frac{J_{l}}{2} - t_{s} \left(B_{total} + \frac{\tau_{l}}{2x_{0}^{\frac{1}{2}}} \right)$$
 and $K_{l} \leq \frac{J_{l}}{2t_{s}}$ (5.34)

These equations would generate values for K_p and K_l such that one pole was at the origin with the other was on the positive real axis, generating arbitrarily fast response with no overshoot.

We can also use Eq. (5.31) to assess the stability of the system. Let us consider the situation where Z = -1. This is the only case we need to consider since from Fig. 5.12 it can be seen for positive C and positive (1-B) this is the only route out of the unit circle. In this case we have,

$$1 - \left(B - 2 + \frac{K_p A}{C}\right) + 1 - B - K_p A = 0$$
 (5.35)

and substituting for C we have,

$$K_{p} = \frac{2-B}{A} - \frac{K_{I}t_{s}}{2}$$
(5.36)

This is a straight line with points above the straight line producing values of K_p and K_I that are unstable and points below it that are stable. Fig. 5.13 graphically summarises the ideas put forward in this sub-section and shows, as a function of K_p and K_I , the regions of system stability and instability and the line of system response with no overshoot (one pole at the origin and the other close by on the positive real axis).



Figure 5.13 - Closed loop system as a function of K_p and K_1 .

The feedback gain selection guidelines so far presented assume the designer has knowledge of the relevant system parameters and that they will not change. In practise the parameter variation that the controller has to deal with (in particular the load inertia and load torque) will be quite large and so the designer must base his gain selection calculations on the worst case. This will inevitably result in a compromise between closed-loop performance and robustness to parameter changes. In addition to parameter changes, the designer must also be aware that the model used in the analysis is linearised about an operating point, x_0 and so the closed-loop performance will degrade to some degree as the system moves from this point.

The analysis presented also assumes the control action to be unlimited but obviously this will be finite for the real mechanism. This will lead to a degradation in closed-loop performance due to the effective feedback gain being reduced as the demand power exceeds that can be supplied.

As a further step in the feedback gain selection the designer should now use simulation so that such effects as finite control action, parameter variation and changes in the linearisation point can be assessed. The steps to accomplish simulation using the MATLAB toolbox have already been explained in Section 5.2.3. In that explanation the simulations were performed open-loop but now the designer can explicitly set values for K_p and K_I and perform simulations of the closed-loop system.

5.4 Example Design

This section runs through an example design to show how the design process will work in practice and also acts as a summary of the previous four sections. Let us assume the specification of the device to be that given in **Table 5.1** shown below,

Motor power	500 watts
Motor velocity	100 rad/s (≈960 rpm)
Output velocity range	± 500 rad/s (≈4800 rpm)
Switching rate	> 5 cycles/s

 Table 5.1 - Example specification

The target switching rate for this design is not really known but it is assumed to be a minimum of 5 energy cycles/s but will be designed such that it has the largest value that the design constraints will allow.

The first step in the design process is to find values for the clutch torque and the stiffness and physical size of the spring. To achieve this we will use the routine CADJS which will solve the component design equations (Eqs. 5.9, 5.10, 5.11 and 5.13) and produce families of component values which will meet the design specification. This routine was called with four different values of switching rate (5,10 15 and 20) so that the price paid, in terms of design restrictions, could be assessed as the switching rate was increased. A initial choice of spring system inertia of $1x10^{-4}$ to $1x10^{-3}$ kg m² was chosen based on experience of typical clutches. The MATLAB command line is given below,

>> cadjs(500,[5 10 15 20],0,500,100,[1e-4:1e-4:1e-3]);



The corresponding graphical output is shown in Fig. 5.14,

Figure 5.14 - Families of parameter values for four different switching rates (5,10,15 and 20 energy cycles/s)

Four separate families of parameters are shown, one family for each value of switching rate. The value of maximum switching rate shown in this graph is the expected switching rate at the maximum velocity given in the design specification. Assuming no other criteria need to be applied it is best to choose the minimum value

of clutch torque that gives the full range of switching rates since a smaller clutch will be cheaper and will normally switch faster. From this figure the value of clutch torque is 20 Nm. At this value of clutch torque we can read off the values of the other design parameters. These values are given **Table 5.2**. Also shown in this table is the required energy capacity of the spring which can be trivially calculated from the maximum extension and spring rate.

Switching rate (cycles/s)	Spring system inertia (kg m ²)	Maximum switching rate (cycles/s)	Maximum spring extension (radians)	Spring rate (Nm/rad)	Spring capacity (joules)
5	3.94 x 10 ⁻⁴	7.52	11.5	1.71	113.07
10	2.03 x 10 ⁻⁴	15.07	5.88	3.43	59.29
15	1.32 x 10-4	22.63	3.81	5.36	38.90
20	1.00 x 10 ⁻⁴	30.0	2.86	7.07	28.91

Table 5.2 - Design parameters with clutch torque = 20 Nm

From this data we can see the effect on the design parameters when the switching rate is increased: Both the spring system inertia, and the spring capacity decreases. These trends are quite intuitive since to increase the switching rate the natural frequency of the spring system needs to be increased (implying a reduction in its inertia or an increase in its stiffness) and having an increased switching rate implies a smaller energy packet size (assuming constant power). It can be seen why the design of the spring assembly is very important. Having a larger switching rate will always be advantageous since it means that there will be less ripple in the output velocity. However this is at the cost of much lower spring inertia and means that the design of these components must be such that their inertia must be kept at an absolute minimum. It might mean the design, in terms of a specified switching rate, is unrealisable for just this reason alone. The inverse relationship between switching rate and spring system inertia would help in this respect (a smaller spring capacity implies a smaller spring), however, in many practical systems the spring system inertia will be dominated by the inertia of the clutch components and connecting rod

etc.

Assuming that the designer is confident that the spring system inertia restrictions can be met then he can go forward and simulate the system in open-loop so that the performance specification can be verified. By performing a simulation the designer can see the effects of so far unmodelled effects such as clutch delay, finite motor inertia, bearing losses and sample rate.

For our example we will assume that the worst spring system inertia case can be met and so use a switching rate of 20 energy cycles per second and a target spring system inertia of $1 \times 10^{-4} \text{ kg m}^2$. Another important parameter to select at this stage is the sample rate of the system (the rate at which the controller samples the mechanism). This must be large enough such that enough points are sampled over one energy transfer cycle of the system thus allowing accurate switching of the clutches to be achieved. Table 5.2 shows that the switching rate will increase to 30 energy cycles/s at the maximum output velocity and so choosing a sample rate of 1000 samples/s would give the system at least 30 sampled points per energy cycle. Assuming friction data for bearings of an appropriate size and the clutch switching delay is known a simulation of the device's open-loop response can be done. For the purposes of this example identical data to that used for the 100 watt prototype was used. To produce this simulation the routine SIMSUSD was used and the associated parameter file for this simulation is contained in Appendix J. The open loop simulation of the system in both forward and reverse output velocities using a constant load torque of 0.8 Nm (which always opposed movement) and a load inertia of 0.005 kg m^2 is shown in Fig. 5.15.



Figure 5.15 - Open loop simulation in both forward and reverse directions with a constant load torque of 0.8 Nm and load inertia of 0.005 kg m²



Figure 5.16 - Half second snap shot of open-loop test

As can be seen the maximum and minimum velocities, as defined in the original specification, have been met and at the same time the maximum spring extension never goes significantly beyond the design maximum of 2.86 radians. This latter

point is quite important since it ensures that when the design of the spring is done the maximum stresses that the spring is subjected to is known. The load torque of 0.8 Nm also suggest that significant power levels (at least 400 watts) are being transferred at this velocity. **Fig. 5.16** shows a half-second snap shot of this test close to its steady-state and it can be seen that the predicted maximum switching rate of 30 cycles/sec is a good estimate (the maximum switching rate will occur at the maximum output velocity).

Now that the open-loop performance criteria have been met let us now calculate appropriate feedback gains for the device and test its closed-loop performance. The equations defining the feedback gains giving the fastest speed of response with no overshoot were derived in Section 5.3.2 and are repeated here,

$$K_{p} = \frac{J_{l}}{2} - t_{s} \left(B_{total} + \frac{\tau_{l}}{2x_{0}^{\frac{1}{2}}} \right) \quad \text{and} \quad K_{I} \leq \frac{J_{l}}{2t_{s}}$$
(5.37)

Consider first the maximum velocity (500 rad/s) as the operating point. At this velocity the open-loop test showed a torque of 0.8 Nm gave an almost steady output velocity. The viscous friction coefficient, B_{total} , representing the overall efficiency is not known but can be estimated using the load torque and the input power of the device. The torque generating power losses will be the torque assuming 100% efficiency minus the actual load torque and so we have, at the operating point,

$$B_{total} x_0^{\frac{1}{2}} = \frac{P}{x_0^{\frac{1}{2}}} - \tau_1$$
 (5.38)

Since the input power is 500 watts and the operating velocity is 500 rad/s we have the following estimate for B_{total} ,

$$B_{total} = 4 \times 10^{-4}$$
 Nm s/rad

The value of t_s is 1/30 s since the switching rate at this velocity is of 30 cycles/s. Bringing these values together we have the following values for K_p and K_l ,

$$K_{p} = \frac{J_{l}}{2} - t_{s} \left(B_{total} + \frac{\tau_{l}}{2x_{0}^{\frac{1}{2}}} \right)$$

= 0.0025 - 0.00004 (5.39)
= 0.00246

and

$$K_{I} = \frac{J_{I}}{2t_{s}}$$
(5.40)
= 0.075

Consider now the feedback gains at a very much lower operating point, say, 50 rad/s. At this operating point K_I is unaffected but the value K_p is reduced to,

$$K_p = 0.0025 - 0.00028$$
$$= 0.00222$$

As can be seen the value of K_p is not greatly affected by the operating point but is far more dominated by the value of load inertia. Choosing the lower value of K_p and the value of K_I already calculated we can simulate the system under closed-loop control. A 400 rad/s step input is shown in **Fig. 5.17**.



Figure 5.17 - Closed-loop test with a 400 rad/s step input

As can be seen the reference velocity has been achieved with negligible steady-state error and almost non-existent overshoot. It should also be noted that the peak spring extension (representing the amount of power transferred) is very close to its design maximum (2.86 radians) right up to the point where the output velocity matches the reference velocity. This implies that the system is performing close to its maximum speed of response within the power limitations of the device.

Let us now look at how the system can track ramp reference signals and switch from positive to negative output velocities. **Fig. 5.18** shows a mixed step and ramp test in both positive and negative directions. It can be seen that the tracking error to ramp reference signals is perfectly acceptable given the inherent switching nature of the device.



Figure 5.18 - Closed loop performance with mixed step and ramp reference signals for positive and negative output directions

To complete the closed-loop design process the effects of changes in particular system parameters will be investigated. This will show how robust the controller is to parameter variations and it is essential to know this for any practical controller. The most obvious candidates to test are variations in the load torque and variations in the load inertia since these are likely to occur in an everyday situation. **Fig. 5.19(a)** shows the 400 rad/s step test repeated with different load torques (0.4, 0.8 and 1.0 Nm) with the load inertia kept at 0.005 kg m², and **Fig. 5.19(b)** shows the same test with different load inertias (0.002 0.005 0.01 and 0.02 kg m²) with the load torque fixed at 0.8 Nm.



Figure 5.19(a) - 400 rad/s step test with four different values of load torque (all figures quoted in Nm)



Figure 5.19(b) - 400 rad/s step test with four different values of load inertia (all values quoted in kg m²)

Let us first consider variations in the load torque. We can see that the reference velocity has been met with very little overshoot in all cases. This result fits in with what was expected, since, from Eqs. (5.39) and (5.40), it can be seen that K_p is not greatly dependent on load torque.

For the case of variations in the load inertia it can be seen that decreasing the load inertia from the design point of 0.005 kg m² to 0.002 kg m² caused large amplitude oscillations to be introduced implying the closed-loop system is unstable. Note that the run with load inertia of 0.002 kg m² has been deliberately not plotted after 2 sec. so that the shapes of the other runs could be seen, its oscillatory response from 1 to 2 sec. in fact continues *ad infinitum*. This is to be expected, however, since as can be seen from **Eqs. (5.39)** and (**5.40**) the value of the feedback gains are dominated by the load inertia and decreasing the load inertia will mean the feedback gains will be larger than required and produce closed-loop poles that are quite likely to be outside the unit circle (see **Fig. 5.12**). Increasing the load inertia meant that small amplitude damped oscillations were introduced which is also to be expected, since, having feedback gains lower than required implies closed-loop poles inside the unit circle with a lightly damped oscillatory response.

Obviously if the system is to be used with a load inertia less than that it is designed for (in this case 0.005 kg m²) then the feedback gains must be changed. Using the system with larger inertias (within limits) only causes the system response to slow down and introduces small damped oscillations. It is up to the designer to balance the needs for fast response with those of robustness to parameter variations.

There is also the potential for the application of more advanced control algorithms which could use the dynamics of the spring and load inertia system to effect some form of parameter estimation but this is considered to be beyond the scope of this thesis.

5.5 Conclusions

This chapter has described a method for the design of step-up/step-down mechanisms which will produce a desired system performance. This work covers both component

selection and suggests a feedback scheme to allow output velocity control and gives guidelines for the selection of feedback gains.

The basis of the design approach has been the three different mathematical models that were used. The first model allowed mathematical analysis to be undertaken that generated complete families of components capable of generating the same system performance. The power of this result, from a design perspective, is great since the designer can see, early on, whether the design was feasible for the application and also know exactly how much flexibility there is in the choice of particular components. The second model, which was quite different to the first in that it used a discrete time input/output representation, allowed the design and analysis of the feedback scheme to be achieved and allowed appropriate selection of feedback gains. The third model, based on the internal dynamics of the mechanism, was the most sophisticated and was designed to be the basis for simulations of the mechanism. Simulation of these mechanism is very important since it allows the designer to assess the effects of characteristics such as the dynamics of the motor, clutch delay and bearing losses, not included in the other two models.

A suite of programs, designed to work within the popular control systems design package MATLAB, have been written which will give significant aid to the design process. These programs help the designer in component selection by visually displaying the component families and will also perform the simulation and use the extensive plotting capabilities of MATLAB to display the results. Together these tools, along with manual guides and manufacturer's data, allow the rapid design, to a high degree of accuracy, of these complex machines.

Chapter 6 The Design & Performance of a Prototype Step-Up/Step-Down Mechanism

6.1 Introduction

This chapter describes the design and performance analysis of a step-up/step-down prototype mechanism. This prototype was constructed to see what problems a practical implementation of the device would reveal and to also test the theoretical design methodology discussed in **Chapter 5**. Section 6.2 describes the computer aided design work that was undertaken before the prototype was built. Section 6.3 describes the initial parameter identification tests performed on the device which determine parameter values required by the controller. Section 6.4 performs further parameter identification tests required by the simulation model and compares both the 3-mass model and the controller model to the prototype. Section 6.5 looks at the overall efficiency of the device and also, with the aid of the mathematical model, apportions power losses to the device's various constituent components. Section 6.6 considers the prototype's closed loop response and the last section draws conclusions from the work described in this chapter.

6.2 Computer Aided Design

To start the computer aided design process, as outlined in **Chapter 5**, a specification had to be agreed. This specification is quite straightforward and is quoted in **Table 6.1**,

Motor power	100 watts	
Motor velocity	100 rad/s	
Output velocity range	± 400 rad/s	
Switching rate	> 5 cycles/s	

Table 6.1 - Example specification for step-up/step-down prototype

It was required, if possible, to use 5 Nm clutches manufactured by ZF (Zahnradfabrik Friedrichshafen AG, Germany) since, after consultation with the manufacturers, it was known that these devices could reliably provide high frequency switching and had relatively low armature inertia $(1 \times 10^{-5} \text{ kg m}^2)$. The driving motor used was a servo motor, model G12M4 manufactured by Printed Motors Ltd and was the same motor used for the step-up prototype. This unit could easily provide the required 100 watts of input power. Driving with a constant 21 volts it has a peak power of 105 watts at a speed of approximately 96 rad/s. The complete power vs. velocity characteristic for this input voltage is given in **Fig. 6.1**.



Figure 6.1 - Power vs. velocity characteristic for motor driven at a constant 21 volts It can be seen that the motor power drops off considerably above and below its peak power velocity.

There still remained two important parameters that were required at this stage in the design: the stiffness and inertia of the spring system. To help size these parameters the MATLAB design toolbox routine CADJS was used with three choices of switching rate (10,15 and 20) and an initial spring system inertia search range of $1x10^{-5}$ to $1x10^{-4}$ kg m² (representing one to ten times the clutch armature inertia). The maximum specified output speed of 400 rad/s was also given. The command line used to call this routine is given below,

>>cadjs(100,[10 15 20],0,400,96,[1e-5:1e-5:1e-4]);



This generated the corresponding graphical output shown in Fig. 6.2,

Figure 6.2 - Graphical output from routine CADJS

It can be seen from these graphs that for the torque level of the chosen clutches (5 Nm) three choices of spring rate and spring system inertia are possible. Also shown in these graphs are the maximum spring extension and maximum switching rate. **Table 6.2** gives the values for the three sets of available parameters.

Min. switching rate (cycles/s)	Spring system inertia (kg m ²)	Max. switching rate (cycles/s)	Max. spring extension (rad)	Spring stiffness (Nm/rad)
10	7.3 x 10 ⁻⁵	14.20	5.09	0.98
15	4.8 x 10 ⁻⁵	21.20	3.36	1.48
20	3.5 x 10 ⁻⁵	28.32	2.56	1.95

Table 6.2 - Calculated parameter choices for clutch torque of 5 Nm

This result was encouraging since it suggested that the design was definitely feasible, with the switching rate of the device dependent on how low the inertia of the spring system could be made.

The next stage in the design process was to create drawings of the physical prototype, bearing in mind the requirement that the spring system inertia be as small as possible. For this reason the flanges holding the clutch armatures and the connecting rod were all made of aluminium. The engineering drawing of the prototype is given in **Appendix K**. From this drawing it is quite easy to make an estimate of the spring system inertia by dividing the connecting rod and clutch flanges etc. into a set of cylinders. This estimate came to 8.5e-5 kg m² but was considered to be, if anything, an over estimate due to the approximations made. This suggested it was a feasible design if the set of parameters corresponding to a switching rate of 10 (spring stiffness = 0.98, spring system inertia = 7.3e-5 kg m²) was used and assumed the estimated inertia to be on the large side. Obviously when choosing a spring care must be taken to make sure the spring has sufficient energy storage capacity. From **Table 6.2** we can see that the maximum spring extension is 5.09 radians and so the chosen spring must be able to reliably extend, in either direction, to this amount.

After the drawing was complete the next step was to simulate the open loop performance to check that the specification can still be met when clutch switching times, bearing friction and motor inertia are taken into account. The switching time of the ZF clutches were measured at approximately 4-5 ms (see Section 6.3) and for the bearing friction data, data from the step-up prototype were used but doubled in value due to the larger bearings that would be required for this device. Since the

120

motor used for this prototype is the same as that for the step-up device (the voltage it was operating at is different) the same mathematical model can be used for this simulation. The additional motor inertia $(1.6 \times 10^{-3} \text{ kg m}^2)$ used for the step-up prototype was used as well, and a load torque of 0.15 Nm was applied. This level of load torque was chosen since it was found to be the largest value that allowed the output velocity requirements (±400 rad/s) to be met. The simulation was performed using routine SIMSUSD from the MATLAB design toolbox. The results are shown in **Fig. 6.3** with a load inertia of 0.0023 kgm² for both forward and reverse output directions.



Figure 6.3 - Open-loop simulation of step-up/step-down prototype in both forward and reverse directions with an applied output torque of 0.15 Nm

As can be seen the maximum velocity of 400 rad/s quoted in the original specification has been met and the output load torque of 0.15 Nm represents an acceptable power output of 63 watts at this speed. We can also see that the motor inertia is acceptable since the motor velocity varies from about 80 to 120 rad/s representing a average input power of about 100 watts (see Fig. 6.1).

Let us now consider the maximum spring extension and the maximum switching rate for the device. **Fig. 6.4** shows a 1 second snap shot of the simulation at the maximum load velocity (in the reverse direction), showing motor, spring and load velocities.



Figure 6.4 - One second snap-shot at maximum load velocity

As can be observed the switching rate is approximately 14.5 energy transfer cycles per second and the maximum spring extension is 5 radians. These compare very favourably with the predicted values in the previous component selection analysis (**Table 6.2**), showing about a 2% difference in both maximum switching rate and maximum spring extension.

The next stage in the design is to consider the device's closed-loop performance. This requires determination of the controller parameters K_p and K_l and guidelines for selecting these values were outlined in Section 5.3.2. This selection involved two design equations (Eq. (5.34)) and the implementation of these equations requires an operating velocity to be chosen and values for load torque (τ_l) , total viscous damping

 (B_{total}) , load inertia (J_m) and switching time (t_s) to be selected. By choosing an operating point of 400 rad/s the previously described open-loop simulation can be used and so we have $\tau_l = 0.15$ Nm, $J_m = 0.0023$ kg m² and $t_s = 1/14.5$ (1/switching rate). This only leaves the total viscous damping which can easily be found by considering the total output torque assuming no losses (which at 400 rad/s = 100/400 = 0.25 Nm) and the actual load torque (=0.15 Nm) and so we have,

$$B_{total} = \frac{load \ torque \ assuming \ no \ power \ losses - actual load \ torque}{operating \ velocity}$$
$$= \frac{0.25 - 0.15}{400}$$
$$= 0.00025$$

and so using the design equations we have the following values for the feedback gains,

$$K_p = 0.00112$$
 and $K_I = 0.0161$ (6.1)

Fig. 6.5 shows a simulation of the device under closed loop control following a mixed step and ramp reference trajectory. As can be seen the device shows perfectly acceptable closed-loop performance with the gains calculated. This is perhaps to be expected however since the design parameters used in the simulation are exactly those used to calculate the feedback gains. However it should be realised that the simulation operates away from the operating point used in the feedback design.



Figure 6.5 - Simulation of closed-loop response for a mixed step and ramp reference trajectory

This simulation completed the initial design phase for the device and a prototype based on the drawings in Appendix K was built. A photograph of the completed prototype is shown in Appendix L.

6.3 Initial Test Results

When the prototype was being designed the spring design guidelines, as described in Section 5.1.2, were not mature and sizing the spring was a case of guesswork and engineering judgement. This unfortunately led to the first springs being tried either breaking or twisting beyond their elastic limit well before the design power of 100 watts was applied. In hindsight this was hardly surprising since the springs had a volume (and hence an energy capacity) well below that recommended by the guidelines. For infinite fatigue life the design equations suggest a spring volume of $1.58 \times 10^{-4} \text{ m}^3$ but unfortunately the prototype was already built before this information was available and the prototype was such that the spring was limited to

have a coil diameter of roughly 3 cm and a maximum length of 9 cm. This restriction meant that the volume and stiffness criteria could not be met with commercially available springs. To significantly increase the volume under these conditions meant that the stiffness had to be increased as well. The spring chosen had a stiffness of 1.32 Nm/rad and a volume of $3.97 \times 10^{-5} \text{ m}^3$ which was still 1/4 of the volume recommended by the design guidelines. However this spring was such that the maximum stress applied would only be half its tensile stress and this should give a fatigue life of a few thousand cycles (see Fig. 5.5). The important design parameters and dimensions for the spring used are shown in Table 6.3,

Coil diameter (mm)	Wire diameter (mm)	No. of coils	Stiffness (Nm/rad)	Volume (m ³)
27.81	3.43	12.3	1.32	3.97 x 10 ⁻⁵

Table 6.3 - Design parameters for spring used in step-up/step-down prototype The device was found to operate at the design power even though the spring was found not to have a significant fatigue life (the prediction of a few thousand cycles was about right).

Before open-loop tests could be performed certain parameters needed to be identified for use by the controller. These parameters are,

- 1. Value of $\sqrt{\frac{K}{J}}$ (square root of ratio of spring stiffness to spring system inertia)
- 2. Clutch delay

A good approximation of $\sqrt{k_J}$ can readily be found from the natural oscillation of the spring since the damping will be very low. This oscillation was measured at 21 Hz which gave,

$$\sqrt{K/J} \approx 2\pi \times 21 = 132 \text{ rad / s}$$

The clutch delay could not be accurately deduced from the manufacturer's data since the drivers used were different to those usually supplied by the manufacturers. These drivers were designed to minimise the engagement and disengagement time of these clutches and were built especially for this project. When engaging the clutch, the clutch driver would give a short initial burst of +100 volts to overcome the clutches inductance and build up the magnetic field, and after this initial burst the voltage would reduce to the normally rated voltage of 24 volts. When disengaging the clutch the driver would apply a short burst of -100 volts to quickly suppress the magnetic field before returning to 0 volts. The output voltage signals from these drivers for a single engagement/disengagement cycle is given in **Fig. 6.6**.



Figure 6.6 - Output voltage waveform for specially designed clutch drivers

The widths of the high voltage bursts could be finely adjusted and in this way the clutch and driver combination could be tuned to minimise the engagement/disengagement times.

To measure the clutch delay and also tune its response the clutch was removed from the prototype mechanism and the two sides of the clutch were electrically insulated from one another. Small plastic bolts were used for this purpose and allowed the correct air gap to be maintained between the plates. An electrical circuit was created by attaching wires to either sides of the clutch and, with a resistor in series, connecting the other ends of the wires to a voltage supply. This meant that when the clutch faces meet an electrical connection is made (the plates are made of metal) and a current will flow. This arrangement is shown in **Fig. 6.7**



Figure 6.7 - Experimental set-up to measure clutch delay

The voltage across the resistor was monitored by a PC which also drove the clutch through the clutch driver. When the voltage across the resistor reached a threshold value the clutch faces were assumed to be making contact and when the voltage was close to zero the contact was assumed to be broken. In addition the current through the clutch was monitored by using a resistor in series with the supply to the clutch. **Fig. 6.8** shows the clutch driver signal, clutch contact signal and clutch current for a single engagement/disengagement cycle.



Figure 6.8 - Clutch signal and clutch contact signal for a single engagement/disengagement cycle

As can be seen the engagement delay is approximately 4 ms and the disengagement delay 4-5 ms. It should be realised however that the engagement time only shows the delay until clutch plate contact is made, it does not necessarily imply full torque is being applied. However it can be seen that the clutch current, and by implication the attractive force on the plates, reaches its steady-state value before the plates actually make contact.

It should be realised that this value of clutch delay (4-5 ms) is considerably faster than that quoted in the manufacturer's data (30 ms engagement time and a 15 ms disengagement time using a standard 24v supply) and this type of driver is crucial if these clutches are to be used successfully in these drives.

This experiment completed the identification of the parameters required by the controller to operate the device in open-loop. The list of parameters required by the controller and their values are summarised below,

128

spring stiffness	= 1.32 Nm/rad
$\sqrt{K_{J}}$	= 132 rad/s
clutch delay	=4 ms

The results of the initial 100 watt tests, with no load torque applied, for both forward and reverse output directions are shown in **Fig. 6.9**.



Figure 6.9 - Initial 100 watt tests for forward and reverse load directions

These results are encouraging since they show the concept can be made to work in practice. The shape of the load trajectory is almost identical to that shown in the original simulation (see **Fig. 6.3**), the motor takes longer to accelerate but this is because its input signal was ramped upwards (to limit the current) whereas in the simulation it was a step. It will be noticed that the original specified velocity of ± 400 rad/s has been easily met on the negative direction and almost met in the positive direction (even though it eventually makes -400 rad/s when it reaches its steady-state). It is apparent that less power is being transmitted in the positive direction (the acceleration is less) but this is not surprising since the average spring velocity and hence the viscous losses will be greater in this case (this is also true but to a lesser extent in original simulation). These results were performed without any load torque

but the original simulation of the device had a load torque of 0.15 Nm (representing an output power of 63 watts) and still managed to attain these output velocities. This means there is a discrepancy with the original mathematical model and the actual prototype. The next section improves the model by performing further parameter identification tests and pinpoints the deficiency in the original model.

6.4 Mathematical Model of Prototype

This section attempts to improve the model for the prototype so that a more detailed analysis of its performance can be undertaken. Section 3.1 developed a three mass model for this device and this model was used when designing the prototype (see Fig. 6.3). This initial model made certain assumptions concerning the parameters of the device (in particular the exact spring system inertia and the bearing friction). This section will describe experiments performed to identify the uncertain model parameters and to help verify whether the original model structure is adequate at modelling these devices. The first sub-section deals with the bearings, finds accurate friction parameters for them and also verifies that the bearing alignment is correct. The second sub-section deals with the spring and performs experiments to obtain the stiffness, hysteresis curve and spring system inertia (which includes all components rigidly attached). The third sub-section compares the model to the prototype and raises issues concerning the modelling of the spring.

The model of the motor used with this mechanism has already been verified in **Section 4.3.1** since it was the motor used with the step-up mechanism.

6.4.1 Bearing Friction Coefficients

The bearings used in this mechanism were double row angular contact roller bearings. These bearings were configured as opposed pairs and meant that axial as well as shear loads could be withstood. This is important since the spring and the clutches apply axial loads on the bearings. With reference to the engineering drawing

130

(see **Appendix K**) it will be noticed that one pair of bearings was used on the motor side of the device, two pairs in the spring section and one pair on the load side.

The same parameter identification experiment used for the step-up prototype was applied i.e. the load inertia was accelerated to a high velocity and allowed to coast to a halt (the output clutch was disengaged). The output encoder measured the load displacement and this data was differentiated to obtain velocity and acceleration data. The data was heavily averaged so that the discretisation and sampling effects could be minimised. Twenty velocity vs. acceleration points were obtained and since the load inertia was known (0.0023 kg m²) the frictional torque could be found. The results for the output bearing is shown in **Fig. 6.10**.



Figure 6.10 - Velocity vs. friction data for the output bearing

This is an acceptable linear fit and the viscous and coulomb coefficients are easily obtained from the gradient and the intercept with the friction axis and are quoted below,

The next experiment attempted to see if there was a problem with the bearing alignment which would show up as increased bearing friction when any of the clutches were engaged. Two tests were performed, the first to test alignment between the output and spring bearings and the second the alignment between the input and the spring bearings. The first test innvolved removing the spring and repeating the previous experiment with the input clutch disengaged and the output clutch engaged, the second test involved removing the spring, replacing the motor with the load inertia and repeating the experiment with the input clutch engaged and the output clutch disengaged. To obtain the frictional torque from the acceleration data an estimate of the inertia of the spring components (i.e. the clutch rotor inertia and connecting rod) had to be made. The clutch rotor inertia was available from the manufacturer's data and the connecting rod could be approximated by a cylinder. The result obtained was about 6 x 10⁻⁵ kg m² but this was insignificant compared to the load inertia (0.0023 kg m²) and so any inaccuracy in its value would make very little difference to the final result. The results of these experiments are shown in Fig. **6.11(a)** and **Fig. 6.11(b)**.



Figure 6.11(a) - Velocity vs. friction data for the spring and output bearings


Figure 6.11(b) - Velocity vs. friction data for the spring and input bearings

The corresponding friction coefficients are,

Spring and output bearing viscous coefficient = 16.1 x 10⁻⁵ Nm s/rad coulomb coefficient = 0.0078 Nm

Spring and input bearing viscous coefficient = 14.05 x 10⁻⁵ Nm s/rad

coulomb coefficient = 0.009 Nm

Assuming the bearings to be perfectly aligned these results should be about three times that found for the output bearing alone since now three sets of bearings will be contributing friction. Both the viscous coefficients shown above are below one third and the coulomb coefficients are only slightly over. This result is encouraging since it suggests that no significant friction has appeared when either of the clutches are engaged, showing that there is no problem with bearing alignment.

6.4.2 Spring stiffness and spring system inertia

The method used to measure the spring system inertia was to get an accurate value of K/J (ratio of stiffness to inertia) from a free vibration test, to independently measure the spring stiffness and use these two results to find the spring system inertia. This is similar to the approach used for the step-up prototype, that approach was slightly different in that the spring stiffness was not independently measured but two different inertias were used in the free vibration response tests.

It was important to measure the spring stiffness when it is installed in the device since the clips holding either side of it will reduce the number of active coils and so the theoretical stiffness will be slightly inaccurate. To measure the stiffness a torque transducer was attached to the output shaft, the output clutch engaged and the output shaft rotated whilst measuring the spring displacement and torque. Results for both positive and negative output directions are shown in Fig. 6.12(a) and Fig. 6.12(b). Along with the raw data, lines of best fit have been added.



Figure 6.12(a) - Displacement vs. torque data (positive direction) for spring used in the step-up/step-down prototype



Figure 6.12(b) - Displacement vs. torque data (negative direction) for spring used in the step-up/step-down prototype

The spring stiffness will be the gradient of the line of best fit which is 1.43 Nm/rad for the positive direction and 1.37 Nm/rad in the negative direction. The average of these two results, which is 1.40 Nm/rad, will be used as the representative stiffness. As is apparent this value is higher than that calculated (see **Table 6.3**) due to the effect of the clips reducing the number of active coils.

With an accurate measure of stiffness now found an accurate measure of K/J needs to be obtained. The previous section of this chapter found approximate value for $\sqrt{K/J}$ by simply finding the natural frequency of the spring system. The experiment considered here will improve this estimate by fitting a combined coulomb and viscous model to the free vibration response and so take into account damping effects. Fig. 6.13 shows the free vibration response with the model fitted for displacement, velocity and acceleration data.



Figure 6.13 - Measured free vibration response of the spring with the fitted model The fitted model had a value of *K/J* equal to 18038 which, using the previously measured spring stiffness, implies a spring system inertia of 7.76 x 10⁻⁵ kg m². This value is very close to the original design specification of 7.3 x 10⁻⁵ kg m² and the value of $\sqrt{K/J}$ is 134.3 rad/s which is less than 2% different from the value found from considering the natural frequency.

The next experiment attempted to look at the hysteresis curve for the spring. The mathematical model developed in **Section 3.1** assumed the spring to be 100% efficient which, in practice, is not the case and this could be where the model has some deficiencies. **Fig. 6.14** shows three hysteresis curves for the spring, the first extends the spring to 3 radians, the second to 3.3 and the third to 4 radians. This last value is just about the largest extension that the spring will be subjected to in operation since it required a torque of almost 5 Nm (the maximum torque of the clutches).



Figure 6.14 - Three hysteresis curves for the spring

As can be seen the spring is obviously not 100% efficient and the efficiency is seriously degraded as the maximum extension is reached. The ratio of area under the return path to the area under the forward path will show the energy efficiency of the spring. This can be found by numerical integration and the results for the three curves are shown below,

Maximum extension (rad)	Energy efficiency (%)
3	91
3.3	89
4	84

Table 6.4 - Energy efficiency for the different hysteresis curves

Steidel in his book on mechanical vibrations [21] investigates hysteric damping and shows that classical hysteretic damping will have an energy loss proportional to extension squared. He also goes on to show that this is also the case for viscous damping. Where they differ is that viscous losses will always increase with frequency where as hysteretic damping is invariably independent of frequency or only decreases

slightly with frequency. Since the free vibration test has almost constant frequency there will be no significant difference between viscous and hysteritic damping. A method of determining the damping that a system is subjected to is to look at how the natural log of the peak amplitude varies with the cycle number of free vibration. If the damping is viscous or hysteretic (energy dissipation proportional to position squared) then the plot will be linear, if the damping is coulombic then the curve will be a downward concave shape. The plot of the log of peak amplitude vs. cycle number for **Fig. 6.13** is shown in **Fig. 6.15**.



Figure 6.15 - Log of peak amplitude vs. cycle no for free vibration test

As can be seen, at higher extensions (and velocities) the response is dominated by viscous or hysteretic damping but at lower extensions (and velocities) the damping becomes coulombic in nature. This is what is intuitively expected since coulomb damping becomes dominant at lower velocities and viscous damping at higher velocities. This result is very important since it shows that hysteretic damping can be modelled by simply increasing the viscous friction term. This is what was done for the step-up mechanism (see Section 4.4) and this result gives justification for doing

SO.

6.4.3 Comparison of Model and Real System

The previous section found accurate values for the spring system inertia and spring stiffness but highlighted a problem with the previous model, namely, that inefficiencies in the spring were not taken into account. However, it was proposed that theoretically this can be modelled by simply increasing the viscous friction coefficient used for the spring. To find a viscous coefficient that takes into account these extra losses the first cycle of the free vibration test (see **Fig. 6.13**) will be considered and a viscous only fit will be applied (coulombic forces will be negligible at these higher velocities). This can be justified since when the device is operating the spring will only ever move through one free oscillation before being loaded up with more energy. The maximum extension of the spring will be typically 3 radians when transferring 50 watts and close to 4 radians when transferring 100 watts. **Fig. 6.16** shows a viscous only fit for the first cycle from the free vibration test of **Fig. 6.13**.



Figure 6.16 - Viscous fit for the first cycle of the free vibration test

This gave a value of B/J equal to 8.71 which, using the previously calculated value for J (7.76 x 10⁻⁵ kg m²), gives the following value for the spring viscous friction coefficient,

 $B_s = 67.6 \times 10^{-5} \text{ Nm s/rad}$

The extension from this cycle varied from about 3 to 2.5 radians but running the device at 100 watts would mean having spring extensions of nearly 4 radians and from the hysteretic losses, given in **Table 6.4**, it will be observed that losses are greater at these higher extensions. Running the device at 50 watts input power gives a maximum extension of 3.2 radians and so the viscous coefficient just found would be more applicable in this case. **Fig. 6.17(a)** compares the open-loop response of the device and the equivalent simulation using the previously calculated value for B_s , and **Fig. 6.17(b)** shows a small snap-shot of this test showing the spring extension as well.



Figure 6.17(a) - Comparison of model and real system with input power of 50 watts power and Bs = 67.6 x 10^{-5} kg m²



Figure 6.17(b) - Snap-shot of above showing spring extension as well

This result shows the model to be very accurate, both when considering the device's overall performance and the device's internal dynamics, that is, the spring extension and variations in motor velocity etc. over a single energy cycle. This shows the structure of the original 3-mass model to be perfectly acceptable but care must taken in choosing the correct level of spring viscous friction so that inefficiencies in the spring are taken into account. This later point is important since the value of viscous friction needs to be multiplied by six to account for losses in the spring.

A good fit between the model and real system running at 100 watts can be made if the spring viscous term is increased still further to a value of 107 Nm s/rad and the spring stiffness reduced from 1.4 to 1.2 Nm/rad (taking into account the degradation in stiffness found at higher extensions). This result is shown in **Figs. 6.18 (a)** and **(b)**,

141



Figure 6.18(a) - Comparison of model and real system with input power of 100 watts power and Bs = $107 \times 10^{-5} \text{ kg m}^2$



Figure 6.18(b) - Snap-shot of above showing spring extension as well

The further increase in the viscous coefficient is caused here by the increase in the spring's inefficiency as the spring extension approaches 4 radians. Table 6.4 shows that the proportionate spring losses almost double when the spring extension is

increased from 3.3 radians (that found in the 50 watt test) to 4 radians (that found the 100 watt test) so it is not surprising this would require a large increase in the viscous term when the mechanism operates at this increased power.

As a conclusion to this section we will now consider the model used to design the feedback controller. This model was defined as Eq. (2.5) and is repeated here,

$$P = \frac{J_{l}}{2}\dot{x} + B_{total}x + \tau_{l}x^{\frac{1}{2}}$$
(6.2)

When no additional load is put on the device the load torque will not be zero but will represent the total coulombic losses in the mechanism and so we have,

$$P = \frac{J_{l}}{2} \dot{x} + B_{total} x + C_{total} x^{\frac{1}{2}}$$
(6.3)

The input power to the device (P) is known, as is the load inertia (J) and, using a trial and error approach, the values for B_{total} and C_{total} were found that fitted Eq. (6.3) for the 50 and 100 watt open loop tests. The following coefficients were found to give a good fit,

50 watt test

$$B_{total} = 2.0 \times 10^{-4} \text{ Nm s/rad}$$

$$C_{total} = 0.046 \text{ Nm}$$

100 watt test

$$B_{total} = 1.72 \times 10^{-4} \text{ Nm s/rad}$$

$$C_{total} = 0.12 \text{ Nm}$$

The comparison between the model and the real tests are shown in Fig. 6.19(a) (50 watt test) and Fig. 6.19(b) (100 watt test),



Figure 6.19(a) - Controller design model and real system for an input power of 50 watts



Figure 6.19(b) - Controller design model and real system for input power of 100 watts

This result is again encouraging since it shows the structure of the controller model to be perfectly adequate at modelling the device's input/output response. However, as

can be seen, the coulomb terms are much greater than those expected from considering the bearings alone, and the viscous term is much less than that used for the spring in the previous 3-mass model. This is to be expected however since a viscous loss in the spring part of the 3-mass model will show it self as a power loss proportional to velocity in the input/output model. Since x in Eq. (6.3) is the velocity squared it is the coulomb coefficient that will produce power losses proportional to velocity, whereas the viscous coefficient produces losses proportional to velocity squared.

The quality of this fit suggests that the structure of the controller model is acceptable but the total losses need to be modelled by both a coulomb and viscous loss term. This means that the structure of the equation can remain unchanged but the load torque term should also include the coulomb loss, and hence the revised controller equation is,

$$P = \frac{J_{l}}{2} \dot{x} + B_{total} x + (\tau_{l} + C_{total}) x^{\frac{1}{2}}$$
(6.4)

6.5 Efficiency Analysis

This section describes the efficiency tests performed on the device. These are almost identical to those performed on the step-up prototype as described in **Section 4.4**. A torque transducer was attached to the output shaft of the drive and a variable friction device was attached to the output shaft of the torque transducer which allowed different load torques to be applied. The device was driven open loop and by adjusting the load torque different steady-state output speeds would result. Since both the output torque and velocity were measured the steady-state output power could easily be found.

Even though a good mathematical model of the motor is available the input power vs. speed characteristic was still measured so that losses in the torque transducer and its attachments could be taken into account. This was done by

145

attaching the torque transducer directly to the output shaft of the motor and obtaining steady-state torque and velocity measurements. As it turned out the difference between the powers predicted by the model and that found by measurement was only about 1 to 3 %. The input and output powers were found by integrating the torque \times velocity measurements with respect to time.

Twenty-seven separate open-loop tests were performed over a 60 to 430 rad/s output velocity range and the percentage efficiency (100 x output power/input power) results are shown in **Fig. 6.20**



Figure 6.20 - Percentage efficiency for step-up/step-down prototype operating with an input power = 100 watts

As can be seen the percentage efficiency is between 70 and 80% for a speed range of about 60 to 260 rad/s representing a effective ratio range of 0.6:1 to 1:2.6. Above 260 rad/s the efficiency drops off until, at the maximum design speed of 400 rad/s, it is about 40%. However it must be stressed that these efficiency figures are obtained with the load velocity moving in the opposite direction to the motor (the slightly more efficient output direction) and do not include the power consumption of the

electro-magnetic clutches. These clutches consume 10 watts each when on continuously and a reasonable approximation is that they are each on for half the time suggesting a total additional loss of 10 watts. This efficiency curve may be compared with the equivalent curve for a typical commercial variator [22] as shown in Fig. 6.21,



Figure 6.21 - Typical variator efficiency

The next stage in the efficiency analysis is to apportion the power losses to the various components to see whether improvements can be made to the design. The previous section of this chapter found accurate physical parameters for this device and went on to show that it was an acceptable model for both the internal and external behaviour of the device. Using these parameters the power losses can be apportioned as outlined in **Section 3.3**. The calculated power losses, attributed to the input and output bearings, the total losses of the spring (both friction losses to the bearing and hysteretic losses in the spring) and the losses to the input and output clutches are shown in **Fig. 6.22**.



Figure 6.22 - Losses attributed to the individual components of the step-up/stepdown prototype

As can be seen the spring losses are by far the greatest which is not surprising considering the high level of spring viscous friction required to obtain a good fit between the model and the real system. However it should be realised that the spring is somewhat undersized (the volume is half what it should be) for the power throughput that the device is handling. The hysteresis curves for the spring shown in **Fig. 6.14** suggest that using a larger spring would make the energy efficiency greater. **Table 6.4** suggests at least 90% should be achievable which would be almost twice as efficient as the current spring and having a more efficient spring would significantly reduce the losses at higher velocities.

Another important feature of **Fig. 6.22** is the relatively small power losses attributed to the clutches. This is undoubtedly due to the switching algorithm which time the clutch engagements so that velocity differences across the faces are minimised. This is important since low power loss implies a low wear rate of the clutch faces. This is important when assessing the long term reliability and maintenance costs of these devices.

It will also be noticed that below about 100 rad/s the input clutch power loss increases (causing a decrease in the overall efficiency). Further investigation showed this to be a problem with the switching algorithm when the load is rotating more slowly than the motor (the average motor velocity was 100 rad/s). In this situation the spring will not have enough energy to attain the motor velocity and the controller realises this and so switches on the clutch immediately thus causing energy losses since the clutch face velocities are unmatched. However what it should do is one of two things; i) wait until the spring attains its maximum velocity (i.e. closest to the motor velocity), or ii), leave residual energy in the spring so that it can always attain the motor velocity. The latter solution would be preferable since it would eliminate clutch slipping at these lower load velocities.

6.6 Closed Loop Results

Fig. 6.24(a) shows the result of a closed-loop test using the feedback gain values, $K_p = 0.00112$ and $K_1 = 0.0161$ which were calculated as part of the design procedure. Fig. 6.24(b) shows the error between the reference and the load velocity for the same test.



Figure 6.23(a) - Closed loop tests for the step-up/step-down prototype



Figure 6.24(b) - Error between reference and load velocities for closed loop test

As can be seen the velocity tracking error, though not perfect lies within 8.5% of the maximum velocity (this is after it reaches the reference velocity which is approximately 2 seconds into the test). This is acceptable for many applications and

can be achieved for both forward and reverse reference directions. In fact Fig. **6.24(b)** shows that the error is often much lower than that (under 5% or <10 rad/s) for a great part of the test.

It will also be seen that when the load is decelerating (2.5 < t < 4 s) the speed of the motor increases and the additional kinetic energy which this represents gives the load a boost in power when accelerating in the opposite direction (4 < t < 5.5 s). This shows one of the important features of the device: its bi-directional power transfer capabilities. This means it can as easily transfer energy from the motor to the load as transfer energy from the load to the motor. This is important since it means energy extracted from the load is not wasted, as would be the case for a some kind of brake or a conventional direct drive, but is stored as kinetic energy in the motor and can be immediately used again to accelerate the load if needed. If a single motor were used to drive multiple loads it could transfer energy from one decelerating load to another accelerating one.

6.7 Conclusion

The construction of the prototype has showed that the step-up/step-down mechanism can be designed to work in practice and shows remarkable agreement with the original design model.

The prototype also highlighted the inefficiencies in the spring which turned out to be the dominant power loss and was unaccounted for in the original design. However it was shown that, by simply increasing the spring viscous friction coefficient, this extra power loss could be adequately modelled, as verified by the subsequent quality of fit between the model and the prototype. In fact the structure of both the three-mass model (used in simulations) and the controller model (used for controller design) were shown to be perfectly acceptable at modelling the behaviour of these devices .

151

It was unfortunate that the largest spring that could be accommodated within the device was still only half the capacity of that required by the design guidelines (due to physical constraints built in before the spring design guidelines were known). This meant that the fatigue life was severely compromised and its energy efficiency was found to degrade as the spring's maximum extension was reached. Even with this problem the mechanism's efficiency was still between 70 and 80% for an output velocity range of 60 to 200 rad/s and the specified velocity range of ± 400 rad/s was still attained, all with an input velocity of 100 rad/s. This velocity range extends beyond the velocity range of the motor and also generates faster accelerations than could be achieved by using the motor with a fixed gear ratio. Analysis of the power losses attributed to the device's constituent components highlighted the dominant losses found in the spring and showed that the clutches contributed relatively little (for most velocities this was under 5 watts). This latter point is encouraging since it shows the effectiveness of the switching algorithm and, since low power loss implies low clutch plate ware, this will have important implications when considering the reliability and maintenance costs of these devices.

Lastly the closed-loop performance of the device was assessed. It showed that the closed-loop performance was very close to that expected from the simulations and gave acceptable performance in both forward and reverse directions (under 5% for the majority of the test, peaking to 8.5%). This test also showed the device was capable of bi-directional energy transfer, i.e. energy could be transferred from the load to the motor as well as from the motor to the load.

152

Chapter 7 Conclusion

This thesis has shown that mechanical switched mode drives can be made to work in practice and potentially offer benefits to system performance that cannot be easily obtained with current variable drive technology. Two design variations have been presented: the step-up design, which allows an increase in velocity from input to output, and the step-up/step-down design, which allows both an increase and a decrease in velocity from input to output. The latter design has, in addition, the capabilities of forward and reverse output directions as well as forward and reverse energy directions and so has greater suitability for practical applications.

The operation of these mechanisms can be thought of analogous to electrical switched-mode power supplies (SMPS) and it was obvious that the control strategies used in SMPSs would not work well for these devices and that new control schemes, that consider the problems peculiar to the mechanical devices, were required.

To help in the understanding of the internal dynamics of these devices, separate three-mass models were developed for each device. These models were designed to produce accurate simulations that took account of both the input/output and internal behaviour. These models could predict the onset of clutch or brake slip and model the engagement and disengagement delays associated with electromechanical clutch/brakes. Both of these aspects of clutch/brake behaviour were of crucial importance to the performance of these mechanisms.

The construction of the step-up prototype showed the device worked in practice and, after performing parameter identification experiments, remarkable agreement between the three-mass model and the prototype was apparent. However a deficiency in the model was pin-pointed in that losses in the spring were inadequately modelled. These became significant when the spring was being stressed close to its elastic limit. The model was then used to assess the amount of energy loss attributable to each component. This information could then be used for further system development. The construction of this prototype also highlighted the need for some form of design aids due to the complex relationship between system parameters and system performance.

The next issue to be addressed was how these devices should be designed so that a particular design specification was fulfilled. This work concentrated on the step-up/step-down design due to its perceived greater suitability for practical applications. A design methodology was presented and, crucial to this, was the development of a new model that can predict the performance of these devices in terms of physically meaningful performance measures. The design equations allow the construction of complete families of compatible design parameters that create devices with identical performance characteristics. A set of user friendly tools, working within the popular MATLAB computer package, have been developed to aid in the design process and the use of these have been outlined.

The final chapter dealt with the construction and analysis of a step-up/stepdown prototype. This prototype was developed using many of the design tools described in **Chapter 5** so it would not only show if the drive concept would work in practice it would also show how well the design tools could predict the device's performance. The prototype showed that the concept would work in practice and it's predicted bi-directional output velocity and bi-directional energy transfer capabilities were realisable. These latter two features were shown succinctly when the device was used to track an output waveform that had both step and ramp trajectories in both positive and negative directions. When the load decelerated the motor was seen to accelerate as the energy from the load was transferred to the motor. Where the device's performance differed from that predicted was in its energy efficiency at high output velocities. Analysis showed this to be the result of an undersized spring. In fact, if the design guidelines were adhered to then this mistake would not have occurred. This point shows the importance of the design tools since it is very easy to get the design of these devices wrong with subsequent inefficient operation. As with the step-up mechanism the three-mass model was fitted to this prototype and showed excellent agreement.

The performance of the step-up/step-down prototype indicates that such drives have considerable potential for applications where rapid acceleration is required. The velocity range is greater than a direct drive motor and the acceleration is also better than fixed gear systems. The efficiency is comparable to, but not yet as good as, available variators but has the additional advantages of allowing very rapid ratio change, a bi-directional output capability and even a zero ratio is available (by disengaging the output clutch). Closed-loop velocity control is an inherent feature of the device requiring no extra electronics and instrumentation which is not the case for conventional variable ratio drives. These characteristics make it particularly useful for applications where large accelerations coupled with a large range of output velocities are required. Its bi-directional energy capabilities is a particularly useful feature if system energy efficiency is important.

7.1 Further work

One important analysis task, not so far carried out on the prototypes, was an assessment of their reliability. Of particular concern is the reliability of the electromagnetic clutches. Clutch performance could easily degrade with time and it is not inconceivable that the return springs could fail or soften, thus affecting their engagement/disengagement times.

Of crucial importance to the overall performance of step-up/step-down devices is the armature inertia to torque ratio of the clutches. Since the armatures are directly connected to the spring, and will often contribute a significant proportion of the spring system inertia, it is critical to get the clutch armature inertia as low as possible. The clutches used in the prototype are designed to allow slipping and forcibly equalise the velocity across both plates. However, it has been shown that the switching algorithm used with the prototype has almost eliminated the need for

155

slipping since the face velocities are almost always identical when the clutches are actuated. This would allow clutches with serrations or some form of teeth to be considered [23,24]. These devices would produce considerably higher levels of torque for little gain in inertia and so allow better performance drives to be realised.

Looking a little further into the future there is much promising research in the development of novel clutch designs. Of particular note are piezo-electric clutches [28,29,30] and electro-rheological clutches [31,32,33]. Both of these technologies offer a lot of potential for use in switched-mode drives due to their low inertia, fast actuation capabilities and quite operation.

The PI feedback controller described in the design of these devices is not the only form of controller that could be used with these mechanisms. In fact it is shown that the controller performance is sensitive to changes in load inertia, and that these changes are even capable of causing instability. A better controller to use would be some form of robust controller such as an adaptive of self-tuning controller [25] that attempts to identify changes in system performance and modify the controller accordingly. The pulsed nature of the drive would provide an excellent signal to excite the system dynamics and so allow their identification. Another example of a robust controller [26,27]. These controllers are simple to implement and are generally very robust to parameter changes but since they are switching controllers they have the undesirable feature of adding noise to a system. Since switched-mode mechanisms are inherently switching systems a variable structure controller would be excited to this application.

Appendix A - Calculating the timing of the clutch for the step-up mechanism

This appendix describes how the controller for the step-up mechanism calculates the actuation timing of the clutch so that the velocity differences across the faces of the clutch are minimised. This timing takes the form of a delay between the brake release and the clutch actuation to allow the spring to accelerate to the velocity of the load. The correct delay time can be calculated if the resonant frequency of the spring (and all rigidly attached fixings) and the motor velocity (assumed fixed during this period) are known. Neglecting bearing friction the following differential equaion will define the spring's motion.

$$K(\theta_m - \theta_s) = J_s \dot{\theta}_s \tag{A.1}$$

where

K = Spring rate θ_m = Motor position θ_s = Spring position J_s = Spring inertia

differentiating with respect to time we have,

$$J_s \ddot{\omega}_s + K \omega_s = K \omega_m \tag{A.2}$$

where

 ω_m = Motor angular velocity (assumed fixed) ω_s = Spring angular velocity

leading to the solution,

$$\omega_{s}(t) = C_{1} \cos(\sqrt{\frac{K}{J_{s}}t}) + C_{2} \sin(\sqrt{\frac{K}{J_{s}}t}) + \omega_{m}$$
(A.3)

and

$$\dot{\omega}_{s}(t) = \sqrt{\frac{K}{J_{s}}} C_{2} \cos(\sqrt{\frac{K}{J_{s}}}t) - \sqrt{\frac{K}{J_{s}}} C_{1} \sin(\sqrt{\frac{K}{J_{s}}}t)$$
(A.4)

The inital conditions are,

$$\omega_s(0) = 0$$
 and $\dot{\omega}_s(0) = \frac{K\theta_e}{J_s}$ (A.5)

where θ_e is the initial extension of the spring. Substituting Eq. (A.5) in Eq. (A.4) we have,

$$\omega_s(t) = \sqrt{\frac{K}{J_s}} \theta_e \sin(\sqrt{\frac{K}{J_s}}t) - \omega_m \cos(\sqrt{\frac{K}{J_s}}t) + \omega_m$$
(A.6)

Which is the time series solution for the movement of the spring immediately after the brake has been released. The required solution is to find the time when $\omega_s(t)$ equals $\omega_l(t)$ so that the velocities on either side of the clutch match. This means that Eq. (A.6) must be solved for time explicitly. This can be achieved by performing the following procedure:

Let $\tan(\frac{\theta}{2}) = u$ which then implies $\sin(\theta) = \frac{2u}{1+u^2}$ and $\cos(\theta) = \frac{1-u^2}{1+u^2}$, and substituting into Eq. (A.6) we obtain the following quadratic equation in u,

$$u^{2}(\omega_{s}-2\omega_{m})-\sqrt{\frac{\kappa}{J_{s}}}\theta_{e}^{2}u+\omega_{s}=0$$
(A.7)

Solving this equation and substituting back for t we have,

$$t = 2\sqrt{\frac{J_{s'K}}{K}} \tan^{-1} \left[\frac{\sqrt{\frac{K}{J_s}} \theta_e \pm \sqrt{\frac{K}{J_s}} \theta_e^2 - \omega_s (\omega_s - 2\omega_m)}{\omega_s - 2\omega_m} \pm n\pi \right]$$
(A.8)

This equation will allow the explicit calculation of the time it takes for the spring velocity to reach the load velocity and will also allow compensation for the clutch actuation delay. If this is known then it can be simply taken away from the time calculated from Eq. (A.8).

Appendix B - Calculating the timing of the clutches for the step-up/stepdown mechanism

This appendix describes how the controller calculates the timing of the clutches for the step-up/step-down mechanism. Two delay times are required a delay time between turning "off" the input clutch and turning "on" the output clutch (which is called the *transfer delay*) and a delay between turning "off" the output clutch and turning "on" the input clutch (which is called the *extraction delay*). These timings are shown, for the case of a positive output velocity, in **Fig. B.1** below,



Figure B.1 - Transfer and extraction delay times for positive load velocity

These timings can be calculated very accurately since in these periods both the clutches are "off" and the spring (with its attached components) is oscillating freely.

The problem can be thought of as finding the time taken for the spring to reach a target velocity (the motor or load velocity dependent on energy direction)

given an initial extension and velocity. For the extraction delay this initial extension will normally be close to zero but for the transfer delay it will be generally be nonzero. If we assume bearing friction to be negligible then the following differential equation will define the springs motion during these times,

$$0 = J_s \ddot{\theta}_s + K \theta_s \tag{B.9}$$

Which has the solution,

$$\theta_{s} = \theta_{e} \cos\left(\sqrt{\frac{K}{J_{s}}t}\right) + \frac{\omega_{e}}{\sqrt{\frac{K}{J_{s}}}} \sin\left(\sqrt{\frac{K}{J_{s}}t}\right)$$
(B.10)

and

$$\dot{\theta}_{s} = \omega_{e} \cos\left(\sqrt{\frac{K}{J_{s}}t}\right) + \sqrt{\frac{K}{J_{s}}}\theta_{e} \sin\left(\sqrt{\frac{K}{J_{s}}t}\right)$$
(B.11)

where,

$$\theta_s = \text{Spring position}$$

 $\theta_e = \text{Initial spring position (at t = 0)}$
 $\omega_e = \text{Initial spring angular velocity (at t = 0)}$
 $K = \text{Spring rate}$
 $J_s = \text{Inertia of spring (plus attached components)}$

This equation can be solved by using the following procedure:

Let $\tan(\frac{\theta}{2}) = u$ which then implies $\sin(\theta) = \frac{2u}{1+u^2}$ and $\cos(\theta) = \frac{1-u^2}{1+u^2}$, substituting in **Eqs. (B.10)** and (**B.11**) we have,

$$\dot{\theta}_{s} = \omega_{e} \frac{1 - u^{2}}{1 + u^{2}} - \sqrt{\frac{\kappa}{J_{s}}} \theta_{e} \frac{2u}{1 + u^{2}}$$
(B.12)

$$\dot{\theta}_{s}(1+u^{2}) = \omega_{e} - \omega_{e}u^{2} - \sqrt{\kappa_{f}}\theta_{e}^{2}u \qquad (B.13)$$

⇒

 \Rightarrow

$$u^{2}(\dot{\theta}_{s} + \omega_{e}) + u2\sqrt{\kappa_{f_{s}}}\theta_{e} + (\dot{\theta}_{s} - \omega_{e})$$
(B.14)

which is a quadratic equation with solution,

$$u = \frac{-2\sqrt{\kappa_{J_s}}\theta_e \pm \sqrt{4\kappa_{J_s}}\theta_e^2 - 4(\dot{\theta}_s + \omega_e)(\dot{\theta}_s - \omega_e)}{(\dot{\theta}_s + \omega_e)}$$
(B.15)

$$\Rightarrow \qquad t = 2\sqrt{J_{s'K}} \tan^{-1} \left[\frac{-\sqrt{K_{J_s}} \pm \sqrt{K_{J_s}} \theta_e^2 - (\dot{\theta}_s + \omega_e)(\dot{\theta}_s - \omega_e)}{(\dot{\theta}_s + \omega_e)} \pm n\pi \right] \qquad (B.16)$$

Let the target velocity be ω_t and so the required time delay (t_d) will be when $\dot{\theta}_s = \omega_t$ and so we have,

$$t_{d} = 2\sqrt{\frac{J_{s/K}}{K}} \tan^{-1} \left[\frac{-\sqrt{\frac{K}{J_{s}}} \pm \sqrt{\frac{K}{J_{s}}} \theta_{e}^{2} - (\omega_{t} + \omega_{e})(\omega_{t} - \omega_{e})}{(\omega_{t} + \omega_{e})} \pm n\pi \right]$$
(B.17)

This equation has two solutions generated by the '+' and '-' sign of the square root term and an infinity of solutions generated by the $\pm n\pi$ term. This equation is the basis for the calculation of the timing delays and must be interpreted correctly so that the correct solution is chosen for both positive and negative load velocities.

Since this equation needs to be solved on-line by the microprocessor it is crucial that this calculation takes a minimal amount of processing time. The calculation of the tan^{-1} argument will be dominant in terms of processing time ([20] gives and idea of how many processing steps these maths functions will take in practice) and so if prior knowledge of whether the '+' or '-' solution is the correct solution would almost halve the processing time required.

Let us first consider energy extraction. Fig. B.2 shows the correct quadrant for the target velocity (ω_t) when it is greater than zero and the correct quadrant when it is less than zero.



Figure B.2 - Quadrants containing correct solution for extraction delay

The correct quadrants will be when the sign of the spring extension equals the sign of target velocity and will mean that the force applied by the spring will be in the opposite direction to the movement of the target mass (either the motor or the load dependent on the energy direction).

Also shown on this graph are the positions of the '+' and '-' solutions for different target velocities. From this graph it is obvious that when $\omega_t > 0$ the '+' solution needs to be chosen and when $\omega_t < 0$ the '-' solution plus π needs to be chosen. This figure shows the case where $\omega_e > 0$ and from symmetry arguments the opposite solutions to the ones just mentioned are required when $\omega_e < 0$. In summary it can be concluded that correct the choice of solution can be found simply from the sign of the initial spring velocity (ω_e) and the sign of the target velocity (ω_t). These solutions are shown in **Table B.1**,

	$\omega_e \ge 0$	$\omega_e < 0$
$\omega_t \ge 0$	'+'	'+' + π
$\omega_t < 0$	'-' + π	'_'

Table B.1 - Choice of solution for energy extraction as a function of the sign of ω_e

and ω_t

Let us now consider the case of energy transfer. Fig. B.3 shows the equivalent quadrants for energy transfer to that shown for energy extraction in Fig. B.2.



Figure B.3 - Quadrants containing correct solution for transfer delay

For this situation the quadrants containing the correct solutions are the ones where the sign of the spring extension is opposite to the sign of the spring velocity. From this figure it can be seen that the correct solutions are the '+' solution for $\omega_t > 0$ and the '-' solution for $\omega_t < 0$. Again using symmetry arguments for the case of $\omega_e < 0$ we arrive at the equivalent table for energy transfer i.e.,

	$\omega_e \ge 0$	$\omega_e < 0$
$\omega_t \ge 0$	'-' + π	'_'
$\omega_t < 0$	'+'	'+' + π

Table B.2 - Choice of solution for energy transfer as a function of the sign of ω_e and

 ω_t

This table along with **Table B.1** allows the choice of solution to be made to Eq. (B.17) and without calling the tan^{-1} function first. The extra processing time in terms of the evaluation of two sign functions and a small look-up table is negligible to the processing time for an extra call to the tan^{-1} function.

Appendix C - Model file used for the simulation of step-up mechamisms

```
/*
      STEPUP_M.C
      Flexible drive rig states space equations
      (of the step-up variety)
      RSO 10/02/94
*/
#include "clubra.h"
#include "control.h"
#include "cmex.h"
double sign(double);
double fabs(double);
static double sample_rate;
/* pointers to channels */
static double *x_data, *xd_data, *u data;
static double *a_data,*d_data,*e_data,*ev_data;
/* define default physical parameters */
static double Kt,Ke,Jm,R,Vc,Kcm,K;
static double Cm, Bm, Jl, Js, Cs;
static double Bs,Cl,Bl,tb,tc;
static double N;
/* more of an integration parameter really */
static double velocity_tol=0.01;
/* needed for clutch and brake initialisation */
static double Cd, Bd;
static int brake_count,clutch_count;
double get_param( char *);
double get_sr( void );
/* initialisation routine */
void model init (void )
{
/* get sample rate */
     sample_rate=get_sr();
/* get parameters from list */
     Kt=get_param("Kt");
     Ke=get_param("Ke");
     Jm=get_param("Jm");
     R=get_param("R");
     Kcm=get_param("Kcm");
     K=get_param("K");
     Bm=get_param("Bm");
```

```
Cm=get_param("Cm");
     Jl=get_param("Jl");
     Js=get_param("Js");
     Cs=get_param("Cs");
     Bs=get_param("Bs");
     Cl=get_param("Cl");
     Bl=get_param("Bl");
     tb=get_param("tb");
     tc=get_param("tc");
     N=get_param("N");
/* initailise clutch and brake software */
     Bd=get_param("Bd");
     Cd=get_param("Cd");
     brake_count=(int) (Bd*sample_rate);
     clutch_count=(int) (Cd*sample_rate);
     init_brake( brake_count
                               , 1 );
     init_clutch( clutch_count , 0 );
/* now for the state space pointers */
     x_data = assign_channel(STATE,1);
     xd_data = assign_channel(STATE_DERIVATIVE,1);
     u_data = assign_channel(INPUT,1);
     d_data = assign_channel(DAC,1);
     e_data = assign_channel(ENCODER_DIS,1);
     ev_data = assign_channel(ENCODER_VEL,1);
}
void model_input( void )
{
/* motor voltage */
     u_data[0]=d_data[0];
/* clutch */
     u_data[1]=d_data[1];
/* brake */
     u_data[2]=d_data[2];
}
void model( void )
{
     int i;
     int c_on,b_on;
     Vc= u_data[0];
/* clutch state */
     c_on = (int) u_data[1];
/* brake state */
     b_{on} = (int) u_{data}[2];
/* test for motor locked */
     xd_data[0]=x_data[1];
     if ( (fabs(x_data[1])<velocity_tol) &&
           ( fabs((Kt*Vc*Kcm*N)/R-K*(x_data[0]-x_data[2])) <</pre>
Cm ) ) {
           xd_data[1]=0;
      } else {
           xd_data[1] = (1/Jm) * ( (Kt*Vc*Kcm*N)/R -
(Kt*Ke*x_data[1]*N*N)/R
```

```
- Cm*sign(x_data[1])
                       - Bm*x_data[1]
                       - K*(x_data[0]-x_data[2]) );
     }
/* test for clutch locked */
     xd_data[2]=x_data[3];
     xd_data[4]=x_data[5];
     if ( (fabs(x_data[3]-x_data[5])<velocity_tol) &&
           ( fabs( K*(x_data[2]-
x_data[0] + (b_on*tb+Cs) * sign(x_data[3])
                     +Bs*x_data[3]-Cl*sign(x_data[5])-
Bl*x_data[5])
                 < c_on*tc )) {
           xd_data[5]=1/(Js+J1)*(-K*(x_data[2]-x_data[0]) -
(Cs+Cl) * sign(x_data[3])
                        (B1+Bs) * x_data[3]
                       - b_on*tb*sign(x_data[3]) );
           xd_data[3]=xd_data[5];
/* Now see if brake locked */
           if ( (fabs(x_data[3])<velocity_tol) &&</pre>
                 ( fabs(K*(x_data[2]-x_data[0])) <
(Cl+Cs+b_on*tb)) ) {
                 xd_data[5]=0;
                 xd_data[3]=0;
           }
      } else {
/* check for brake being locked with clutch un locked ?? */
           if ( (fabs(x_data[3])<velocity_tol) &&
                 ( fabs( K*(x_data[2]-
x_data[0]+c_on*tc*sign(x_data[3]-x_data[5]))
                       < (Cs+b_on*tb) ) ) {
                 xd_data[3]=0;
            } else
                       {
                 xd_data[3] = (1/Js) * ( -K*(x_data[2]-x_data[0])
- Cs*sign(x_data[3])
                       - Bs*x_data[3]
                       - b_on*tb*sign(x_data[3])
                       - c_on*tc*sign(x_data[3]-x_data[5]) );
            }
           xd_data[5]=(1/J1)*(-c_on*tc*sign(x_data[5]-
x data[3]
                       -B1*x_data[5]
                       -Cl*sign(x_data[5]));
      }
      return;
}
void model_output( void )
{
/* motor position */
      e_data[0] = x_data[0];
/* motor velocity */
      ev_data[0]=x_data[1];
/* spring position */
      e_data[1] = x_data[2];
/* spring velocity */
      ev_data[1]=x_data[3];
```

```
/* load position */
    e_data[2] =x_data[4];
/* load velocity */
    ev_data[2]=x_data[5];
}

double sign( x )
double x;
{
    if (x== (double) 0.0 ) return 0.0;
    if (x < (double) 0.0) return -1.0;
    else return 1.0;
}</pre>
```
Appendix D - Model file used for the simulation of step-up/step-down mechanisms

```
/*
     STUSTD_M.C
     Flexible drive rig states space equations
     step-up/step-down configuration
     RSO 30/07/94
*/
#include <math.h>
#include "control.h"
#include "clubra.h"
#include "cmex.h"
double sign(double);
/* simulation params */
static double sample_rate;
/* define default physical parameters */
static double Kt,Ke,Jm,R,Vc,Kcm;
static double K;
static double Jm,Bm,Cm,Js,Bs,Cs,Jl,Bl,Cl;
static double ta,Cad,Car,tb,Cbd,Cbr;
static double velocity_tol=0.01;
static double tm,Jm_acc,Jl_acc,Js_acc;
/* channel pointers */
static double
*x_data, *xd_data, *u_data, *e_data, *ev_data, *d_data;
/* initialisation routine */
void model_init( void )
{
     int i;
     int delay_counta, delay_countb;
     int rise_counta, rise_countb;
/* get sample rate */
     sample_rate=get_sr();
/* get parameters from list */
     Kt=get_param("Kt");
     Ke=get_param("Ke");
     Jm=get_param("Jm");
     R=get_param("R");
     Kcm=get_param("Kcm");
     K=get_param("K");
     Bm=get_param("Bm");
     Cm=get_param("Cm");
     Jl=get_param("Jl");
     Js=get_param("Js");
     Cs=get_param("Cs");
```

```
Bs=get_param("Bs");
     Cl=get_param("Cl");
     Bl=get_param("Bl");
     ta=get_param("ta");
     Cad=get_param("Cad");
     Car=get_param("Car");
     tb=get_param("tb");
     Cbd=get_param("Cbd");
     Cbr=get_param("Cbr");
/* assign channels */
     x_data = assign_channel(STATE,1);
     xd_data = assign_channel(STATE_DERIVATIVE,1);
     u_data = assign_channel(INPUT,1);
     d_data = assign_channel(DAC,1);
     e_data = assign_channel(ENCODER_DIS,1);
     ev_data = assign_channel(ENCODER_VEL,1);
/* initialise clutches */
     delay_counta=Cad*sample_rate;
     rise_counta=Car*sample_rate;
     delay_countb=Cbd*sample_rate;
     rise_countb=Cbr*sample_rate;
     init_clutcha( delay_counta , rise_counta , 0 );
     init_clutchb( delay_countb , rise_countb , 0 );
}
void model_input( void )
{
/* motor voltage */
     u_data[0]=d_data[0];
/* clutcha */
     u_data[1]=d_data[1];
/* clutchb */
     u_data[2]=d_data[2];
}
void model (void )
{
     int i;
     double ca_on, cb_on;
     int clutcha_locked, clutchb_locked;
     double m_pos,m_vel,m_acc;
     double s_pos,s_vel,s_acc;
     double l_pos,l_vel,l_acc;
/* motor voltage */
     VC
          = u_data[0];
/* clutch a on/off */
     ca_on = u_data[1];
/* clutch b on/off */
     cb_on = u_data[2];
     m_pos=x_data[0];
     m_vel=x_data[1];
     s pos=x_data[2];
     s vel=x_data[3];
```

```
l_pos=x_data[4];
     l_vel=x_data[5];
/* define motor torque */
     tm=Kt*(Vc*Kcm/R - Ke*m_vel)/R;
set_diag((float) tm);
/*
   define state equations initially as
   inertia*acc's
*/
     Jm_acc=
      (tm-Bm*m_vel-Cm*sign(m_vel)
                                         -ca_on*ta*sign(m_vel-
s_vel));
     Js_acc=
      ( -Bs*s_vel-Cs*sign(s_vel)-K*s_pos-ca_on*ta*sign(s_vel-
m vel)
                               -cb_on*tb*sign(s_vel-l_vel) );
     Jl_acc=
      ( -Bl*l_vel-Cl*sign(l_vel)
                                        -cb_on*tb*sign(l_vel-
s_vel) );
/* check for clutcha locked */
     clutcha_locked=0;
      if ( fabs(m_vel-s_vel)<velocity_tol ) {</pre>
           if (fabs( (tm-Bm*m_vel
                         -Cm*sign(m_vel) )/Jm +
                               Bs*s_vel
                         (
                         +Cs*sign(s_vel)
                                +K*s_pos
                         +cb_on*tb*sign(s_vel-l_vel) )/Js )
                       < (ca_on*ta/Js+ca_on*ta/Jm) )
clutcha_locked=1;
      }
/* check for clutchb locked */
      clutchb_locked=0;
      if ( fabs(s_vel-l_vel)<velocity_tol ) {</pre>
           if (fabs( ( -Bs*s_vel
                      -Cs*sign(s_vel)
                       -ca_on*ta*sign(s_vel-m_vel) )/Js +
                       Bl*l_vel
                    (
                      +Cl*sign(l_vel) )/Jl ) <
(cb_on*tb/Js+cb_on*tb/Jl) )
                 clutchb_locked=1;
      }
/* define default solution */
     m_acc=Jm_acc/Jm;
      s_acc=Js_acc/Js;
      l_acc=Jl_acc/Jl;
/* Change solution if clutches are locked */
      if (clutcha_locked) {
           m_acc=(Jm_acc+Js_acc)/(Jm+Js);
           s_acc=m_acc;
           s_vel=m_vel;
      }
      if (clutchb_locked) {
           s_acc=(Js_acc+Jl_acc)/(Jl+Js);
            l_acc=s_acc;
```

```
l_vel=s_vel;
     }
     if (clutcha_locked && clutchb_locked) {
           m_acc=(Jm_acc+Js_acc+Jl_acc)/(Jm+Js+Jl);
           s_acc=m_acc;
           l_acc=m_acc;
           s_vel=m_vel;
           l_vel=m_vel;
     }
/* now store results */
     xd_data[0]=m_vel;
     xd_data[1]=m_acc;
     xd_data[2]=s_vel;
     xd_data[3]=s_acc;
     xd_data[4]=l_vel;
     xd_data[5]=l_acc;
     return;
}
void model_output( void )
{
/* motor position */
     e_data[0] =x_data[0];
/* motor velocity */
     ev_data[0] = x_data[1];
/* spring position */
     e_data[1] = x_data[2];
/* spring velocity */
     ev_data[1]=x_data[3];
/* load position */
     e_data[2] = x_data[4];
/* load velocity */
      ev_data[2]=x_data[5];
}
double sign( x )
double x;
{
      if (x== (double) 0.0) return 0.0;
      if (x < (double) 0.0) return -1.0;
      else return 1.0;
}
```

Appendix E Controller function used in stepup prototype and simulation

/* // STEPUP_C.C file for use with "control.c" to perform PWM control // of the flexible drive system. // PWM control with proportional and integral feedback // with resonant switching // With delayed clutch on time until spring and load velocities are matched // RSO 09/09/93 */ #include <math.h> #include <float.h> #include "clubra.h" #include "control.h" /* channel pointers */ static double *e_data, *ev_data, *r_data, *d_data; static int tkcount, tlcount, tccount, tncount; static int tki,tli,tci,tni; static int state,c_on,switching; static float Kp,Ki; static float vel_error,vel_error_prev; static float tk, tk prev; static float max_tk,tl,twist_tol,tn; static float motor_pos,spring_pos,spring_vel,load_pos,load_vel,motor_vel; static float ref_vel,motor_voltage,clutch_state,brake_state,ref_voltage; /* clutch time estimation variables */ static float Vs,Vm,thetae,rootkoj,tv; static double arg1, arg2, arg3; static int tvi,tvc; static float Vsm2Vm; static int clutch_time; static float c_time; static float sample rate; static float min_vel,max_vel; static int clutch_matching=1; /* switching subroutines */ void switch_pwm(void); void switch fdt(void); void switch_mv(void); /* clutch loitering routines */ void set_clutch_loiter(void);

```
void controller_init( void )
/* get sample rate */
     sample_rate=(float) get_sr();
/* controller gains */
     Kp=(float) get_param("Kp");
     Ki=(float) get_param("Ki");
/* twist tol */
     twist_tol=(float) get_param("twist_tol");
/* clutch_matching */
     clutch_matching=(int) get_param("clutch_matching");
/* root koj */
     rootkoj=(float) get_param("rootkoj");
/* clutch actuation time */
     c_time=get_param("clutch_time");
     clutch_time=(int) (c_time*sample_rate);
/* type of switching */
      switching=(int) get_param("switching");
/* PWM switching */
      if (switching==1) {
/* max value for tk */
           max_tk=(float) get_param("max_tk");
/* tl */
           tl=(float) get_param("tl");
           tlcount=(int) (tl*sample_rate);
      } else if (switching==2) {
/* RESONANT switching */
/* max value for tk */
           max_tk=(float) get_param("max_tk");
/* tn */
           tn=(float) get_param("tn");
           tncount=(int) (tn*sample_rate);
      } else if (switching==3) {
/* MOTOR RESONANT switching */
           min_vel=(float) get_param("min_vel");
           max_vel=(float) get_param("max_vel");
/* tn */
           tn=(float) get_param("tn");
           tncount=(int) (tn*sample_rate);
      }
/* assign channels */
      e_data = assign_channel(ENCODER_DIS,1);
      ev_data = assign_channel(ENCODER_VEL,1);
      r_data = assign_channel(REFERENCE,1);
      d_data = assign_channel(DAC,1);
/* set initial conditions */
      state=1;
      tki=0; tli=0;
     brake_on();
      clutch_off();
      vel_error=vel_error_prev=0;
      tk=tk_prev=0;
}
void controller( void )
{
      int i;
```

```
/* set DAC 0 to reference channel 0 */
     d_data[0]=r_data[1];
/* put other channel data into meaningful variables */
             = (float) r_data[0];
     ref_vel
     motor_pos = (float) e_data[0];
     motor_vel = (float) ev_data[0];
     spring_pos= (float) e_data[1];
     spring_vel= (float) ev_data[1];
     load_pos = (float) e_data[2];
     load_vel = (float) ev_data[2];
/* velocity error */
     vel_error=ref_vel-load_vel;
/* PI control */
     tk=tk_prev+Kp*(vel_error-vel_error_prev)+Ki*vel_error;
     tk_prev=tk;
     if (tk>max_tk) tk=max_tk;
     vel_error_prev=vel_error;
/* convert tk to a number of samples */
     tkcount=(int) (tk*sample_rate);
/* set up diagnostic to be value of tk */
/*
     diag=tk;*/
/* call different routines dependent on form of switching */
     if (switching==1) switch_pwm();
     if (switching==2) switch_fdt();
     if (switching==3) switch_mv();
     d_data[1] = (double) check_clutch();
     d_data[2] = (double) check_brake();
     return;
}
void user_fin( void )
{
     brake_off();
     clutch_off();
}
void switch_pwm( void )
{
     tli++;
     if (state==1) {
           tki++;
           if (tki>=tkcount) {
                 brake_off();
                 state=2;
/*
** calculate estimated time taken to achieve load velocity
*/
                 if (clutch_matching) set_clutch_loiter();
                                       tccount=(-1);
                 else
                 set_diag( (float) tccount );
                 tci=0;
           }
      }
      if (state==2) {
           tci++;
```

```
if (tci>=tccount) {
                 clutch_on();
                 state=3;
                 c_{on=1};
           }
     }
     if (state==3) {
           if (tli>=tlcount) {
                 clutch_off();
                 brake_on();
                 tki=0;
                 tli=0;
                 state=1;
           }
           if (c_on) {
                 if ( (motor_pos-spring_pos)<twist_tol ) {
                       clutch_off();
                       c_{on=0};
                 }
           }
     }
}
void switch_fdt( void )
{
     if (state==1) {
           tki++;
           if ( tki>=tkcount ) {
                 brake_off();
                 state=2;
/*
** calculate estimated time taken to achieve load velocity
*/
                 if (clutch_matching) set_clutch_loiter();
                                        tccount=(-1);
                 else
                 tci=0;
           }
     }
     if (state==2) {
           tci++;
           if ( tci>tccount ) {
                 clutch_on();
                 state=3;
           }
     }
     if (state==3) {
           if ( (motor_pos-spring_pos) < twist_tol ) {
                 clutch_off();
                 state=4;
                 tni=0;
           }
     }
     if (state==4) {
           tni++;
           if (tni>tncount) {
                 brake_on();
                 state=1;
                 tki=0;
```

```
}
     }
}
void switch_mv( void )
{
      if (state==1) {
            if ( motor_vel<min_vel ) {</pre>
                 brake_off();
                  state=2;
/*
** calculate estimated time taken to achieve load velocity
*/
                  if (clutch_matching) set_clutch_loiter();
                  else
                                        tccount=(-1);
                  tci=0;
            }
      }
      if (state==2) {
            tci++;
            if ( tci>tccount ) {
                  clutch_on();
                  state=3;
            }
      }
      if (state==3) {
            if ( (motor_pos-spring_pos) < twist_tol ) {</pre>
                  clutch_off();
                  state=4;
                  tni=0;
            }
      }
      if (state==4) {
            tni++;
            if ( tni>tncount && motor_vel>max_vel ) {
                  brake_on();
                  state=1;
                  tki=0;
            }
      }
}
void set_clutch_loiter( void )
{
      double Vs,Vm,thetae,arg1,arg2,arg3,tt,Vsm2Vm;
      Vs=load_vel;
      Vm=motor_vel;
      thetae=motor_pos-spring_pos;
      if (Vs>0.0) {
            Vsm2Vm=Vs-2.0*Vm;
            if (fabs(Vsm2Vm)<DBL_EPSILON) Vsm2Vm=1e-30;
            arg1=rootkoj*rootkoj*thetae*thetae-Vs*Vsm2Vm;
            if (arg1<0.0) arg1=0.0;
            arg2=sqrt(arg1);
            arg3=(rootkoj*thetae-arg2)/Vsm2Vm;
```

Appendix E

```
tt=(2.0/rootkoj)*atan(arg3);
tccount=tt*sample_rate;
tccount=tccount-clutch_time;
} else {
tccount=0 ;
}
return;
}
```

Appendix F Controller function used in stepup/step-down prototype and simulation

/* ** Controller for Flexible drive system (step-up/step-down config) ** Performing various switching strategies with PI feedback ** RSO 01/08/94 */ #include <math.h> #include "control.h" #include "clubra.h" #include "cmex.h" static int start_test, sample_no; static double sample_rate; static double *r_data,*d_data,*e_data,*ev_data; static double m_pos,m_vel,s_pos,s_vel,l_pos,l_vel; static double s_pos_prev; static double K, Js; static double max_tk,tl,tm=0.12,tn; static double Kp, Ki; static double tk,vel_error; static double tk_prev,vel_error_prev; static int state; static double twist_tol; static int tkcount, tlcount, tmcount, tncount, tccount; static int tki,tli,tmi,tni,tci; static int switching; static int direction=(+1); static double rootkoj; static int change_state,no_crosses; static int c_on; static double switch_vel,min_vel,max_vel; static double s_vel_prev; static double Es,Ess,Es_prev,K,Es_pos; static int first_point; static double motor_acc,m_vel_prev,motor_time,Cad; static int tai, tacount; static double clutch_delay; static int cdcount; static int E_direction,ttcount,tecount,tti,tei; static int output_direction;

```
static double tt, te;
static double max_sp,pi=3.14159265358979;
static double min_motor_vel,min_load_vel;
static double t_vel;
#define START_OF_CYCLE
                                  1
#define WAIT_ENERGY_EXTRACTION
                                  2
#define ENERGY_EXTRACTION
                            3
#define WAIT_ENERGY_TRANSFER
                                  4
#define ENERGY_TRANSFER
                                  5
void switch_pwm( void );
void switch_prs( void );
void switch_mrs( void );
void switch_eds( void );
void switch_bds( void );
void set_clutch_loiter( void );
int isign( double );
double extraction_wait(double, double , double );
double transfer_wait(double, double , double , double , int );
void controller_init( void )
{
      int i;
/* need sampe rate */
      sample_rate=get_sr();
/* set those parameters appearing int the p_strings list */
                 = get_param("Kp");
      Кр
                 = get_param("Ki");
      Ki
                      = get_param("rootkoj");
      rootkoj
      twist_tol = get_param("twist_tol");
      clutch_delay = get_param("clutch_delay");
      switching = (int) get_param("switching");
      if (switching==1) {
                            = get_param("max_tk");
           max_tk
                       = get_param("tl");
           tl
      } else if (switching==2) {
                            = get_param("max_tk");
           max_tk
                       = get_param("tn");
            tn
            switch_vel = get_param("switch_vel");
      } else if (switching==3) {
                             = get_param("min_vel");
           min_vel
                       = get_param("tn");
            tn
            switch_vel = get_param("switch_vel");
      } else if (switching==4) {
                       = get_param("tn");
            tn
                             = get_param("min_vel");
           min_vel
                       = get_param("K");
           Κ
      } else if (switching==5) {
                            = get_param("min_motor_vel");
           min_motor_vel
                            = get_param("min_load_vel");
           min_load_vel
                       = get_param("K");
            Κ
      }
/* assign channels */
```

```
r_data = assign_channel(REFERENCE,1);
     d_data = assign_channel(DAC,1);
     e_data = assign_channel(ENCODER_DIS,1);
     ev_data = assign_channel(ENCODER_VEL,1);
/* set other parameters that need to be initialised */
     state=1;
     tki=0;
     tli=0;
/* controller parameters */
     vel_error=vel_error_prev=0;
     tk_prev=0;
     tlcount=tl*sample_rate;
     tmcount=tm*sample_rate;
     tncount=tn*sample_rate;
     tacount=Cad*sample_rate;
     cdcount=clutch_delay*sample_rate;
     s_vel_prev=0.0;
     Es_prev=0;
     first_point=1;
     start_test=1;
     sample_no=0;
/* diagnostic */
     clutcha_off();
     clutchb_off();
}
void controller()
/* set DAC 0 to reference channel 0 */
     d_data[0]=r_data[1];
/* put other channel data into meaningful variables */
     m pos = e_data[0];
     m_vel = ev_data[0];
     s_{pos} = e_{data[1]};
     s_vel = ev_data[1];
      1_pos = e_data[2];
      l_vel = ev_data[2];
/* initially lets start the motor */
      if (start_test) {
           if (fabs(m_vel)>(min_motor_vel*1.3)) start_test=0;
           return;
      }
/* call different routines dependent on form of switching */
      if (switching==1) switch_pwm();
/* normal resonant switching */
      if (switching==2) switch_prs();
/* motor resonant switching */
      if (switching==3) switch_mrs();
/* energy demand motor resonant switching */
      if (switching==4) switch_eds();
/* bi-directional energy demand motor resonant switching */
      if (switching==5) switch_bds();
```

```
s_vel_prev=s_vel;
     m_vel_prev=m_vel;
     d_data[1] = (double) check_clutcha();
     d_data[2] = (double) check_clutchb();
}
void switch_bds( void )
     double Kpp;
/* velocity error */
     if (r_data[0]>0) vel_error = r_data[0]-l_vel;
                  vel_error = l_vel-r_data[0];
     else
/* bodge to find out whas going on */
if (r_data[0]==0) {
Es_prev=0.0;
vel_error_prev=0.0;
}
/* PI control */
     Kpp=Kp*fabs(r_data[0]);
/*
     Es=Es_prev+Kpp*(vel_error-vel_error_prev)+Ki*vel_error;
     Es_prev=Es;
     vel_error_prev=vel_error; */
Es=Kpp*vel_error;
/* determine energy direction ! */
      if (state==START_OF_CYCLE) {
/* determine energy direction and size of energy packet */
           if (Es>=0) {
                 E_direction=1;
                 Ess=Es;
                 te=extraction_wait(rootkoj,s_vel,m_vel);
                 min vel=min_motor_vel;
                 } else {
                       E_direction=-1;
                 Ess = -Es;
                 te=extraction_wait(rootkoj,s_vel,l_vel);
                 min_vel=min_load_vel;
           }
           tecount=(int) (te*sample_rate);
           tecount=tecount-cdcount;
           if (tecount<0) tecount=0;
           tei=0;
           state=WAIT_ENERGY_EXTRACTION;
/* bodge so that it might go at slow speeds? */
           if (fabs(Es)<0.5) state=START_OF_CYCLE;*/
/*
      }
      if (state==WAIT_ENERGY_EXTRACTION) {
           tei++;
           if (tei>tecount) {
/* this calculates the required spring extension for the
required energy packet */
                 Es_pos=sqrt(2*Ess/K + s_pos*s_pos);
                 if (E_direction==1) clutcha_on();
                                      clutchb_on();
                 else
                 state=ENERGY_EXTRACTION;
           }
```

```
}
     if (state==ENERGY_EXTRACTION) {
           if (E_direction==1) t_vel=m vel;
           else
                           t_vel=l_vel;
           if ( fabs(s_pos) >= Es_pos || fabs(t_vel)<min_vel )</pre>
{
                 if (E_direction==1) {
                      clutcha_off();
                 } else {
                      clutchb_off();
                 }
/* calculalate time for velocity matching */
                 if (E_direction==1) {
/* determine ouput power direction from reference velocity */
                       if (r_data[0]>0) output_direction=1;
                                   output_direction=-1;
                       else
     tt=transfer_wait(rootkoj,s_pos,s_vel,l_vel,output_direct
ion);
                 } else {
                       output_direction=1;
     tt=transfer_wait(rootkoj,s_pos,s_vel,m_vel,output_direct
ion);
                 }
                 ttcount=(int) (tt*sample_rate);
                 state=WAIT_ENERGY_TRANSFER;
                 tti=0;
            }
      }
      if (state==WAIT_ENERGY_TRANSFER) {
           tti++;
            if (tti>ttcount) {
                 if (E_direction==1) clutchb_on();
                                      clutcha_on();
                 else
                 state=ENERGY_TRANSFER;
            }
      }
      if (state==ENERGY_TRANSFER) {
           if (output_direction==1) s_pos=-s_pos;
                  fabs(s_pos)<=fabs(s_vel*clutch_delay) ) {</pre>
            if (
                 if (E_direction==1) clutchb_off();
                                      clutcha_off();
                 else
                 state=START_OF_CYCLE;
            }
      }
}
double extraction_wait(rootkoj,spring_vel,input_vel)
double rootkoj,spring_vel,input_vel;
{
      double arg1,arg2;
      if (fabs(spring_vel)<=fabs(input_vel)) return 0.0;
      arg1=sqrt((spring_vel-
input_vel)*(spring_vel+input_vel));
      arg2=arg1/(spring_vel+input_vel);
      if (isign(spring_vel)==-1 && isign(input_vel)==1 ) {
```

```
return (2.0/rootkoj)*(atan(arg2)+pi);
     } else if (isign(spring_vel)==1 && isign(input_vel)==-1
)
 {
           return (2.0/rootkoj)*(atan(-arg2)+pi);
     } else {
           return (2.0/rootkoj)*(atan(arg2));
     }
}
double
transfer_wait(rootkoj,spring_pos,spring_vel,output_vel,output_
direction)
double rootkoj, spring_pos, spring_vel, output vel;
int output_direction;
{
     double arg1,arg2,argp,argn,sarg1,argmax,argmin,argret;
     arg1=(rootkoj*rootkoj*spring_pos*spring_pos) -
(output_vel+spring_vel) * (output_vel-spring_vel);
/* this means there's not enough energy to match velocities */
      if (arg1<=0) {
           if (output_direction==1) {
/* this should be the time the time taken for the spring
veloity to reverse sign */
      argret=(atan(spring_pos*rootkoj/spring_vel)+pi)/rootkoj;
                 return argret;
           } else {
                 return 0;
           }
      }
/* calculate both arguments */
        sarg1=sqrt(arg1);
      arg2=(output_vel+spring_vel);
      if (arg2==0) return 0.0;
      argp=atan((-rootkoj*spring_pos+sarg1)/arg2);
      argn=atan((-rootkoj*spring_pos-sarg1)/arg2);
/* add pi if any negative */
      if (argp<0) argp=argp+pi;
      if (argn<0) argn=argn+pi;</pre>
      argmax=argp;
      argmin=argn;
      if (argmax<argmin) {
           argmax=argn;
           argmin=argp;
      }
      argret=argmin;
      if (output_direction==1) {
            if (spring_vel>output_vel) argret=argmax;
      }
      return (2.0/rootkoj)*argret;
}
int isign( value )
double value;
```

```
{
    if (value>=0.0) return 1;
    else return -1;
}
void controller_fin( void )
{
    clutcha_off();
    clutchb_off();
}
```

. .

Appendix G - Component data sheets for stepup prototype



TORQUE 1.1 Nm. (10 lb.ins.) MAXIMUM UP TO $\frac{1}{8}$ H.P. AT 1440 R.P.M. CYCLE RATES CAN EXCEED 10,000 STARTS/STOPS PER HOUR WHERE LOAD INERTIA PERMITS FULLY SELF-ADJUSTING NO MAINTENANCE REQUIRED CHOICE OF SHAFT DIAMETERS $\frac{5}{16}$ or 8mm.



The Clark Composite Clutchbrake unit is a combination assembly of the well proven Clark model 175 Clutch and Brake. Integral input and output shafts running in substantial sealed bearings reduce fitting to the ultimate in simplicity and low cost.

The Clutch and Brake torque may be fixed at maximum or preset to a reduced value. For exceptionally smooth starting, the Clark **Power Unit type 1024/2R** incorporates an inexpensive **"Silkstart"** electronic control which raises the clutch voltage from zero to 24V over a period which can be preset between $\frac{1}{4}$ second and 10 seconds. **"Boost"** circuits can provide very high acceleration and stopping rates i.e. for high cycle rate indexing drives.

Comprehensive applications advice from address overleaf.





General Specification

Maximum Static Torque	:	1.1 Nm (10 lb. ins.).		
Maximum Speed	:	8,000 r.p.m.		
Standard Coil Windings	:	24 volts D.C. 0.25 Amp. 97 Ohms. Continuously rated.		
Other Voltages available	:	6, 12, 50, 90 Volts D.C.		
Weight	:	0.9 Kg (2 lb)		
Maximum Heat Dissipation (Slipping)				
Input Speed 0-500 r.p.m.	:	870 Nm/min (640 ft.lb/min)		

nput Speed	0-500 r.p.m. :	870 Nm/min (640 ft.lb/min)
	1000 r.p.m. :	1333 Nm/min (980 ft.lb/min)
	1500 r.p.m. :	1500 Nm/min (1100 ft.lb/min)
	3000 r.p.m. :	1800 Nm/min (1325 ft.lb/min)



SERIAL NO

VOLTAGE

-

Printed Armature DC Servo Motor

G12M4

0.3887 N-MOTOR RATINGS WEIGHT Continuous torque at rated speed 55 oz in 8 1b (3.63 kg) Pulse torcue (50 ms at 1" e duty 1200 oz in cyc:e) NOTES 3650 rev/min Rated speed 1. Motor is tested at this voltage for convenience. Other Rated voltage " 48 Vdc voltages may be used provided maximum armature Power output at rated speed 147 W dissipation is not exceeded. ($P_{max} = P_{in} - P_{out} = constant$). Rated current 44A 2. The speed-torque curve is obtained by using the maximum 7.5 A Maximum continuous stall current terminal resistance of the motor at 150 °C armature 0.75Ω Terminal resistance temperature (worst condition). 3. Calculated from the formula: MOTOR CONSTANTS Torque constant (K.) 7.01 x 10 -6 x (Pulse torque)2 15.6 oz in/A Emf constant (K.) 11.5 V/1000 rev/min Inertia Damping constant (Kg) 3.1 oz in/1000 rev/min GENERAL Total inertia (J_{τ}) Regulation at constant voltage 0.020 oz in s² 1. Maximum allowable armature dissipation: $P_{max} = \frac{150 \text{ °C} \text{ -T}_{ambient} (\text{ °C})}{2}$ 5.85 rev/min/oz in (Rm)(2) $\theta_{ac} + \theta_{ca}$ Armature inductance (L_) <100 µH Average friction torque (Tr) 4.0 oz in 2. The curves for forced cooling operation were obtained by Mechanical time constant²¹ 0.0126 s modifying the mechanical configuration of the motor to accept the required air flow. These motors are available on Power rate (3) 507 kWis special request. THERMAL RESISTANCE The maximum allowable armature dissipation in this case is calculated as follows: Armature-to-case (6a.c) 1.15 deg C/W 150 °C-Tampient (°C) Case-to-ambient (6 ...) P.....= θ, With 8 x 16 x i in aluminium 0.87 deg C/W 3. Mass air flow (lb/min) = Air volume (ft3/min) x density (lb:ft2) heat sink With 14 x 14 x } in aluminium 0.70 deg C/V/ heat sink All nominal values at 25 °C ambient except where otherwise stated Forced cooling Armature-to-ambient (6.) With mass air flow of 0.4 lb/min 0.8 deg C/W 0.51 deg C/W 0.28 deg C/W With mass air flow of 0.8 lb/min With mass air flow of 2.0 lb/min

The figures quoted above for the motor constants are typical and cannot be guaranteed unless a technical specification has been negotiated.





- 2

AVERAGE PERFORMANCE CHARACTERISTICS

.





Printed Motors Limited

Eordon Trading Estate, Cakhanger Road, Bordon, Hants, GU359H ((Registered Office and Works) Telephone, Bordon (04203) 3033 (4 lines) Telex, 853766 A MEMBER OF THE Technograph GROUP **Appendix H - Photograph of step-up prototype**

• ~



Appendix I - Reference Guide to the MATLAB CAD toolbox of step-up/stepdown mechanisms

List of functions

cadjs

cadk

effsusd

newsusd

simsusd

springsz

srmin

CADJS

Purpose

To perform initial component selection for flexible drive systems.

Synopsis

```
[tca,K,xf,Sr]=cadjs(P,Srmin,wmin,wmax,Vm,Js)
cadjs(P,Srmin,wmin,wmax,Vm,Js)
[tca,K,xf,Sr]=cadjs(P,Srmin,wmin,wmax,Vm)
cadjs(P,Srmin,wmin,wmax,Vm)
```

Description

This routine will generate vectors containing clutch torques ("tca"), spring K factors ("K"), maximum spring extensions ("xf") and maximum switching rates ("Sr"). The size of these vectors will be the same as the size of the "Js" vector. If no "Js" vector is given then a default vector will be used ([1e-5:1e-5:1e-4]). If no left hand arguments are used the results will be plotted. Any one of the parameters "P", "Srmin", "wmin", "wmax" or "Vm" can be a vector and in which case multiple plots will be produced (no left hand arguments)

See also

CADK, SRMIN

CADK

Purpose

To perform initial component selection for flexible drive systems.

Synopsis

```
[tca,K,xf,Sr]=cadk(P,Srmin,wmin,wmax,Vm,K)
cadk(P,Srmin,wmin,wmax,Vm,K)
[tca,K,xf,Sr]=cadk(P,Srmin,wmin,wmax,K)
cadk(P,Srmin,wmin,wmax,K)
```

Description

This routine will generate vectors containing clutch torques ("tca"), spring inertias ("Js"), maximum spring extensions ("xf") and maximum switching rates ("Sr"). The size of these vectors will be the same as the size of the K vector. If no K vector is given then a default vector will be used. If no left hand arguments are used the results will be plotted. Any one of the parameters "P", "Srmin", "wmin", "wmax" or "Vm" can be a vector and in which case multiple plots will be produced (with no left hand arguments).

See also

CADJS, SRMIN

EFFSUSD

Purpose

To perform efficiency analysis on flexible drive systems

Synopsis

```
[vel, input_power, ouputpower, perc_eff, input_bearing,
    spring_bearing, output_bearing, input_clutch,
    output_clutch] = effsusd(model, load_torque)
```

Description

This function will perform multiple simulation runs using model file "model" performing a separate run for each of the values in the load torque vector "load_torque". If left hand side arguments are given then matrices showing the steady state velocities reached ("vel"), the percentage input/output efficiency ("perc_eff") and the power lost to the three bearings ("input_bearing", "spring_bearing" and "output_bearing") and the two clutches("input_clutch" and "output clutch"). If no left hand side arguments are given then the results will be plotted on the screen.

Example

```
effesusd('susd',[0:0.1:1]);
```

This will perform efficiency analysis useing parameter definition file "susd.m" and eleven ouput load torques ranging from 0 to 1 Nm. The results will be automaically plotted on the screen.

See also

.

NEWSUSD, SIMSUSD

NEWSUSD

Purpose

To create a new parameter file in preparation for the simulation of a stepup/step-down flexible drive system.

Synopsis

newsusd(model)

Description

This function will create a parameter file in preparation for a simulation of a step-up/step-down mechanism. If the string entered was "xxxxxx" then the file produced will be a MATLAB script file called "xxxxxx.m". This model file will be used by SIMSUSD and EFFSUSD for perform simulations and efficiency analysis.

Some parameters will be set to default values and others will be left blank to be filled in by the user using any convenient text editor.

Example

```
newsusd('susd')
```

will create a parameter file called "susd.m"

See also

NEWSUSD, EFFSUSD

SIMSUSD

Purpose

To simulate step-up/step-down flexible drive mechanisms

Synopsis

```
[t,m_pos,m_vel,s_pos,s_vel,l_pos,l_vel,clutcha,clutchb]
=simsusd(model,str1,val1,str2,val2,...str12,val12)
```

Description

This function will use the parameter definitions held in the three m-files defined by the model name "model" to simulate a step-up/step-down flexible drive system. If no left hand side arguments are given then the results will be plotted. If extra right hand side arguments are given then these can be used to overwrite parameters held in the model parameter file.

Example

```
simsusd('susd1');
will simulate model 'susd1' and plot the results
susd('susd1', 'K', 0.9, 'Jm', 0.01)
will simulate model 'susd1' using K=0.9Nm and Jm=0.01 and plot the results
```

See also

NEWSUSD, EFFSUSD

SRMIN

Purpose

To calculate minimum switching rate necessary to achieve a desired velocity output ripple

Synopsis

Sr=Srmin(P,wmin,wrip,Jl)

Description

Defining the motor power, "P" a nominal output velocity "wmin", velocity ripple "wrip" and load inertia "JI" the user can find the necessary minimum switching rate, "Sr". It is based on the following equation

$$Sr_{min} = \frac{P}{W_{min}W_{rip}J_{l}}$$

See also

à,

CADJS, CADK, SPRINGSZ

SPRINGSZ

Purpose

To help size a torsional helical spring of constant wire and coil diameter.

Synopsis

[d, DN, V] = springsz(K, xmax, fatigue)

Description

The user has to provide a spring K-factor("K") and maximum spring extension "xmax". This function will provide "d" the wire diameter and "DN" the product of the number of active coils and the coil diameter. Hence this routine will provide a family of springs all having the same "K" factor and able to withstand the maximum spring extension. "fatigue" is a factor between 0 and 1 that is multiplied by the ultimate tensile stress to form the working stess for these calculations. "fatigue" equal to 0.25 (the default) will allow the spring to have infinite fatigue life.

See also

CADJS, CADK, SRMIN

Appendix J - Parameter model file used in simulation of 500 watt example design

```
% MODEL PARAMETERS
% Example simulation of a 500watt device
8
% define motor model as a motor velocity vs. torque
characteristic
£
Vm=[0 200];
% power is torque X velocity
Tm = 10 - 0.05 * Vm;
% Motor inertia
Jm=0.005;
% Input bearing
Cm=0.0025;
Bm=8.9e-5;
% Spring K factor (C1100 112 2250 SS)
K = 7.07;
% load inertia (use LARGE MASS)
J1=0.005;
% brake center inertia (latest measured value)
%Js=9e-5;
Js=1e-4;
% spring friction (from MSD data )
% same as load friction
% for the spring use double the output load bearing values
Bs=8.9e-5*2; Cs=0.0025*2;
% load friction
Cl=0.0025; Bl=8.9e-5;
% Torque of brake (had to beef this up a bit from manu's
data)
ta=20;
% Torque of clutch (had to beef this up a bit from manu's
data)
tb=20;
% brake and clutch delay (secs)
Cad=0.004; Cbd=0.004;
% Load torque
t1=0.8;
% CONTROLLER PARAMETERS
sample_rate=1000;
                       % 1 for closed loop 0 for open loop
closed_loop=1;
Kp=J1/2-(1/30)*((1-t1)/400+t1/(2*400));
%Kp=J1/2-(1/30)*((1-0.8)/400+0.8/(2*400));
Ki=J1/2;
                      % approx of root(K/J)
rootkoj=sqrt(K/Js);
spring_max=2.86; % maximum spring extension in radians
clutch_delay=0.004; % clutch matching delay
```

Į

```
% min motor velocity
min_motor_vel=70;
                       % integral anti-windup action
anti_windup=1;
% SIMULATION PARAMETERS
% set up default simulation parameters
start_time=0;
ref_signal=[0 +400;3 +400;3.01 -400; 6 -400;10 +400; 11 400;
12 0; 13 0];
min_step=1e-7;
max_step=0.0019;
tol=1e-3;
mode=2;
% initial conditions with motor velocity=100
\mathbf{x}0=[0 0 0 0 0 0];
u0=[0 \ 0];
```

end

Appendix K - Engineering drawing of stepup/step-down mechanism




Appendix L - Photograph of step-up/step-down prototype

Appendix L





References

- [1] Handbook of Mechanical Power Drives. Trade and technical Press Ltd, London, 1980.
- [2] P.G.Gett. Changing Gears: The Development of the Automobile Transmission, Society of Automotive Engineers Inc, Warrendale, USA, 1991.
- [3] E. Spijker. Steering and Control of a CVT Based Hybrid Transmission, CIP-CEGEVENS KONINKLIJKE, Neth, 1995.
- [4] C.V. Lezards. Hydraulic Variable Speed Drives, Engineers Digest Survey No. 18, 1964.
- [5] K.G.Bush. Electrical Variable Speed Drives, Blackwell Sciences Ltd, London, 1995.
- [6] M. Brown. Practical Switching Mode Power Supply Design. Academic Press Inc, San Diego California, 1990.
- [7] R.P. Severns, G. Bloom. Modern DC-DC Switchmode Power Converter Circuits. Van Nostrand Reinhold.
- [8] P.M. Heldt. Torque Converters and Transmissions, 4th Edition NYACK, N. Y., 1951.
- [9] F. Williams, D. Tipping. A Mechanical Torque Converter and its Use as an Automobile Transmission, Proc I. Mech. E Vol 190 pp 447-456, 1976.
- [10] H.F. Olsen. Dynamical Analogies, Van Nostrand Reinhold, New Jersey, 1958.
- [11] P.E. Wellstead. Introduction to Physical System Modelling, Academic Press, London, 1979.
- [12] H.M. Paytnor. Analysis and design of Engineering Systems, MIT Press, Cambridge, Mass, 1961.
- [13] B. Armstrong-Helouvry. Control of Machines with Friction, Normwell, MA, Kluwer Academic Press, 1991.
- [14] T. Soderstrom, P. Stoica System Identification, Prentice Hall, 1989.
- [15] B. Armstrong-Helouvry. Stick Slip and Control in Low Speed Motion. IEEE Trans. Automat. Contr., vol. 38, no. 10, pp 1483-1496, Oct 1993.
- [16] G.E.Forsythe, M.A.Malcolm, C.B.Moler Computer Methods for Mathematical Computations, Prentice Hall, 1977.
- [17] Mathworks Inc. MATLAB Users Guide. 1993
- [18] SAE Spring Committee Spring Design Manual AE-11, Warrendale, PA, SAE International, pg 2.2.
- [19] SAE Spring Committee Spring Design Manual AE-11, Warrendale, PA, SAE International, pg 2.12.

- [20] C.W.Clenshaw. Mathematical Tables volume 5, Chebyshev Series for Mathematical Functions, H.M.S.O, London, 1966.
- [21] R. F. Steidel An Intoduction to Mechanical Vibrations, Wiley & sons, New York, Chapter 6, 1989.
- [22] Allspeed Ltd, The Allspeed Mechanical Variators Brochure, P.O Box 43, Accrington, Lancs BB5 5LP
- [23] D. Baker. Putting teeth into clutch applications, Power Tansm. Des. v 27, 6th June 1985, pp55-58.
- [24] K. Schlicting, J. Schuetzer, ZF Electromagnetic toothed Clutches, Siemmans Rev. n-3, 1976, pp130-132.
- [25] J.E. Slotine, W. Li, Applied Nonlinear Control, Prentice-Hall, New Jersey, 1991.
- [26] V.I. Utkin, Variable Structure Systems with Sliding Modes, IEEE Trans on Automatic control, vol AC-22 no. 2 pp210-222, 1977.
- [27] U. Itkis. Control Systems of Variable Structure, Wiley & sons, New York, 1976.
- [28] A.R. Johnson, J Makin, W.A. Bullough. Electrorheological catch/clutch -Inertial Simulations, International Journal on Modern Physics, vol. 8, pp 2935-2954, 1994.
- [29] M. Whittle, R.J. Atkin, W.A. Bullough . Fluid Dynamic Limitations on the Performance of an Electrorheological Clutch, Journal of Non-Newtonian Fluid Mechanics, vol. 57 pp 61-81, 1995.
- [30] G.J. Monkman. Dielectrophoretic Enharncement of Electrorheological Robotic Actuators, Mechatronics, vol. 3 pp 305-313, 1993.
- [31] J. Thornley, M. Preston, T. King. A Fast Electro-Mechanical Clutch Element Using a Piezoelectric Multilayer Actuator, IEE Colloquium on Robot Actuators, pp 1/1-1/4, Oct. 1991.
- [32] J. Thornley, M.E. Preston, T.G. King, A Very High Speed Piezoelectrically Actuated Clutching Device, Mechatronics, vol. 3, pp 295-304, 1993.
- [33] J. Thornley, Methods of Application of Piezoelectric Multilayer Actuators to High-Speed Clutching Using Displacement Amplification, PhD Thesis, Loughborough University of Tech., Jan. 1993.