

CONTENTS

1	Introduction	1
1.1	System Overview	1
1.2	Manual Overview	2
2	System Description & Operating Instructions	3
2.1	ECP Executive Software	3
2.2	Electromechanical Plant	32
2.3	Safety	40
3	Start-up & Self-Guided Demonstration	43
3.1	Hardware Set-up Verification	43
3.2	Demonstration Of ECP Executive Program	45
3.3	Nonintuitive System Behavior	54
4	Real-time Control Implementation	55
4.1	Servo Loop Closure	56
4.2	Command Generation	58
4.3	Servo Motor & Amplifier	62
4.4	Multi-tasking Environment	67
4.5	Sensors	67
4.6	Auxiliary Analog Output	69
5	Plant Dynamic Models	70
5.1	Two Degree of Freedom Plants	70
5.2	Three Degree of Freedom Plants (Model 205a only)	73
6	Experiments	76
6.1	System Identification	76
6.2	Rigid Body PD & PID Control	82
6.3	Disturbance Rejection of Various Controllers	88
6.4	Colocated PD With 2 DOF Plant	90
6.5	Noncolocated PD Plus Notch Filter	92
6.6	Successive Loop Closure / Pole Placement	95
6.7	LQR Control	99
6.8	Practical Control Implementation	102
6.8.1	Effect of Drive Saturation	103
6.8.2	Effect of Discrete-time Sampling	104
6.8.3	Effect of Finite Wordlength & Sensor Quantization	105
A	Dynamic Modeling Details	107
A.1	Dynamics Of Ideal Plant	107
A.2	Practical Plant Model	113
A.3	One Degree Of Freedom Plants	118

A.4 Three Degree of Freedom Plants (Model 210a only) 118

A	Dynamic Modeling Details	107
A.1	Dynamics Of Ideal Plant	107
A.2	Practical Plant Model	113
A.3	One Degree Of Freedom Plants	118
A.4	Three Degree of Freedom Plants (Model 210a only)	118

(Instructor's Manual Only)

6i	Instructor's Supplement To Experiments	124
6.1i	System Identification	125
6.2i	Rigid Body PD & PID Control	134
6.3i	Disturbance Rejection of Various Controllers	144
6.4i	Colocated PD With 2 DOF Plant	148
6.5i	Noncolocated PD Plus Notch Filter	157
6.6i	Successive Loop Closure / Pole Placement	166
6.7i	LQR Control	174
6.8i	Practical Control Implementation	179
6.8.1i	Effect of Drive Saturation	179
6.8.2i	Effect of Discrete-time Sampling	183
6.8.3i	Effect of Finite Wordlength & Sensor Quantization	184
6.9i	Sensitivity to Parameter Changes	185
6.10i	Effect of Output Disturbance On Multi-DOF Systems	188
6.10i	Suggested Further Experiments	191
Ai	Useful Scripts	194
A.1i	Plant Model Builder	194
A.2i	Notch Filter Designer	196
A.3i	Successive Loop Control Designer	196
A.2i	LQR Synthesis	198

1 Introduction

Welcome to the ECP line of educational control systems. These systems are designed to provide insight to control system principles through hands-on demonstration and experimentation. Shown in Figure 1.1-1, each

consists of an electromechanical plant and a full complement of control hardware and software. The user interface to the system is via a friendly, versatile, PC window environment which supports a broad range of controller specification, trajectory generation, data acquisition, and plotting features. The systems are designed to accompany introductory through advanced level controls courses and support either high level usage (i.e. direct controller specification and execution) or detailed user-written algorithms.

The electromechanical apparatus may be transformed into a variety of dynamic configurations which represent important classes of "real life" systems. The Model 210 spring/mass apparatus represents many such physical plants including rigid bodies; flexibility in linear drives, gearing and belts; and coupled discrete vibration with actuator at the drive input and sensor collocated or at flexibly coupled output (noncollocated). Thus the plant models may range from a simple double integrator to a fourth order^[1] case with two lightly damped poles and either two or no zeros.

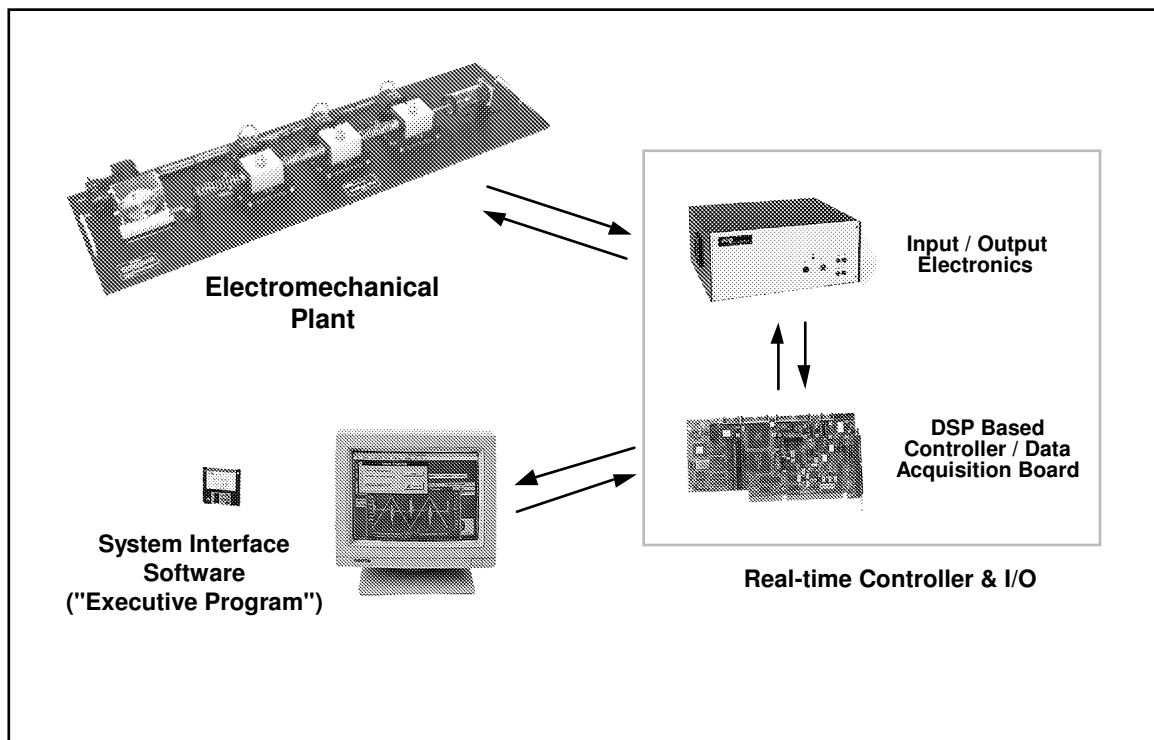


Figure 1.1-1. The Experimental Control System

1.1 System Overview

The experimental control system is comprised of the three subsystems shown in Figure 1.1-1. The first of these is the electromechanical plant which consists of the spring/mass mechanism, its actuator and sensors. The design features a brushless DC servo motor, high resolution encoders, adjustable masses, and reconfigurable plant type.

Next is the real-time controller unit which contains the digital signal processor (DSP) based real-time controller^[2], servo/actuator interfaces, servo amplifier, and auxiliary power supplies. The DSP is capable of executing control laws at high sampling rates allowing the implementation to be modeled as being continuous or discrete in time. The controller also interprets trajectory commands and supports such functions as data acquisition, trajectory generation, and system health and safety checks. A logic gate array performs motor commutation and encoder pulse decoding. Two optional auxiliary digital-to-analog converters (DAC's) provide for real-time analog signal measurement. This controller is representative of modern industrial control implementation.

The third subsystem is the executive program which runs on a PC under the DOS or Windows[™] operating

system. This menu-driven program is the user's interface to the system and supports controller specification, trajectory definition, data acquisition, plotting, system execution commands, and more. Controllers may assume a broad range of selectable block diagram topologies and dynamic order. The interface supports an assortment of features which provide a friendly yet powerful experimental environment.

1.2 Manual Overview

The next chapter, Chapter 2, describes the system and gives instructions for its operation. Section 2.3 contains important information regarding safety and is mandatory reading for all users prior to operating this equipment. Chapter 3 is a self-guided demonstration in which the user is quickly walked through the salient system operations before reading all of the details in Chapter 2. A description of the system's real-time control implementation as well as a discussion of generic implementation issues is given in Chapter 4. Chapter 5 presents dynamic equations useful for control modeling. Chapter 6 gives detailed experiments including system identification and a study of important implementation issues and practical control approaches.

2 System Description & Operating Instructions

This chapter contains descriptions and operating instructions for the executive software and the mechanism. The safety instructions given in Section 2.3 must be read and understood by any user prior to operating this equipment.

2.1 ECP Executive Software

The ECP Executive program is the user's interface to the system. It is a menu driven / window environment that the user will find is intuitively familiar and quickly learned - see Figure 2.1-1. This software runs on an IBM PC or compatible computer and communicates with ECP's digital signal processor (DSP) based real-time controller. Its primary functions are supporting the downloading of various control algorithm parameters (gains), specifying command trajectories, selecting data to be acquired, and specifying how data should be plotted. In addition, various utility functions ranging from saving the current configuration of the Executive to specifying analog outputs on the optional auxiliary DAC's are included as menu items.

2.1.1 The DOS Version of the Executive Program

2.1.1.1 PC System Requirements

For the ECP Executive (DOS version), you will need at least 2 megabyte of RAM and a hard disk drive with at least 4 megabytes of space. All DOS versions of the Executive program run under any of DOS versions 3.x, 4.x, 5.x, and 6.x. The Executive requires a VGA monitor with a VGA graphics card installed on the PC.

The Executive Program runs best on a 386, 486, or Pentium® based PC with 4 megabytes or more of memory under DOS 5.0 or higher with HIGHMEM.SYS driver included in your CONFIG.SYS file.^[3] Also, if the software does not "see" at least 2 megabytes of free RAM, you may find the program executing somewhat slowly since it will use the hard disk as virtual memory.

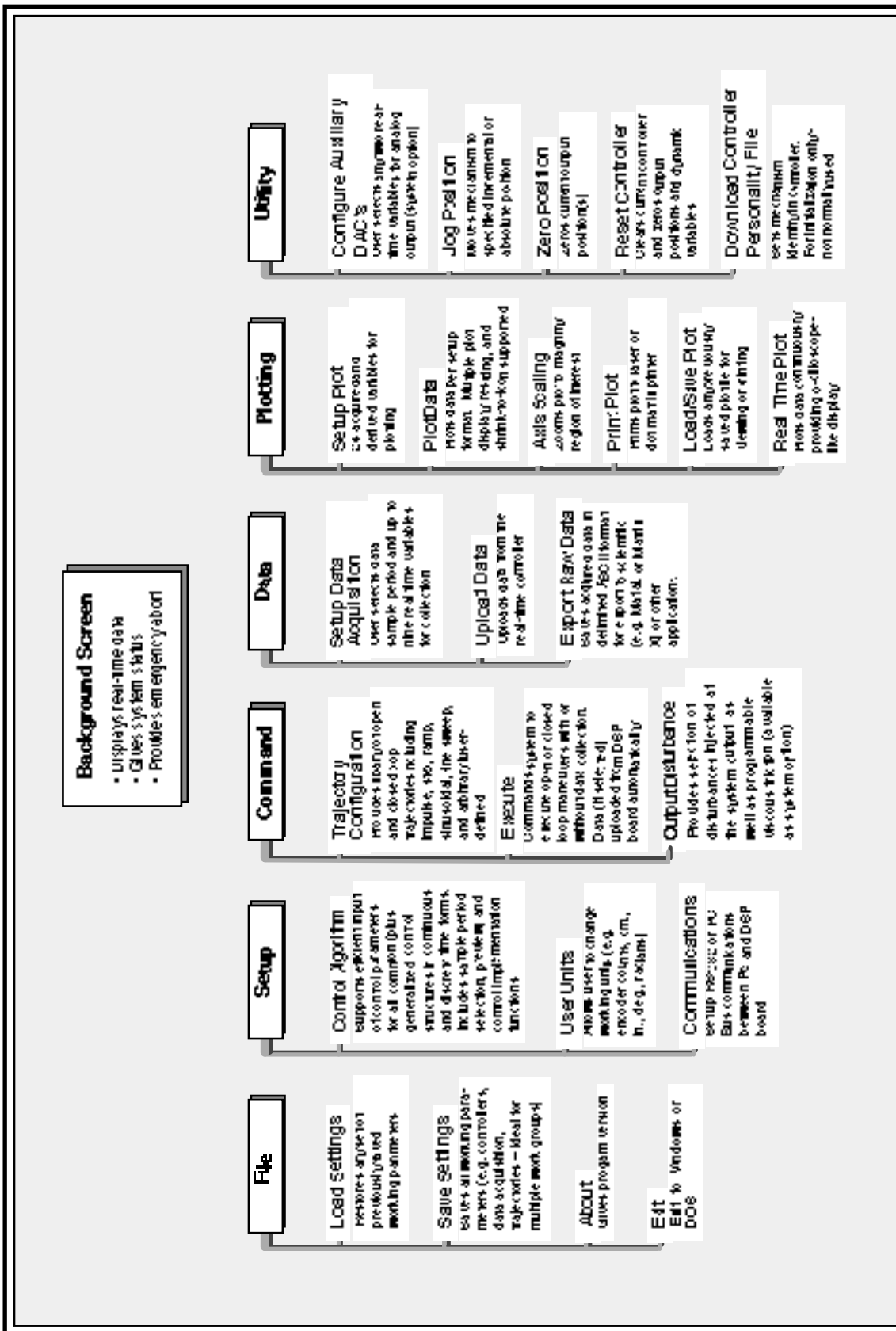


Figure 2.1-1. Structure and Key Features of Executive Program

2.1.1.2 Installation Procedure For The DOS Version

The ECP Executive Program consists of several files on a 3.25" 1.44 megabyte distribution diskette in a compressed form. The key files on the distribution diskette are:

```
ECP.EXE
ECP.DAT
ECPBMP.DAT
*.CFG
*.PLT
*.PMC
```

The "ECP*.*" files are needed to run the Executive Program. The "*.CFG" and "*.PLT" files are some driving function configuration and plotting files that are included for the initial self-guided demonstration. The "*.PMC" file is the controller Personality File and should only be used in the case of a non curable system fault (see Utility Menu below).

To install the Executive program, it is recommended that you make a dedicated sub directory on the hard disk and enter this sub directory. For example type:

```
>MD ECP
>CD ECP
```

Next insert the distribution diskette in either "A:" or "B:" drive, as appropriate. Copy all files in the distribution diskette to the hard disk under the "ECP" sub directory. For example if the "B:" drive is used:

```
>COPY B:*. * C:
```

Next execute INSTALL.EXE by typing:

```
>INSTALL
```

You will notice some file decompression activities. This completes the installation procedure. You may run the ECP Executive by typing:

```
>ECP
```

The Executive program is window based with pull-down menus and dialog boxes. You may either use the cursor keys on the keyboard or a mouse to make selections from the pull-down menus. Vertical movement within these menus is accomplished by the up and down arrow keys, respectively. To make a selection with the keyboard, simply highlight the desired choice and press <ENTER>. Menu choices with highlighted letters may also be selected by pressing the corresponding function key. (The indicated key for menus; "alt" plus the indicated key within dialog boxes).

Within dialog boxes, movement from one object to the next is accomplished by using the <TAB> and the <SHIFT-TAB> keys. Here, "objects" includes input lines, check boxes, and "radio buttons". As you move from one object to the next, the selected object is highlighted. Pressing <ENTER> will effect the function of the highlighted button (e.g. termination of the dialog box will result if the Cancel button is highlighted).

2.1.2 The Windows Version of the Executive Program

2.1.2.1 PC System Requirements

The ECP Executive 16-bit code runs on any PC compatible computer under Windows 3.1x and/or Windows 95. You will need at least 8 megabyte of RAM and a hard disk drive with at least 12 megabytes of space. The 16-bit Windows version of the Executive Program runs best with Pentium® based PC having 16 megabytes or more of memory.

-

2.1.2.2 Installation Procedure For The Windows Version

The ECP Executive Program consists of several files on two 3.25" 1.44 megabyte distribution diskettes in a compressed form. The key files on the distribution diskettes are:

ECP .EXE

ECP .DAT

ECPBMP .DAT

* .CFG

* .PLT

* .PMC

The "ECP*.*" files are needed to run the Executive Program. The "*.CFG" and "*.PLT" files are some driving function configuration and plotting files that are included for the initial self-guided demonstration. The "*.PMC" file is the controller Personality File and should only be used in the case of a non curable system fault (see "Utility Menu" below).

To install the Executive program enter the Windows operating system. Then go to the "Run" menu, and simply run the SETUP.EXE file from diskette labeled 1. Follow the interactive dialog boxes of the installation program until completion.

2.1.3 Background Screen

The *Background Screen*, shown in Figure 2.1.-1, remains in the background during system operation including times when other menus and dialog boxes are active. It contains the main menu and a display of real-time data, system status, and an *Abort Control* button to immediately discontinue control effort in the case of an emergency.

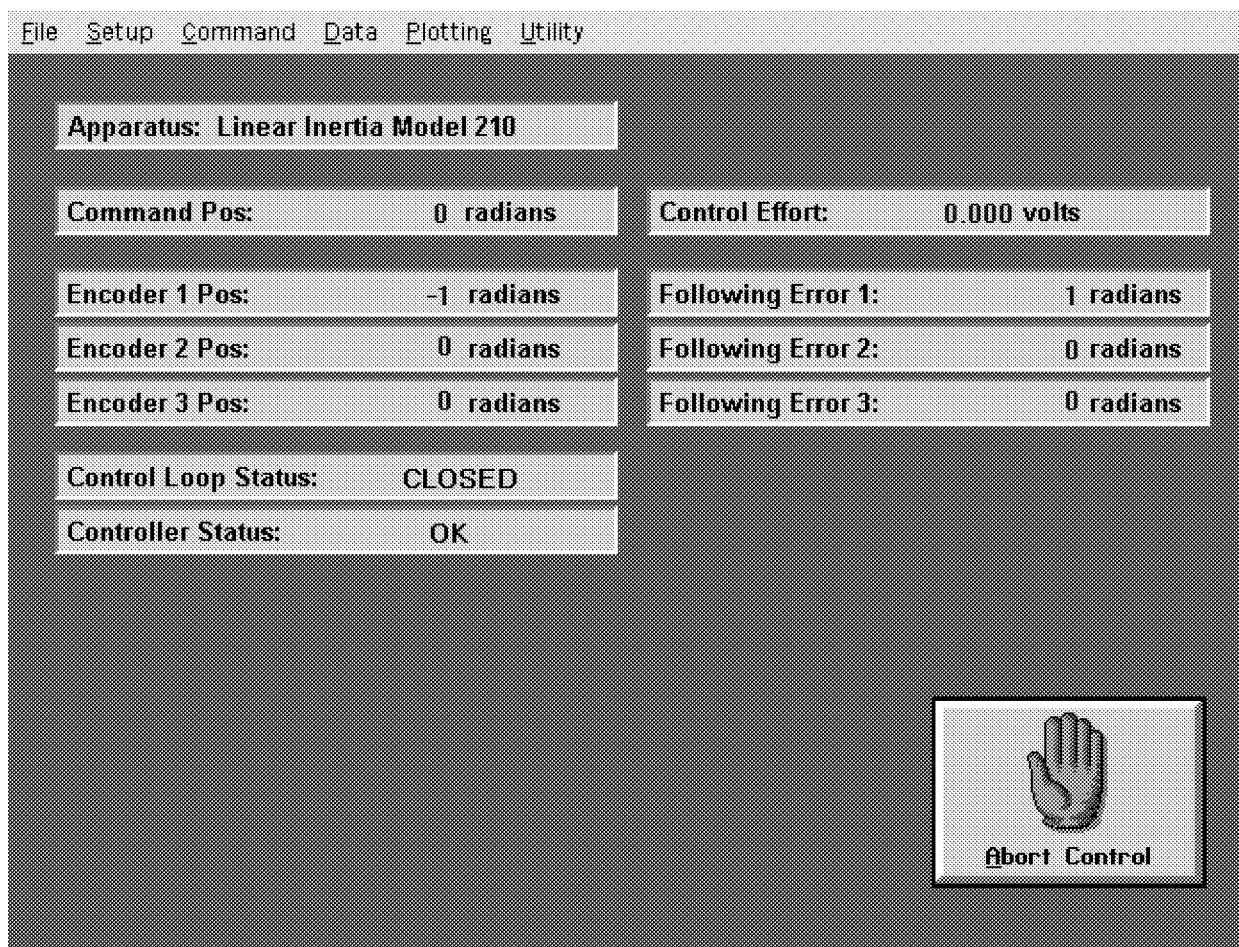


Figure 2.1-1. The Background Screen

2.1.3.1 Real-Time Data Display

In the *Data Display* fields, the instantaneous commanded position, encoder positions, following errors (instantaneous differences between the commanded position and the actual encoder positions), and control effort in volts (on the DAC) are shown.

2.1.3.2 System Status Display

The Control Loop Status ("Open" or "Closed"), indicates "Closed" unless an open loop trajectory is being executed or a "Limit Exceeded" condition has occurred. In either of these cases the Control Loop Status will indicate "Open". The Controller Status field will indicate "Active" unless a motor overspeed, an over-travel (limit switch), or motor/amplifier over-temperature condition has occurred (see Section 2.3 for more details). In any of these cases the Controller Status will indicate "Limit Exceeded". The Limit Exceeded indicator will reoccur unless the user takes one of the two following actions depending on the nature of the over-limit cause. Either a stable controller (one that does not cause limiting conditions) must be implemented via the Control Algorithm box under the Setup menu or an acceptable trajectory must be executed under the Command menu. An "acceptable" trajectory is one that does not overspeed the motor, cause excessive travel of the discrete masses, or result in sustained high current to the motor. The controller must be "re implemented" in order to clear the Limit Exceeded condition – see Section 2.1.5.1.1.

The Disturbance Status field will indicate "Active" when the viscous friction disturbance is invoked and/or when a disturbance force profile is selected during a trajectory execution. It will otherwise indicate "Not Active" unless, due to disturbance motor amplifier over-current or limit switch contact, a "Limit Exceeded" condition develops. In this situation the "Limit Exceeded" indication will continue to appear until a new disturbance force is implemented which does not cause a limit exceeding condition. Note that the disturbance drive is optional

for the Model 210 system.

2.1.3.3 Abort Control Button

Also included on the Background Screen is the Abort Control button. Clicking the mouse on this button simply opens the control loop. This is a very useful feature in various situations including one in which a marginally stable or a noisy closed loop system is detected by the user and he/she wishes to discontinue control action immediately. Note also that control action may always be discontinued immediately by pressing the red "OFF" button on the control box. The latter method should be used in case of an emergency.

2.1.3.4 Main Menu Options

The *Main menu* is displayed at the top of the screen and has the following choices:

- File
- Setup
- Command
- Data
- Plotting
- Utility

2.1.4 File Menu

The File menu contains the following pull-down options:

- Load Settings
- Save Settings
- About
- Exit

2.1.4.1 The Load Settings dialog box allows the user to load a previously saved configuration file into the Executive. A configuration file is any file with a ".cfg" extension which has been previously saved by the user using Save Settings. Any "*.cfg" file can be loaded at any time. The latest loaded "*.cfg" file will overwrite the previous configuration settings in the ECP Executive *but* changes to an existing controller residing in the DSP real-time control card will not take place until the new controller is "implemented" – see Section 2.1.5.1. The configuration files include information on the control algorithm, trajectories, data gathering, and plotting items previously saved. To load a "*.cfg" file simply select the Load Settings command and when the dialog box opens, select the desired file from the appropriate directory.^[4] Note that every time the Executive program is entered, a particular configuration file called "default.cfg" (which the user may customize - see below) is loaded. This file must exist in the same directory as the Executive Program in order for it to be automatically loaded.

2.1.4.2 The Save Settings option allows the user to save the current control algorithm, trajectory, data gathering and plotting parameters for future retrieval via the Load Settings option. To save a "*.cfg" file, select the Save Settings option and save under an appropriately named file (e.g. "pid2.cfg"). By saving the configuration under a file named "default.cfg" the user creates a default configuration file which will be automatically loaded on reentry into the Executive program. You may tailor "default.cfg" to best fit your usage.

2.1.4.3 Selecting About brings up a dialog box with the current version number of the Executive program.

2.1.4.4 The Exit option brings up a verification message. Upon confirming the user's intention, the Executive is exited.

2.1.5 Setup Menu

The Setup menu contains the following pull-down options:

- Control Algorithm
- User Units
- Communications

2.1.5.1 Setup Control Algorithm allows the entry of various control structures and control parameter values to the real-time controller – see Figure 2.1-2. In addition to feedforward which will be described later, the currently available feedback options are:

- PID
- PI With Velocity Feedback
- PID+Notch
- Dynamic Forward Path
- Dynamic Prefilter/Return Path
- State Feedback
- General Form

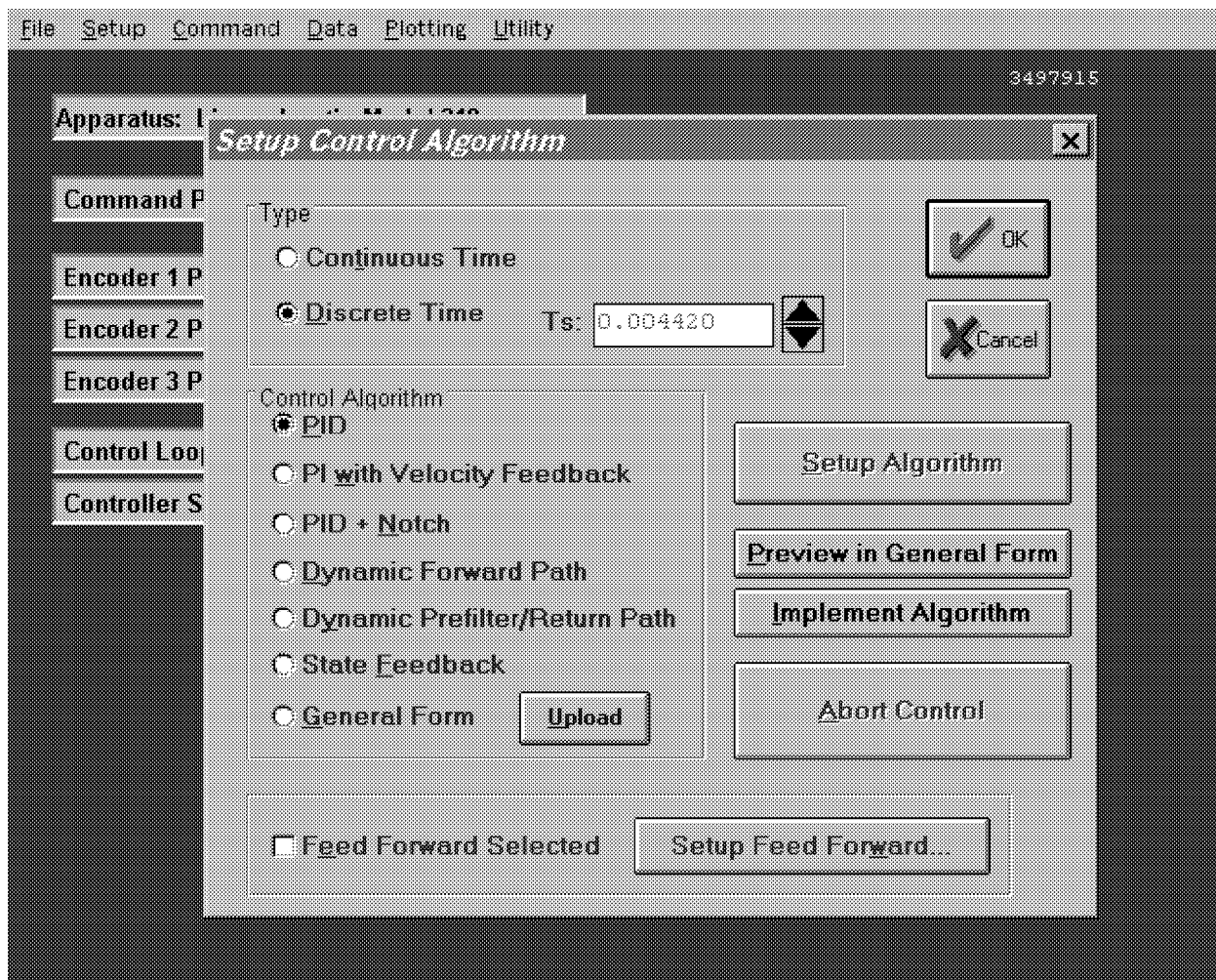


Figure 2.1-2. Setup Control Algorithm Dialog Box

2.1.5.1.1 Discrete Time Control Specification

The user chooses the desired option by selecting the appropriate "radio button" and then clicking on Setup Algorithm. The user must also select the sampling period which is always in multiples of 0.000884 seconds (1.1 KHz is the maximum sampling frequency).^[5] To run the selected choice on the real-time controller click on the Implement Algorithm button. The control action will begin immediately. To stop control action and open the loop with zero control effort click on the Abort button. To upload the current controller select General Form then Upload Algorithm followed by Setup Algorithm. Here you will find the current controller in the form that is actually executed in real-time – see Figure 2.1-3.

Configure Generalized Form Control Algorithm (discrete time)

Block Diagram: $T(z^{-1})$, $S(z^{-1})$, $\frac{1}{R(z^{-1})}$, $H(z^{-1})$, $I(z^{-1})$, $\frac{1}{J(z^{-1})}$, $E(z^{-1})$, $F(z^{-1})$, $\frac{1}{G(z^{-1})}$

Parameters:

T0: 0.9340000	S0: 10.21790	R1: 0.000000000	H0: 1.000000000
T1: 0.000000000	S1: -9.40400	R2: 0.000000000	H1: 0.000000000
T2: 0.000000000	S2: 0.000000000	R3: 0.000000000	I0: 0.000000000
T3: 0.000000000	S3: 0.000000000	R4: 0.000000000	I1: 0.000000000
T4: 0.000000000	S4: 0.000000000	R5: 0.000000000	J1: 0.000000000
T5: 0.000000000	S5: 0.000000000	R6: 0.000000000	E0: 1.000000000
T6: 0.000000000	S6: 0.000000000	R7: 0.000000000	E1: 0.000000000
T7: 0.000000000	S7: 0.000000000		F0: 18.95650000
			F1: -18.8360000
			G1: 0.000000000

Buttons: OK, Cancel, Import

Loop 1: Encoder 1, Encoder 2, Encoder 3

Loop 2: Encoder 1, Encoder 2, Encoder 3

Loop 3: Encoder 1, Encoder 2, Encoder 3

Figure 2.1-3. Dialog Box For Generalized Control Algorithm Input

A typical sequence of events is as follows: Select the desired servo loop closure sampling time T_s in multiples of 0.000884 seconds; then select the control structure you wish to implement (e.g. radio buttons for PID, PID+Notch etc.). Select Setup Algorithm to input the gain parameters (coefficients). You must also select the desired feedback channel by choosing the correct encoder(s) used for your particular control design. Exit Setup by selecting OK. Now you should be back into the Setup Control Algorithm dialog box with a selected set of gains for a specified control structure. To download this set of control parameters to the real-time controller click on Implement Algorithm. This action results in an immediate running of your selected control structure on the real-time controller. If you notice unacceptable behavior (instability and/or excessive ringing or noise) simply click on Abort Control which opens up the control loop with zero control effort commanded to the actuator.

To inspect the form by which your particular control structure is actually implemented on the real-time controller, simply click on Preview In General Form. You may edit the algorithm in the General Form box, however when you exit, you must select General Form prior to "implementing" if you want the changes to become effective. (i.e. the radio button will still indicate the box you were in prior to previewing and this one will be downloaded unless General Form is selected).

The Setup Feed Forward option allows the user to add feedforward action to any of the above feedback structures. By clicking on this button a dialog box appears which allows the feedforward control parameters (coefficients) to be entered. To augment the feedforward action to the feedback algorithm the user must then check the Feedforward Selected check-box. Any subsequent downloading (via the Implement Algorithm button) combines the feedforward control algorithm with the selected feedback control algorithm.

Important Note: Every time a set of control coefficients are downloaded via Implement Algorithm button, the commanded position as well as all of the encoder positions are reset to zero. This action is taken in order to prevent any instantaneous unwanted transient behavior from the controller. The control action then begins immediately.

Important Note: For high order control laws (those using more than 2 or 3 terms of either the R, S, T, K, or L polynomials), it is often important that the coefficients be entered with relatively high precision– say at least 5 to 6 points after the decimal. The real-time controller works with 96-bit real number arithmetic (48-bit integer plus 48-bit fraction). Although the Executive displays the coefficients with nine points after the decimal, it accepts higher precision numbers and downloads them correctly.

2.1.5.1.1 Continuous Time Control Specification

Depending on your course of study, It may be desirable to specify the control algorithm in continuous time form.^[6] The method for inputting control parameters is identical to that described for the discrete time case. Again you may preview your controller in the continuous General Form prior to implementing. Upon selecting either Implement Algorithm or Preview in General Form, the algorithm also gets mapped into the discrete General Form where it may be viewed either before (following "Preview") or after (following "Implement") downloading to the real time controller.^[7]

Again it is the discrete time general form that is actually executed in real time. The input coefficients are transformed to discrete time using one of the two following substitutions. For polynomials: $n(s)$, $d(s)$ in PID + Notch; $s(s)$, $t(s)$, and $r(s)$ in Dynamic Forward Path, Dynamic Prefilter / Return Path, and the General Form; and $k(s)$, $l(s)$ in Feed Forward, the Tustin (bilinear) transform

$$s = \frac{2}{T_s} \frac{1-z^{-1}}{1+z^{-1}}$$

is used. All other cases (first order) use the Backwards Difference method:

$$s = \frac{1-z^{-1}}{T_s}$$

Blocks using the Tustin transform must be proper in s while those using backwards difference may be improper – e.g. a differentiator.^[8]

2.1.5.1.3 Importing Controller Specifications From Other Applications

You may import controllers designed using other applications such as Matlab[®] and Matrix X[®].^[9] Within each controller specification dialog box is an Import button by which the user download the control gains or coefficients previously saved as an ASCII text file with a extension “*.par”. The format for the file is as shown in Table 2.1-1.

Table 2.1-1 File Format For Importing Controller Coefficients

Continuous Time Controller Specification		Discrete Time Controller Specification		Continuous Time Controller Specification		Discrete Time Controller Specification	
Control Algorithm	File Format	Control Algorithm	File Format	Control Algorithm	File Format	Control Algorithm	File Format
PID	[PID_C] kp=n.n kd=n.n ki=n.n	Dynamic Prefilter/Return Path	[DYNPR_C] t0=n.n : t7=n.n s0=n.n : s7=n.n r0=n.n : r7=n.n	PID	[PID_D] Kp=n.n Kd=n.n Ki=n.n	Dynamic Prefilter/Return Path	[DYNPR_D] T0=n.n : T7=n.n S0=n.n : S7=n.n R1=n.n : R7=n.n
PID w/ Velocity Feedback	[PID_C] kp=n.n kd=n.n ki=n.n	State Feedback	[STATEF_C] kpf=n.n k1=n.n k2=n.n k3=n.n k4=n.n k5=n.n k6=n.n	PID w/ Velocity Feedback	[PID_D] Kp=n.n Kd=n.n Ki=n.n	State Feedback	[STATEF_D] Kpf=n.n K1=n.n K2=n.n K3=n.n K4=n.n K5=n.n K6=n.n
PID + Notch	[PIDNOTCH_C] kp=n.n kd=n.n ki=n.n n0=n.n n1=n.n n2=n.n FIX ON TORS d0=n.n : d4=n.n	General Form	[GENERAL_C] t0=n.n : t7=n.n s0=n.n : s7=n.n r0=n.n : r7=n.n h0=n.n h1=n.n i0=n.n i1=n.n j0=n.n j1=n.n e0=n.n e1=n.n f0=n.n f1=n.n g0=n.n g1=n.n	PID + Notch	[PIDNOTCH_D] Kp=n.n Kd=n.n Ki=n.n N0=n.n : N4=n.n D1=n.n : D4=n.n	General Form	[GENERAL_D] T0=n.n : T7=n.n S0=n.n : S7=n.n R1=n.n : R7=n.n H0=n.n H1=n.n I0=n.n I1=n.n J1=n.n E0=n.n E1=n.n F0=n.n F1=n.n G1=n.n
Dynamic Forward Path	[DYNFWD_C] s0=n.n : s7=n.n	Feed Forward	[FF_C] k0=n.n : k6=n.n	Dynamic Forward Path	[DYNFWD_D] S0=n.n : S7=n.n	Feed Forward	[FF_D] K0=n.n : K6=n.n

r0=n.n ⋮ r7=n.n	l0=n.n ⋮ l7=n.n	R1=n.n ⋮ R7=n.n	L1=n.n ⋮ L7=n.n
-----------------------	-----------------------	-----------------------	-----------------------

2.1.5.2 The User Units dialog box provides the user with various choices of angular or linear units. For Model 210 the choices are *counts*, *centimeters*, *millimeters*, and *inches*. There are 1604 counts, per centimeter travel of the mass carriages. By clicking on the desired radio button the units are changed automatically for trajectory inputs as well as the Background Screen displays, plotting and jogging activities. Units of *counts* are used exclusively for the examples in this manual.

2.1.5.3 The Communications dialog box is usually used only at the time of installation of the real-time controller. The choices are serial communication (RS232 mode) or PC-bus mode – see Figure 2.1-4. If your system was ordered for PC-bus mode of communication, you do not usually need to enter this dialog box unless the default address at 528 on the ISA bus is conflicting with your PC hardware. In such a case consult the factory for changing the appropriate jumpers on the controller. If your system was ordered for serial communication the default baud rate is set at 34800 bits/sec. To change the baud rate consult factory for changing the appropriate jumpers on the controller. You may use the Test Communication button to check data exchange between the PC and the real-time controller. This should be done after the correct choice of Communication Port has been made. The Timeout should be set as follows:

ECP Executive For Windows with Pentium Computer:	Timeout \geq 50,000
ECP Executive For Windows with 486 Computer:	Timeout \geq 20,000
ECP Executive For DOS with Pentium Computer:	Timeout \geq 150
ECP Executive For DOS with 486 or lower Computer:	Timeout \geq 80

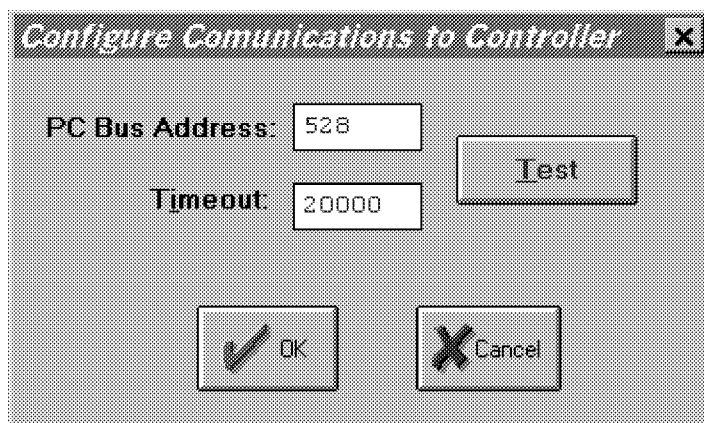


Figure 2.1-4. The Communications Dialog Box

2.1.6 Command Menu

The Command menu contains the following pull-down options

- Trajectory . . .
- Disturbance . . .
- Execute . . .

2.1.6.1 The Trajectory Configuration dialog box (see Figure 2.1.-5) provides a selection of trajectories through which the apparatus can be maneuvered. These are:

- Impulse
- Step
- Ramp
- Parabolic
- Cubic
- Sinusoidal
- Sine Sweep
- User Defined

A mathematical description of these is given later in Section 4.1.

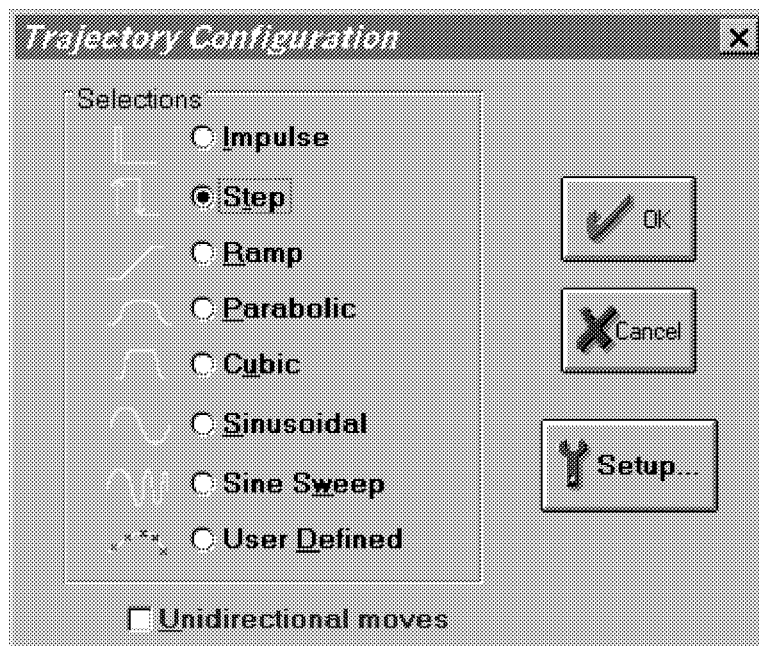


Figure 2.1-5. The Trajectory Configuration Dialog Box

All *geometric* input shapes – Impulse through Cubic – may be specified as Unidirectional or Bi-directional. Examples of these shape types are shown in Figure 2.1-6. The bi-directional option should normally be selected whenever the system is configured to have a rigid body mode (one that rotates freely) and the system is operating open loop. This is to avoid excessive speed or displacement of the system.

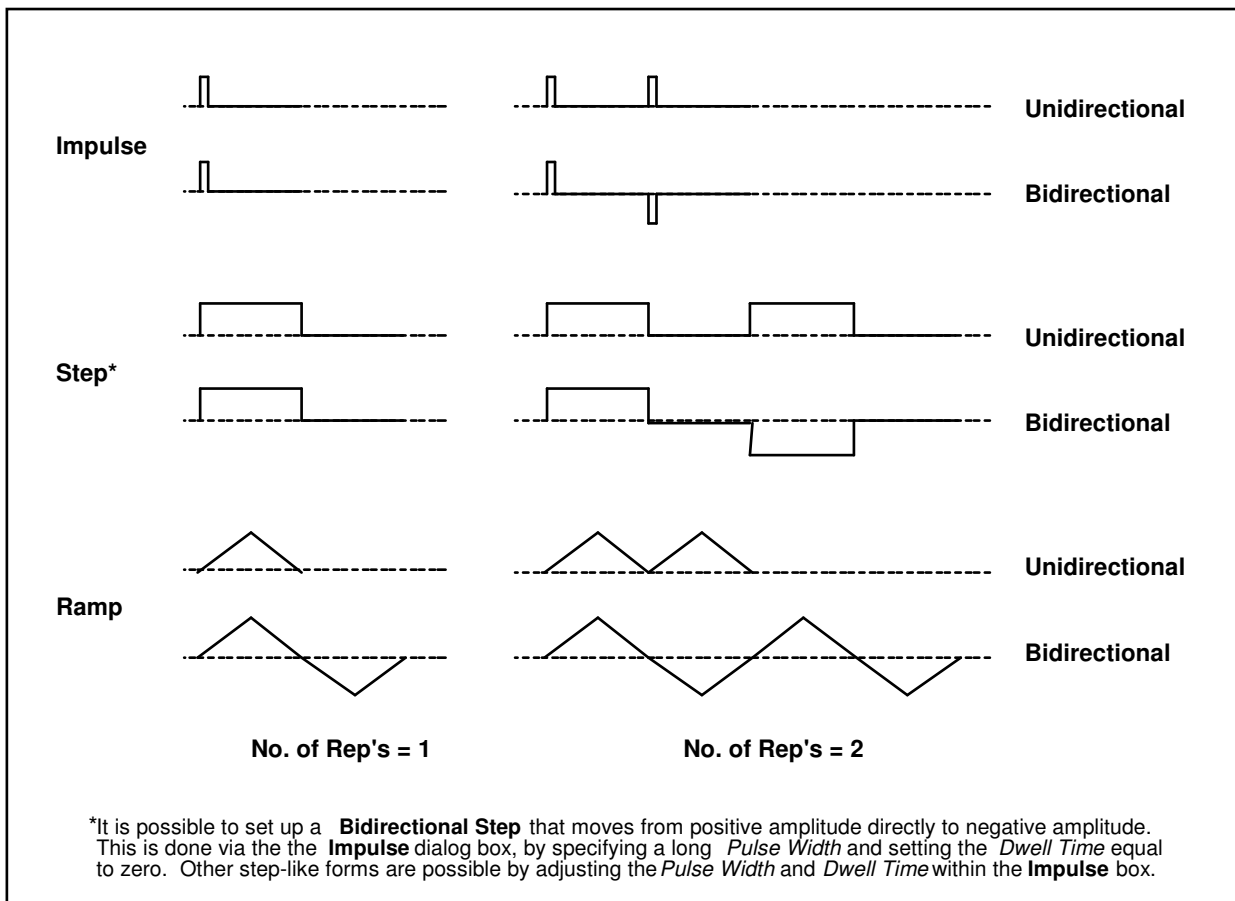


Figure 2.1-6. Example Geometric Trajectories

By selecting the desired shape followed by Setup, one enters a dialog box for the corresponding trajectory. Examples of these boxes are shown in Figure 2.1-7. The amplitude is specified in units consistent with the selected User Units (Setup menu) under closed loop operation and in units of DAC volts (0-5 VDC) under open loop. The closed loop units will change automatically to be consistent with the selected User Units. Amplitudes are always incremental from the value that exists at the beginning of the maneuver (see Execute, Section 2.1.6.3). The characteristic durations of the various shapes are specified in units of milliseconds.

The Impulse, Step, Sinusoidal, Sine Sweep, and User-defined trajectories may be specified as open or closed loop. The remaining shapes are closed loop only.

Important Note: It is possible to specify amplitudes and/or abruptly changing shapes that exceed the linear range of the motor and drive electronics or cause large excursions of the mechanism due to system dynamic response. These may result in inaccurate test results and could lead to a hazardous operating condition or overstressing of the apparatus^[10]. If in doubt as to whether the drive linear range has been exceeded, you may view Control Effort (either by real-time plotting or via data acquisition/plotting^[11]). When specifying an unfamiliar shape the user should generally begin with small amplitudes, velocities, accelerations, and RMS power levels and gradually increase them to suitable safe values. Similarly, when specifying driving function parameters, one should begin with conservatively low values; then gradually increase them. See Section 2.3 on safety.

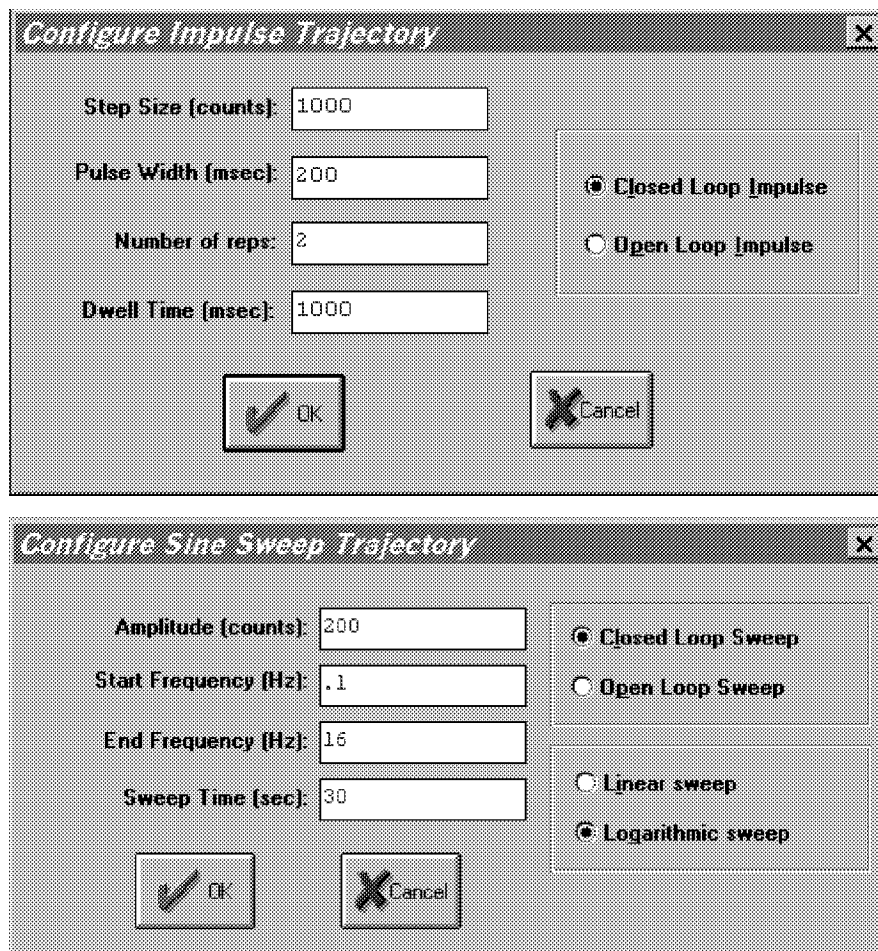


Figure 2.1-7. Example "Setup Trajectory" Dialog Boxes

The Impulse dialog box provides for specification of amplitude, impulse duration, dwell duration, and number of repetitions.^[12] The Step box supports specification of step amplitude, duration, and number of repetitions with the dwell duration being equal to the step duration. The Ramp shape is specified by the peak amplitude, ramp slope (units of amplitude per second), dwell time at amplitude peaks, and number of repetitions. The Parabolic shape is specified by the peak amplitude, ramp slope (units of ampl./s), acceleration time, dwell time at amplitude peaks, and number of repetitions. In this case, the acceleration (units of ampl./s^2) results from meeting the specified amplitude, slope, and acceleration period. The Cubic shape is specified by the peak amplitude, ramp slope (units of ampl./s), acceleration time, dwell time at amplitude peaks, and number of repetitions. In this case, the "jerk" (units of ampl./s^3) results from meeting the specified amplitude, slope, and acceleration period where the acceleration increases linearly in time until the specified velocity is reached.

Note that the only difference between a parabolic input and a cubic one is that during the acceleration/deceleration times, a constant acceleration is commanded in a parabolic input and a constant jerk is commanded in the cubic input. Of course, in a ramp input the commanded acceleration/deceleration is infinite at the ends of a commanded displacement stroke and zero at all other times during the motion. For safety, there is an apparatus-specific limit beyond which the Executive program will not accept the amplitude inputs for each geometric shape.

The Sinusoidal dialog box provides for specification of input amplitude, frequency and number of repetitions.

The Sine Sweep dialog box accepts inputs of amplitude, start and end frequencies (units of Hz), and sweep duration^[13]. Both linear and logarithmic frequency sweeps are available. The linear sweep frequency increase

is linear in time. For example a sweep from 0 Hz to 10 Hz in 10 seconds results in a one Hertz per second frequency increase. The logarithmic sweep increases frequency logarithmically so that the time taken in sweeping from 1 to 2 Hz for example, is the same as that for 10 to 20 Hz when a single test run includes these frequencies. There is an apparatus-specific amplitude limit beyond which the Executive will not accept the inputs.

Important Note: A large open loop amplitude combined with a low frequency may result in an over-speed condition which will be detected by the real-time controller and will cause the system to shut down. In closed loop operations, high frequency, large amplitude tests may result in a shut down condition. For both the open and closed loop cases, even modest commanded amplitude near or at a resonance frequency can cause an excessive shaft deflection. Any of these conditions will cause the test to be aborted and the System Status display in the Background Screen to indicate Limit Exceeded. To run the test again you should reduce the input shape amplitude and then Reset Controller (Utility menu), and re-Implement a stabilizing controller (Command menu). In general, all trajectories which generate either too high a speed, too large a deflection, or excessive motor power will cause this condition – see the safety section 2.3. For a further margin of safety, there is an apparatus-specific amplitude limit beyond which the Executive program will not accept the inputs.

The User Defined shape dialog box provides an interface for the specification of any input shape created by the user. In order to make use of this feature the user must first create an ASCII text file with an extension ".trj" (e.g. "random.trj"). This file may be accessed from any directory or disk drive using the usual file path designators in the filename field or via the Browse button. If the file exists in the same directory as the Executive program, then only the file name should be entered. The content of this file should be as follows:

The first line should provide the number of points specified. The maximum number of points is 923. This line should not contain any other information. The subsequent lines (up to 923) should contain the consecutive set points. For example to input twenty points equally spaced in distance one can create a file called "example.trj" using any text editor as follows

```
20
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
```

95

100

The segment time which is a time between each consecutive point can be changed in the dialog box. For example if a 100 milliseconds segment time is selected, the above trajectory shape would take 2 seconds to complete ($100 * 20 = 2000$ ms). The minimum segment time is restricted to five milliseconds by the real-time controller. When Open Loop is selected, the units of the trajectory are assumed to be DAC bits ($+16383 = 4.88$ V, $+16383 = -4.88$ V). In Closed Loop mode, the units are assumed to be the position displacement units specified under User Units (Setup menu). The shape may be treated by the system as a discrete function exactly as specified, or may be smoothed by checking the Treat Data As Splined box. In the latter case the shapes are cubic spline fitted between consecutive points by the real-time controller. Obviously a user-defined shape may also cause over-speed or over-deflection of the mechanism if the segment time is too long or the distance between the consecutive points is too great.

2.1.6.2 The Disturbance Configuration dialog box (see Figure 2.1.-6) provides a selection of disturbance force profiles for the disturbance motor. This motor and servo drive is optional on the Model 210 and 210a systems. If your system does not include this option, this dialog box will not be accessible. The available disturbances are:

- Viscous Friction
- Step
- Sinusoidal (time)
- Sinusoidal (theta)
- User Defined

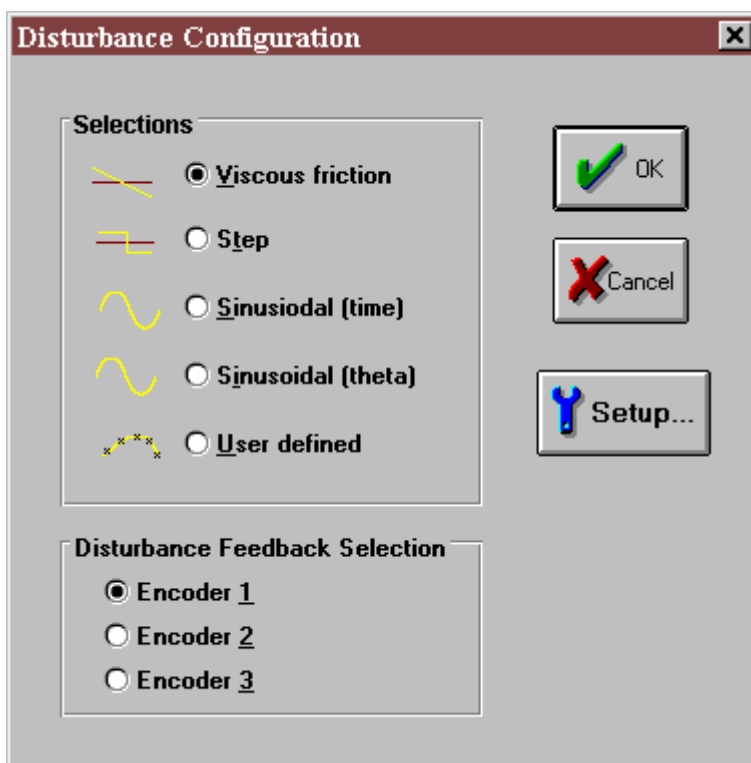


Figure 2.1-6. The Disturbance Configuration Dialog Box

By selecting the desired disturbance profile followed by Setup, one enters a corresponding dialog box.

The Viscous Friction window allows the user to input a disturbance signal proportional to the speed of the selected mass carriage as sensed by the its encoder. The Amplitude entry is the magnitude of the viscous coefficient in units of volts/meter/second. Once the disturbance motor is calibrated this entry translates to a certain number of N-m/rad./sec. The user has a choice of implementing the viscous disturbance either directly through this dialog box or later prior to running a trajectory in the Execute dialog box. In addition, under the Disturbance Feedback Selection dialog box, the user must match the actual physical location of the Disturbance motor with the corresponding Encoder number.

The Step disturbance dialog box allows the user to input the parameters for a square wave force disturbance. The entries in this dialog box are identical to the Open Loop Step trajectory discussed above.

The Sinusoidal (time) option allows the user to input a disturbance to the desired mass carriage via the disturbance motor in the form of a sinusoidal function of time. The entries in this dialog box are identical to the Open Loop Sinusoidal trajectory discussed above.

The Sinusoidal (theta) option specifies a sinusoidal force as a function of carriage position. This allows the simulation of spatially dependent disturbances such as drive motor cogging torque. The amplitude of the disturbance force is entered in terms of volts. The user must enter the number of force cycles revolution of the corresponding mass carriage encoder. The conversion into cycles per length of mass travel is: 1 encoder revolution = 7.06 cm (=16,000 encoder counts)^[14]. The period of time for which the disturbance is required to be active must be specified. It is important to properly specify the encoder (mass carriage) number corresponding to the current disturbance drive location in the Disturbance Configuration dialog box for the Sinusoidal (theta) function to work properly

The User Defined disturbance box provides the interface for the input of any form of disturbance trajectory created by the user. In order to make use of this feature the user must first create an ASCII text file with an extension `.trj` (e.g. `random.trj`). The format is identical to the User Defined trajectories discussed in the pervious section. Note that the maximum number of points are still 100 and the first entry must be the number of points in a particular file. The units of inputs are in DAC bits (+16383 = 4.88 V, +16383 = -4.88 V). After the calibration of the disturbance motor, the exact ratio between a DAC bit and the actual disturbance force on the mass carriage may be determined.

During the active period of any of the above disturbance profiles the Disturbance Status will normally indicate "Active". It is, however, possible that the disturbance motor enters the "Limit Exceeded" condition either as a result of over current or over speed. To return to the "Active" condition, the user must modify the disturbance parameters and implement the disturbance force again via the Execute dialog box (note that the Viscous friction disturbance may also be implemented within its own dialog box).

The following rules apply to disturbance implementation:

1. You must have selected a disturbance (and verified its parameters) under the Disturbance Configuration dialog box and checked "Include XXX Disturbance" when Executing a trajectory.
2. The disturbance will only be active while the trajectory is executing. If the trajectory terminated before the specified disturbance duration, the disturbance will also terminate. (You may of course input a trajectory of zero amplitude to study the effects of the disturbance alone)
3. The only exception to rules 1 and 2 is viscous damping which may be invoked either via its own dialog box (under Disturbance Configuration) or when a trajectory is executed (by checking Include Viscous Friction before Executing). Viscous friction may run simultaneously with other disturbances. Note that Viscous Friction, if implemented, will remain in effect until either it is removed in its own dialog box, or a new trajectory is run without checking Include Viscous Friction.
4. Disturbance control effort is limited to $\pm 4.88V$.
5. For the Viscous Friction and the Sinusoidal (theta) options, the checked Encoder number under the Disturbance Feedback Selection must match the physical location of the Disturbance motor

2.1.6.3 The Execute dialog box (see Figure 2.1-7) is entered after a trajectory is selected. Here the user commands the system to execute the current specified trajectory and may also choose viscous friction and an output disturbance (system option). The user may select either Normal or Extended Data Sampling. Normal Data Sampling acquires data for the duration of the executed trajectory. Extended Data Sampling acquires data for an additional 5 seconds beyond the end of the maneuver. Both the Normal and Extended boxes must be checked to allow extended data sampling. (For the details of data gathering see Section 2.1.7.1 Setup Data Acquisition).

After selecting disturbance and data gathering options, the user normally selects Run. The real-time controller will begin execution of the specified trajectory. Once finished, and provided the Sample Data box was checked, the data will be uploaded from the DSP board into the Executive (PC memory) for plotting, saving and exporting. At any time during the execution of the trajectory or during the uploading of data the process may be terminated by clicking on the Abort button. Finally, if the disturbance force profile has a time period longer than the selected trajectory period, it will be terminated at the end of the trajectory profile.

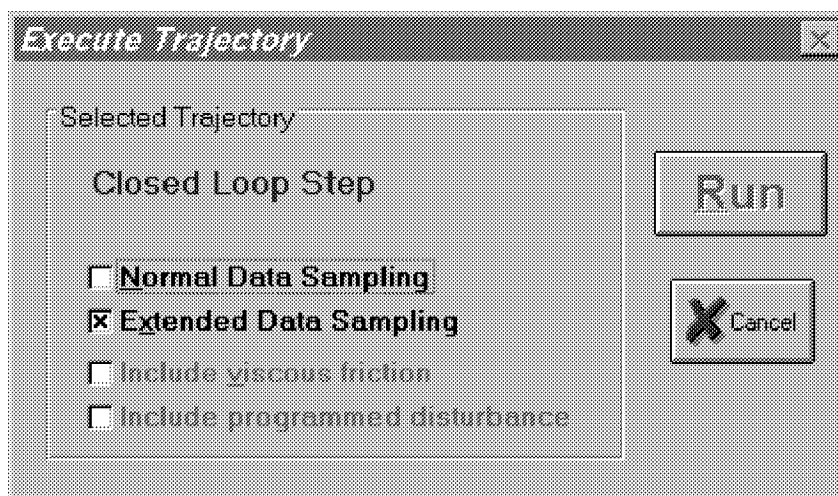


Figure 2.1-7. The Execute Dialog Box

2.1.7 Data Menu

The Data menu contains the following pull-down options

Setup Data Acquisition

Upload Data

Export Raw Data

2.1.7.1 Setup Data Acquisition allows the user to select one or more of the following data items to be collected at a chosen multiple of the servo loop closure sampling period while running any of the trajectories mentioned above – see Figures 2.1-8 and 4.1-1:

Commanded Position

Encoder 1 Position

Encoder 2 Position

Encoder 3 Position

Control Effort (output of the DAC to the servo amplifier (volts))

Disturbance Effort (disturbance motor command [system option])

Node A (input to the H polynomial in the Generalized Control Algorithm)

Node B (input to the E polynomial in the Generalized Control Algorithm)

Node C (output of the 1/G polynomial in the Generalized Control Algorithm)

Node D (output of the feedforward controller which is added to the node C value to form the combined regulatory and tracking controller).

In this dialog box the user adds or deletes any of the above items by first selecting the item, then clicking on the Add Item or Delete Item button. The user must also select the data gather sampling period in multiples of the servo period. For example, if the sample time (T_s in the Setup Control Algorithm) is 0.00442 seconds and you choose 5 for your gather period here, then the selected data will be gathered once every fifth sample or once every 0.0221 seconds. Usually for trajectories with high frequency content (e.g. Step, or high frequency Sine Sweep), one should choose a low data gather period (say 10 ms). On the other hand, one should avoid gathering more often (or more data types) than needed since the upload and plotting routines become slower as the data size increases. The maximum available data size (no. variables x no. samples) is 33,586.

2.1.7.2 Selecting Upload Data allows any previously gathered data to be uploaded into the Executive. This feature is useful when one wishes to switch and compare between plotting previously saved raw data and the currently gathered data. Remember that the data is automatically uploaded into the executive whenever a trajectory is executed and data acquisition is enabled. However, once a previously saved plot file is loaded into the Executive, the currently gathered data is overwritten. The Upload Data feature allows the user to bring the overwritten data back from the real-time controller into the Executive.

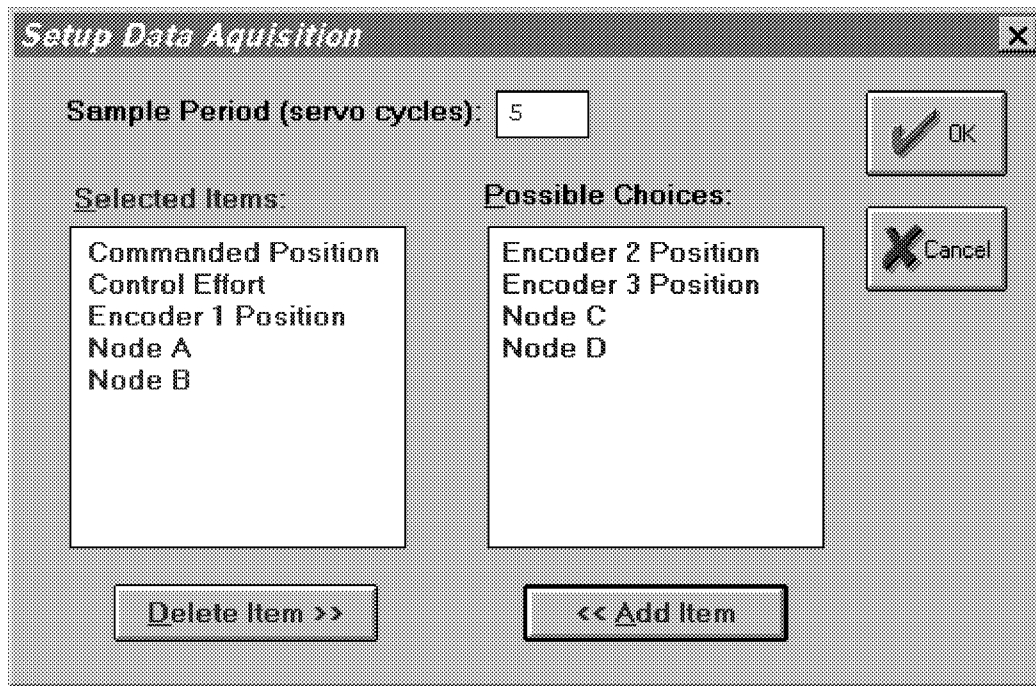


Figure 2.1-8. The Setup Data Acquisition Dialog Box

2.1.7.3 The Export Raw Data function allows the user to save the currently acquired data in a text file in a format suitable for reviewing, editing, or exporting to other engineering/scientific packages such as Matlab[®].^[15] The first line is a text header labeling the columns followed by bracketed rows of data items gathered. The user may choose the file name with a default extension of ".txt" (e.g. lqrstep.txt). The first column in the file is sample number, the next is time, and the remaining ones are the acquired variable values. Any text editor may be used to view and/or edit this file.

2.1.8 Plotting Menu

The Plotting menu contains the following pull-down options

- Setup Plot
- Plot Data
- Axis Scaling
- Print Plot
- Load Plot Data
- Save Plot Data
- Real Time Plotting
- Close Window

2.1.8.1 The Setup Plot dialog box (see Figure 2.1-9) allows up to four acquired data items to be plotted simultaneously – two items using the left vertical axis and two using the right vertical axis units. In addition to the acquired raw data, you will see in the box plotting selections of velocity and acceleration for the position and input variables acquired. These are automatically generated by numerical differentiation of the data during the plotting process. Simply click on the item you wish to add to the left or the right axis and then click on the Add to Left Axis or Add to Right Axis buttons. You must select at least one item for the left axis before plotting is allowed – e.g. if only one item is plotted, it must be on the left axis. You may also change the plot title from the default one in this dialog box.

Items for comparison should appear on the same axis (e.g. commanded vs. encoder position) to ensure the same axis scaling and bias. Items of dissimilar scaling or bias (e.g. control effort in volts and position in counts) should be placed on different axes.

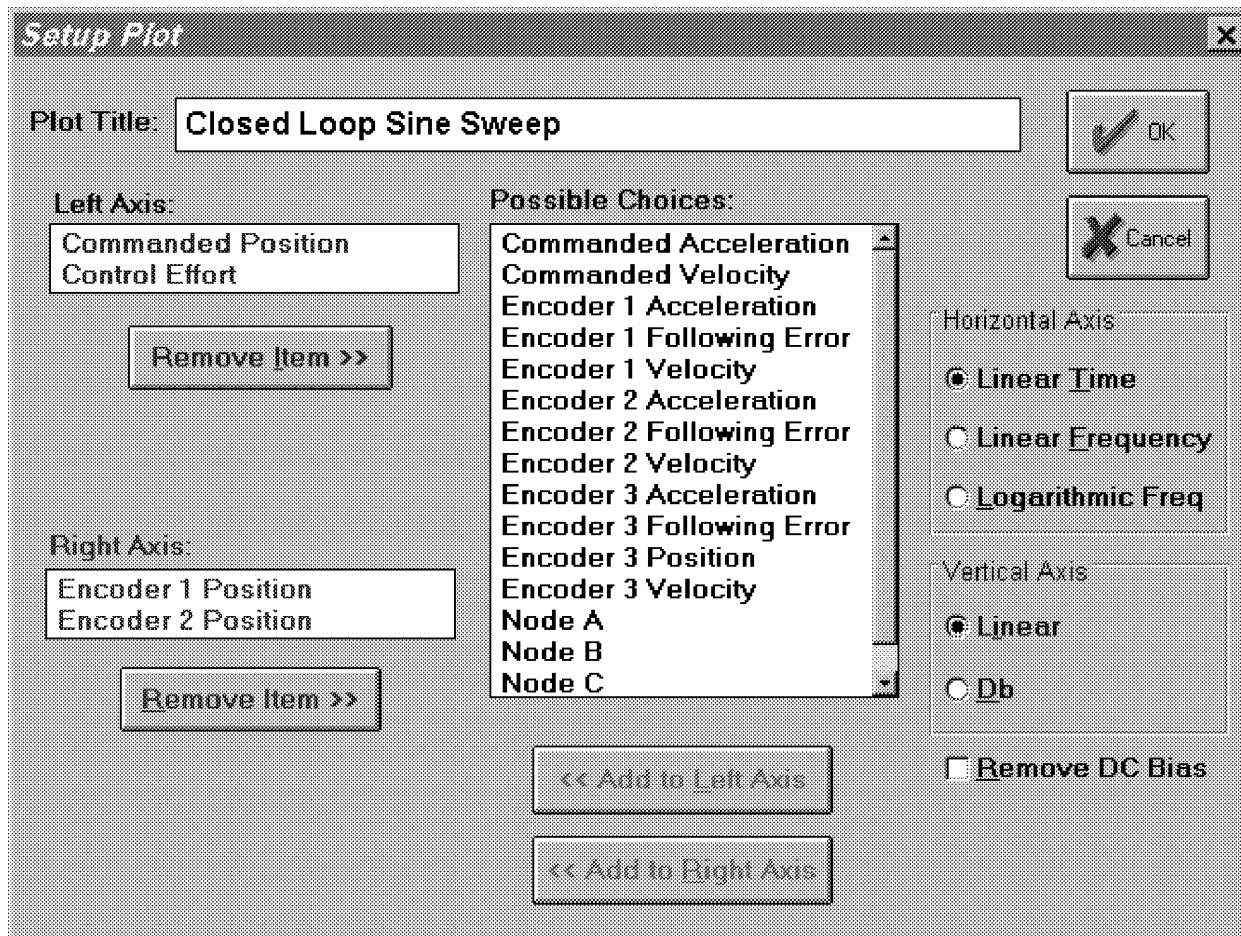


Figure 2.1-9 The Setup Plot Dialog Box

When the current data (either from the last test run or from a previously saved and loaded plot file) is from a Sine Sweep input, several data scaling/transformation options appear in the Setup Plot box. These include the presentation of data with horizontal coordinates of time, linear frequency (i.e. the frequency of the input) or logarithmic frequency. The vertical axis may be plotted in linear or Db (i.e. $20 \cdot \log_{10}(\text{data})$) scaling. In addition, the Remove DC Bias option subtracts the average of the final 50 data points from the data set of each acquired variable. This generally gives a more representative view of the frequency response of the system, particularly when plotting low amplitude data in Db. Sine Sweep must be selected in the Trajectory Configuration dialog box in order for these options to be available in Setup Plot. Examples of sine sweep (frequency response) data plotted using two of these options is given later in Figure 3.2-6.

2.1.8.2 Plot Data generates a plot of the selected items. The plots follow the usual Windows™ format for resizing, exiting, etc. By clicking on the left button on the upper right hand corner of the plot, it is minimized to an icon. It can be expanded back to the full size at any time by double-clicking on the icon. Also, more than one plot may be tiled on the Background Screen. This function is very useful for comparing several graphs. By clicking on any point within the area of a desired plot it will appear over the others. Plots may be arbitrarily shaped by using the cursor to "drag" the edges of the plot. The corners allow you to resize height and width simultaneously (position cursor at corner and begin "dragging" when cursor becomes a double arrow). A typical plot as seen on screen is shown in Figure 2.1-10.



Figure 2.1-10. A Typical Plot Window

2.1.8.3 Axis Scaling provides for scaling or “zooming” of the horizontal and vertical axes for closer data inspection – both visually and for printing. This box also provides for selection or deselection of grid lines and data point labels. When Real-time Plotting is used (see Section 2.1.8.7), the data sweep / refresh speed and amplitudes may be adjusted via the Axis Scaling box.

2.1.8.4 The Print Data option provides for printing a hard copy of the selected plot on the current PC system printer. The plots may be resized prior to printing to achieve the desired print format

2.1.8.5 The Load Plot Data dialog box enables the user to bring into the Executive previously saved “.plt” plot files. Note that such files are not stored in a format suitable for use by other programs. The “.plt” plot files contain the sampling period of the previously saved data. As a result, after plotting any previously saved plot files and before running a trajectory, you should check the servo loop sampling period T_s in the Setup Control Algorithm dialog box. If this number has been changed, then correct it. Also, check the data gathering sampling period in the Data Acquisition dialog box, this too may be different and need correction.

2.1.8.6 The Save Plot Data dialog box enables the user to save the data gathered by the controller for later plotting via Load Plot Data. The default extension is “.plt” under the current directory. Note that “.plt” files are not saved in a format suitable for use by other programs. For this purpose the user should use the Export Raw Data option of the Data menu.

2.1.8.7 The Setup Real Time Plotting dialog box enables the user to view data in real time as it is being generated by the system. Thus the data is seen in an oscilloscope-like fashion. Unlike normal (off-line) plotting, real-time plotting occurs continuously whether or not a particular maneuver (via Execute, Command Menu) is being executed^[16]. The setup for real-time plotting is essentially identical to that for normal plotting (see Section 2.1.8.1). Because the expected data amplitude is not known to the plotting routine, the plot will

first appear with the vertical axes scaled to full scale values of 1000 of the selected variable units. These should be rescaled to appropriate values via Axis Scaling. The sweep or data refresh rate may also be changed via Axis Scaling when real-time plotting is underway. A slow sweep rate is suitable for slow system motion or when a long data record is to be viewed in a single sweep. The converse generally holds for a fast sweep rate.

The data update rate is approximately 50 ms and is limited by the PC/DSP board communication rate. Therefore, frequency content above about 5 Hz is not accurately displayed due to numerical aliasing. The real-time display however is very useful in visually correlating physical system motion with the plotted data and is valid for most practical system frequencies. The data acquired via the data acquisition hardware (for normal plotting) may be sampled at much higher rates (up to 1.1 KHz) and hence should be used when quantitative high speed measurements are desired.

2.1.9 Utility Menu

The Utility menu contains the following pull-down options:

- Configure Auxiliary DACs
- Jog Position
- Zero Position
- Reset Controller
- Rephase Motor
- Download Controller Personality File

2.1.9.1 The Configure Auxiliary DACs (system option) dialog box (see Figure 2.1-10) enables the user to select various items for analog output on the two analog channels (optional) in front of the ECP Control Box. Using equipment such as an oscilloscope, plotter, or spectrum analyzer the user may inspect the following items continuously in real time:

- Commanded Position
- Encoder 1 Position
- Encoder 2 Position
- Encoder 3 Position
- Control Effort
- Node A
- Node B
- Node C
- Node E

The scale factor which divides the item can be less than 1 (one). The DACs analog output is in the range of +/- 10 volts corresponding to +32767 to -32768 counts. For example to output the commanded position for a sine sweep of amplitude 2000 counts you should choose the scale factor to be 0.061 ($2000/32767=0.061$) This gives close to full +/- 10 volt reading on the analog outputs. In contrast, if the numerical value of an item is greater than +/- 32767 counts, for full scale reading, you must choose a scale factor of greater than one. Note that the above items are always in counts (not degrees or radians) within the real time controller and since the DAC's are 16-bit wide, + 32767 counts corresponds to +9.999 volts, and -32768 counts corresponds to -10 volts.

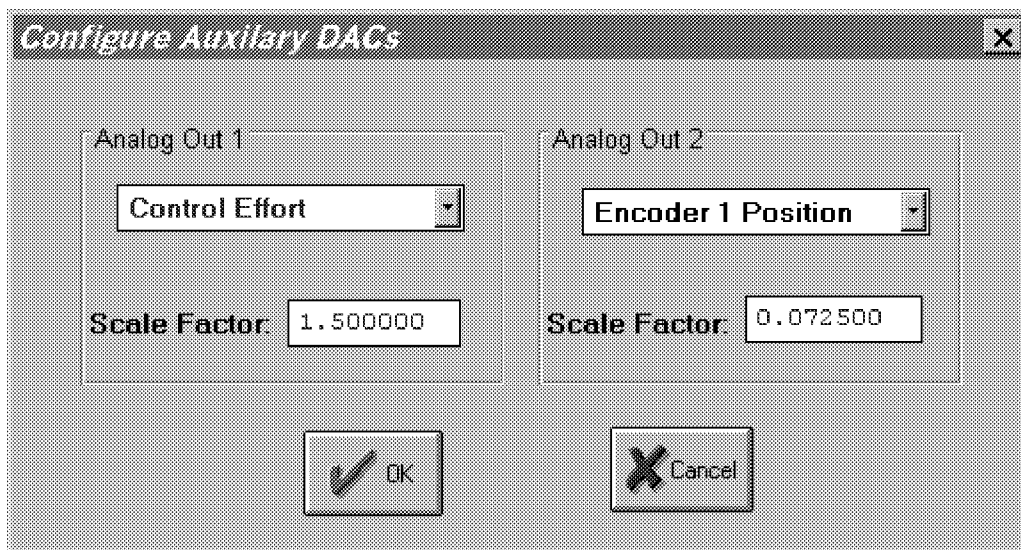


Figure 2.1-11. The Configure Auxiliary DACs Dialog Box

2.1.9.2 The Jog Position option enables the user to move the mechanism to a different commanded position. In contrast to displacements executed under the Trajectory dialog box, during a Jog command no data is acquired for plotting purposes. Since this motion is effected via the current controller, one can only jog under closed loop control with a stable controller. By selecting the appropriate radio button either incremental and absolute displacements may be carried out. The jogging feature allows the user to return to a known position after the execution of the various forms of open and closed loop trajectories.

2.1.9.3 The Zero Position option enables the user to reinitialize the current position as the zero position. Note that if following errors exists, then the actual positions may be other than zero even though the commanded position is at zero (since the action is similar to commanding an instantaneous zero set point, a sudden small

jerk in position may occur).

2.1.9.4 The Reset Controller option allows the user to reset the real-time controller. Upon Power up and after a reset activity, the loop is closed with zero gains and there it behaves in the same way as in the open loop state with zero control effort. Thus the user should be aware that even though the Control Loop Status indicates "closed loop", all of the gains are zeroed after a Reset. In order to implement (or re implement) a controller you must go to the Setup Control Algorithm box.

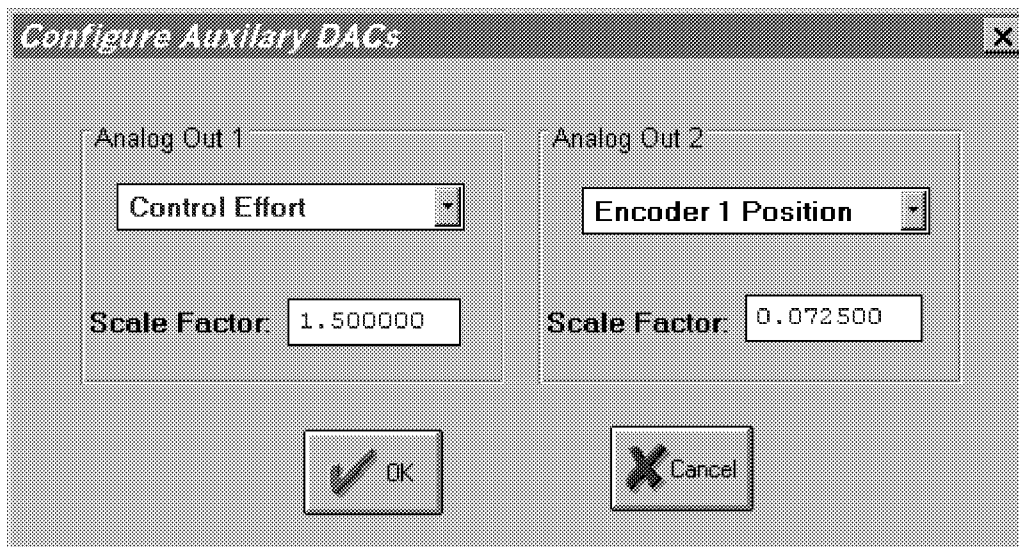


Figure 2.1-10. The Configure Auxiliary DACs Dialog Box

2.1.9.5 The Rephase Motor option enables a user to simply rephase a brushless motor's commutation phase angle. This feature is not used by the Model 210 system since its motors use absolute sensors for commutation.

2.1.9.6 The Download Controller Personality File is an option which should not be used by most users. In a case where the real-time controller irrecoverably malfunctions, and after consulting ECP, a user may download the personality file if a ".pmc" file exists. In the case of Model 210, this file is named "m210.pmc". This downloading process takes a few seconds.

2.2 Electromechanical Plant

2.2.1 Design Description

The mechanism shown in Figure 2.2-1 is designed to emulate a broad range of real-world applications including 1 DOF rigid bodies, flexibility in linear drives, gearing and belts, and other coupled discrete oscillatory systems. The apparatus, shown in Figure 2.2-1, consists of two (Model 210) or three (Model 210a) mass carriages interconnected by bi-directional springs. The mass carriage suspension is an anti-friction ball bearing type with approximately ± 3 cm of available travel. The linear drive is comprised of a gear rack suspended on an anti-friction carriage and pinion (pitch dia. 7.62 cm (3.00 in)) coupled to the brushless servo motor shaft. Optical encoders measure the mass carriage positions – also via a rack and pinion with pinion pitch dia 3.18 cm (1.25 in)^[17].

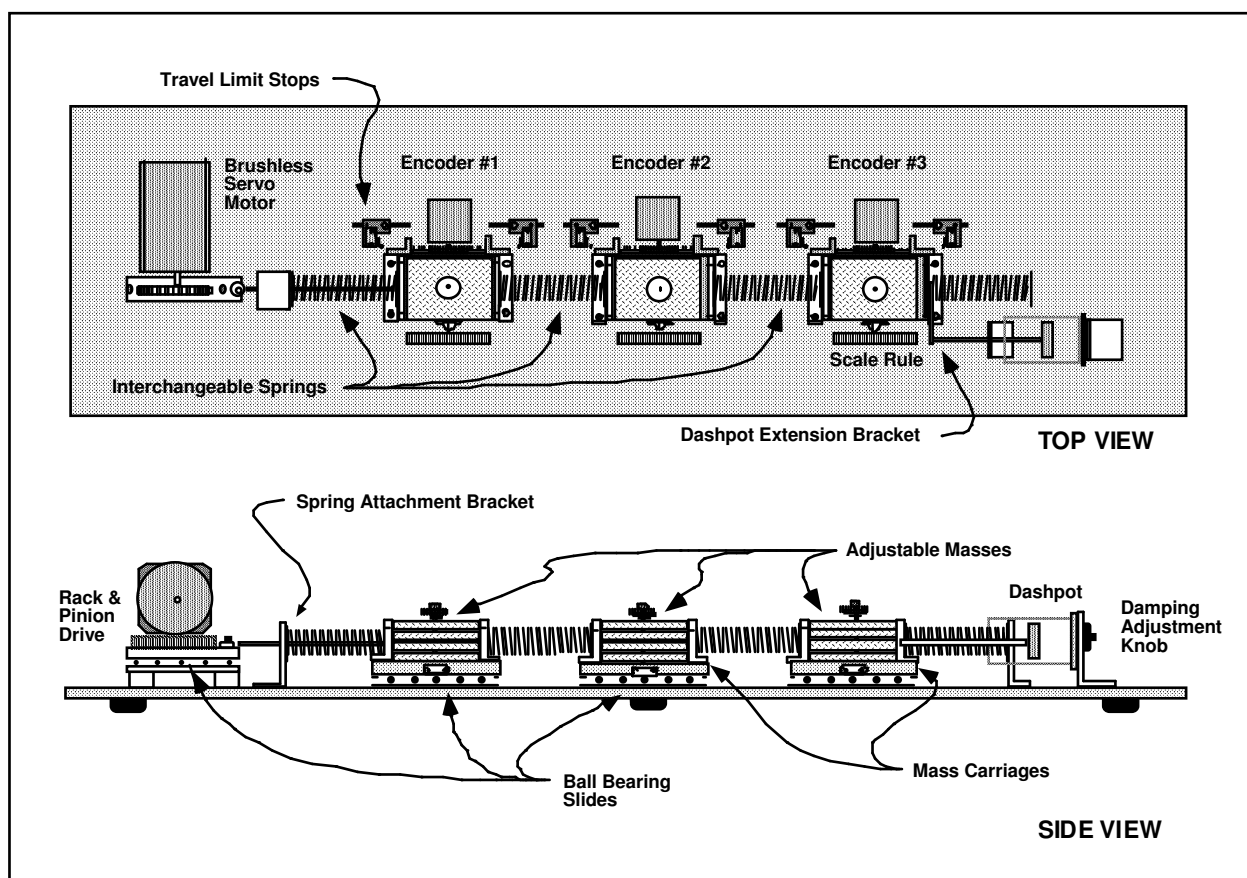


Figure 2.2-1.

Rectilinear Apparatus

Springs of various stiffness may be attached between masses or between the masses and the base plate. A dashpot with adjustable damping may be coupled to any of the masses. Position measurement scales are provided to assist in certain experiments and system setup.

Figures 2.2-2a & b also show the variety of plant types that are supported by each model. Mathematical modeling and parameter identification of these plants are described in Chapters 5 and 6.

2.2.2 Changing Plant Configurations and Parameter Values

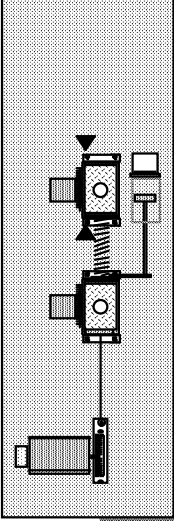
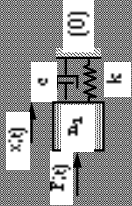
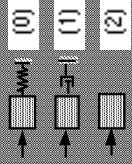
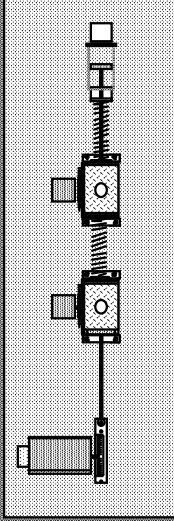
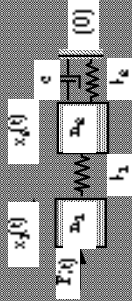
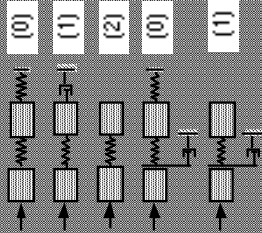
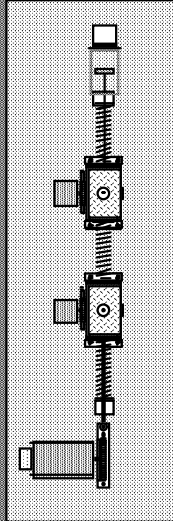

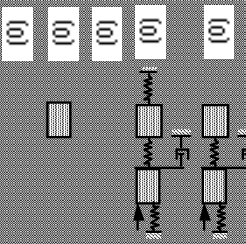
As shown in Figure 5.2.2-2, the plant may be placed in a variety of spring/mass/damper configurations with 1, 2, and 3 (Model 210a) degrees of freedom (DOF). Springs may be mounted interchangeably and are supplied

in three nominal stiffnesses: 175 N/m (1.0 lb/in), 450 N/m (2.25 lb/in), and 800 N/m (4.5 lb/in).

The dashpot connects to any of the mass carriages either through direct attachment or via the extension bracket. The damping constant may be varied by adjusting the air flow valve on the rear of the dashpot. Coarse adjustments are made via the knurled knob and fine adjustments via the metal screw.

The user may change inertia values by changing the number of brass weights (500 ± 5 g each). Make certain that the weights are firmly clamped by the thumb nut before operating or transporting the mechanism.

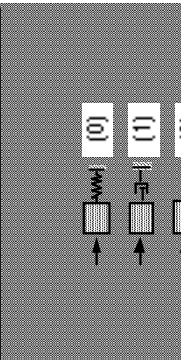
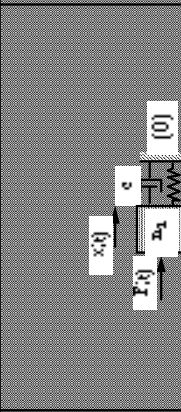
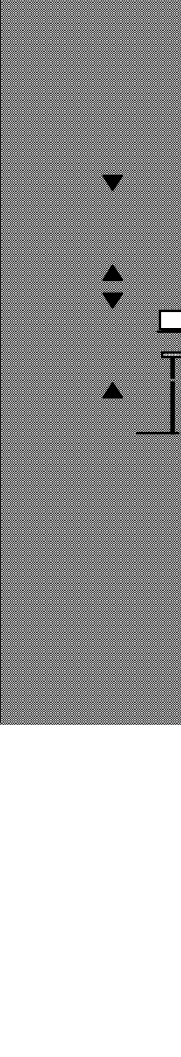
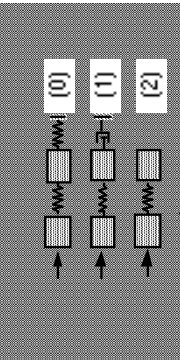
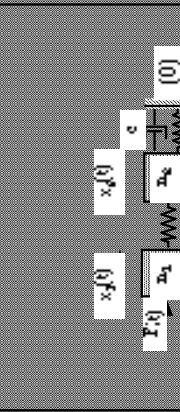
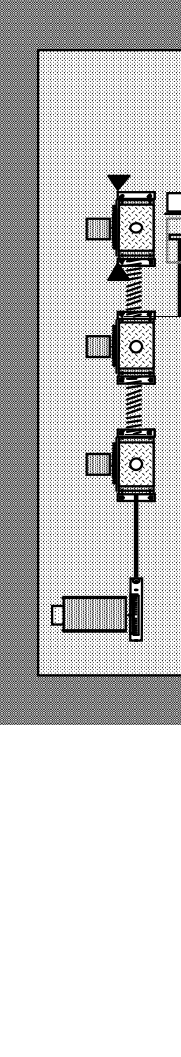
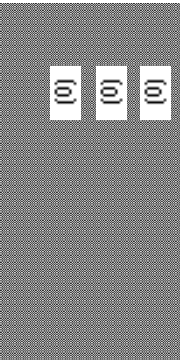
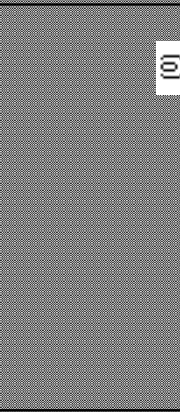
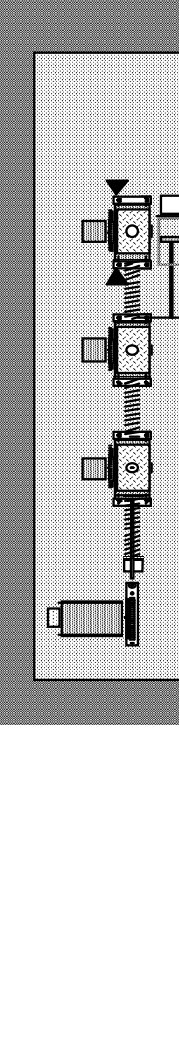
The travel limit stops must also be set before operating or transporting. Figure 2.2-3 gives instructions for adjusting the stop positions. The stop assemblies include a rubber "bumper" to physically stop the motion, and a limit microswitch^[18] set to engage prior to the stop tab contacting the bumper. When a given mass carriage is intended to have free motion, the limits should generally be set to ± 3 cm. Whenever the stop positions are moved, the user should verify that they are securely fastened in their new position. It should also be checked that the limit switches engage ("click" sound) prior to contacting the bumper, and that they are contacted prior to interference between any other mechanical elements (e.g. that the encoder cable ends do not travel beyond their pulley). For some plant configurations, the second or third mass must be clamped. In this case a shim (e.g. a 1/4 in. threaded nut) must be placed between the bumper and the stop tab to ensure that the corresponding limit switch is not engaged – see Figure 2.2-3.

Spring / Mass / Damper Configuration	System Model	Additional Configurations*
		
		
		

* Numbers in parenthesis are the so-called 'system type', or number of integrators in the plant.

Figure 2.2-2a. Plant configurations for Model 210

- indicates mass range is immobilized by limit switch.

Spring / Mass / Damper Configuration	System Model	Additional Configurations*
		
		
		

Spring / Mass / Damper Configuration	System Model	Additional Configurations

* Numbers in parenthesis are the so-called "system type" or number of integrators in the plant.

Figure 2.2-2b. Plant configurations for Model 210a

◀ indicates mass carriage is immobilized by limit switch element.

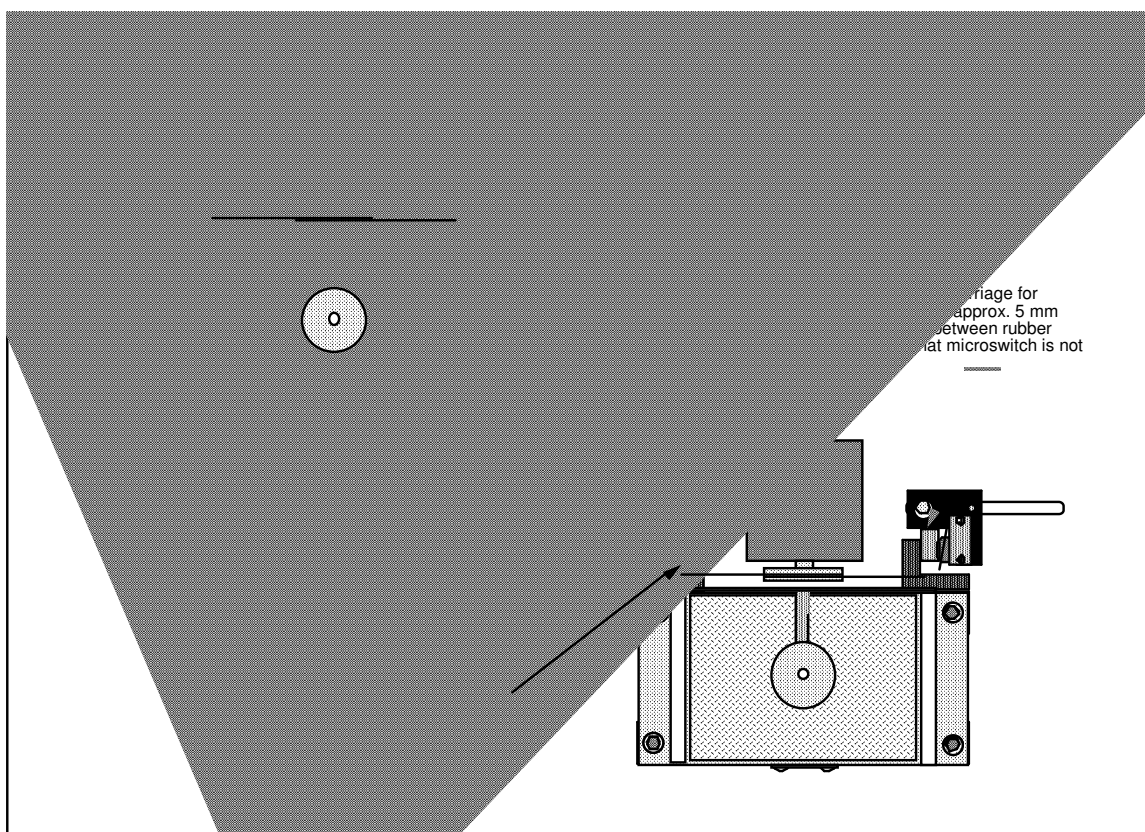


Figure 2.2-3. Guidelines For Setting Stop Positions

Important Notes

1. Prior to operation the user should verify that: a) the brass mass pieces are properly secured, b) the mass carriages travel freely, and c) the limit stop blocks are properly set and secured to limit motion of the mass carriages.
2. Prior to transporting the mechanism the user should verify that: a) all brass mass pieces are removed, b) the limit stop blocks are set to immobilize the mass carriages and are secured, and c) the disturbance drive and its carriage plate (if included as system option) are secured using the provided screws.

2.2.3 Damper Protective Cover

The damper is supplied with a protective black elastomeric cover. The user or using institution may choose to remove the cover to expose the dashpot's glass cylinder body and give a more illustrative view of its functionality. This is easily accomplished by parting the elastomer along its length with a sharp knife and peeling away. The user is warned however that the cover is provided to help protect against damage by objects striking the dashpot. ECP will not be responsible for damage resulting from the removal of the cover.

2.2.3 Optional Disturbance Drive

The optional disturbance drive consists of a high torque density DC motor and capstan drive - see Figure 5.2-4. It may be installed to apply a disturbance to any of the mass carriages using existing features on the Model 210 apparatus. Detailed instructions for installing the drive are given in Figure 5.2-5. The drive should be stowed when not in use according to the reverse procedure of Figure 5.2-5. It is important that the drive be properly secured and the mass carriage be shown to slide freely before operation of the system. The unused ruler and thumb screws should also be secured as per the figures to avoid their loss or damage. If the drive has been purchased as an "add-on" accessory (not purchased with the Model 210 system originally) the two storage holes

shown in the upper right mechanism base plate (upper & lower Figure 2.5-5) may not be available.

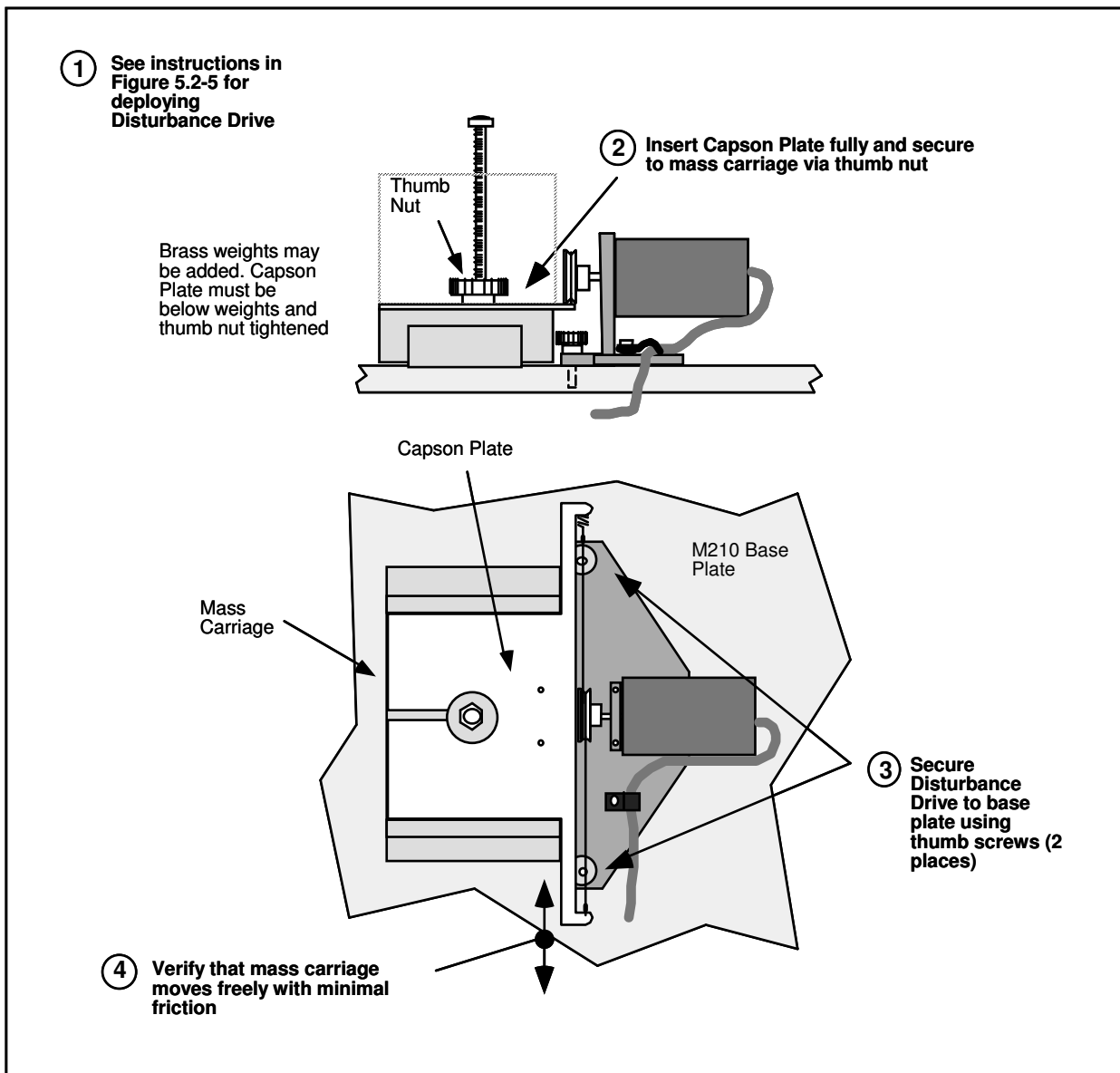


Figure 5.2-4. The Disturbance Drive & Its Installation

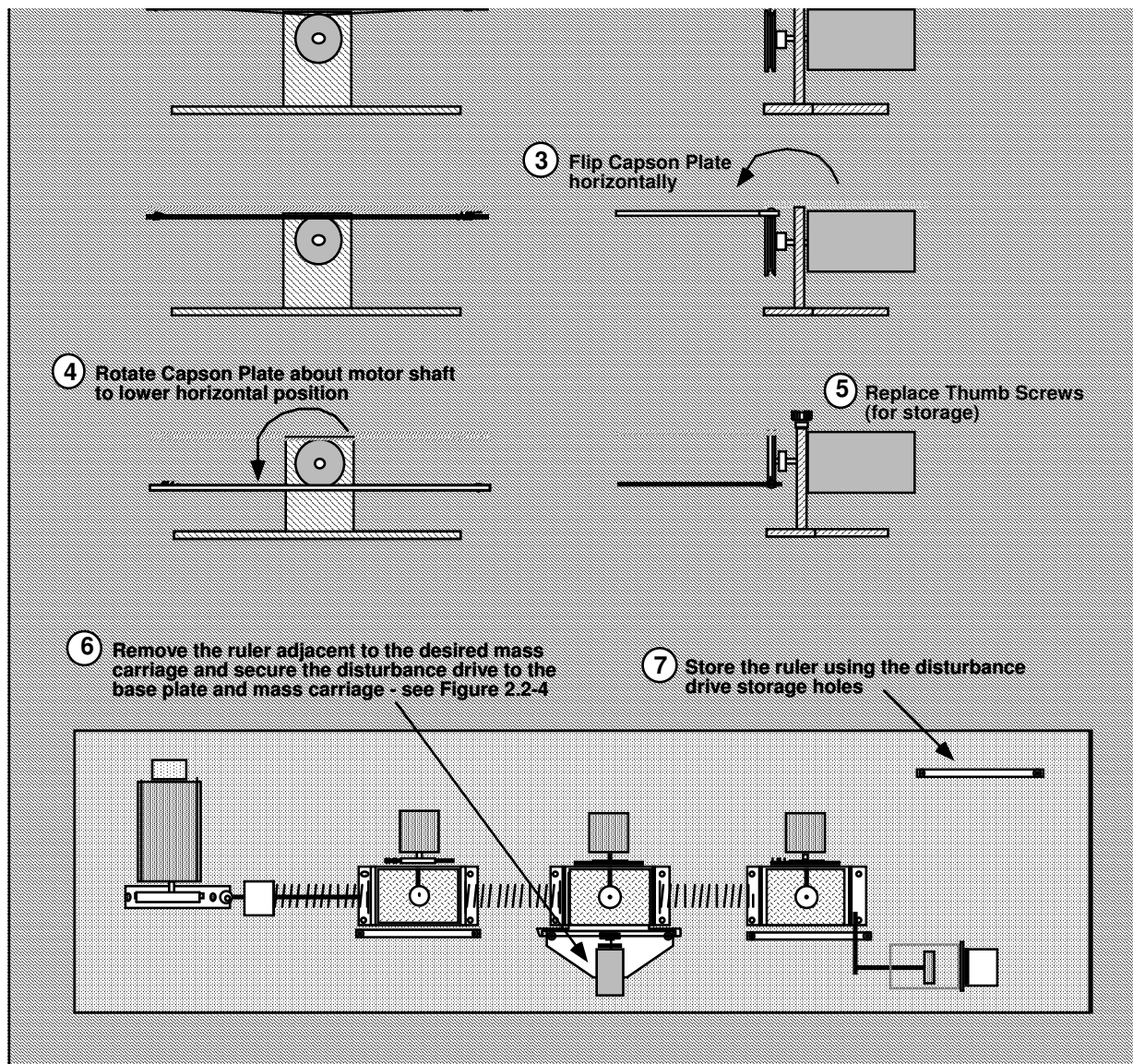


Figure 5.2-5. Disturbance Drive Setup & Installation Details

2.3 Safety

The following are safety features of the system and cautions regarding its operation. This section must be read and understood by all users prior to operating the system. If any material in this section is not clear to the reader, contact ECP for clarification before operating the system.

Important Notice: In the event of an emergency, control effort should be immediately discontinued by pressing the red "OFF" button on front of the control box.

2.3.1 Hardware

A relay circuit is installed within the Control Box which automatically turns off power to the Box whenever the real-time Controller (within the PC) is turned on or off. Thus for the PC bus version^[19] of the real-time Controller the user should turn on the computer prior to pressing on the black ON switch. This feature is implemented to prevent uncontrolled motor response during the transient power on/off periods. The power to the Control Box may be turned off at any time by pressing the red OFF switch.

Although not recommended, it will not damage the hardware to apply power to the Control-Box even when the PC is turned off. However, doing so does not result in motor activation as the motor current amplifier will be disabled. The *amplifier enable* signal input to the Control Box is connected to the real-time Controller via the 60-pin flat ribbon cable. This input operates in a normally closed mode. When power to the real-time Controller is off, this input becomes open which in turn disables the motor amplifier.

The recommended procedure for **start up** is as follows:

First : Turn on the PC with the real-time Controller installed in it.

Second: Turn on the power to Control Box (press on the black switch).

The recommended **shut down** procedure is:

First: Turn off the power to the Control Box.

Second: Turn off the PC.

FUSES: There are two 2.5A 120V slow blow fuses within the Control Box. One of them is housed at the back of the Control Box next to the power cord plug. The second one is inside the box next to the large blue colored capacitor. See warnings in Section 2.3.4 regarding replacement of the fuses.

2.3.2 Software

The Limit Exceeded indicator of the Controller Status display indicates either one or more of the following conditions have occurred:

- Travel limits have been contacted.

- Over speed of the motor.

- Excessive motor drive power

The real-time Controller continuously monitors the above limiting conditions in its background routine (intervals of time in-between higher priority tasks). When one if these conditions occurs, the real-time Controller opens up the control loop with a zero force command sent to the actuator. The Limit Exceeded indicator stays on until a new set of (stabilizing) control gains are downloaded to the real-time Controller. One should first Reset Controller (Utility menu) to be certain that no anomalous states exist in the controller due to the interruption caused by the Limit Exceeded condition, then Implement Algorithm (via the Setup Control Algorithm dialog box). Obviously the input trajectory must also have parameters that do not cause the Limit Exceeded condition.

The Limit Exceeded indicator of the optional Disturbance Motor Status display indicates either one or both of the following conditions have occurred:

- Over speed of the disturbance motor.

- Excessive disturbance motor power

Also included is a *watch-dog timer*. This subsystem provides a fail-safe shutdown to guard against software malfunction and under-voltage conditions. The use of the watch-dog timer is transparent to the user. This shutdown condition turns on the red LED on the real-time Controller card, and will cause the control box to power down automatically. You may need to cycle the power to the PC in order to reinitialize the real-time Controller should a watch-dog timer shutdown occur.

2.3.3 Safety Checking The Controller

While it should generally be avoided, in some cases it is instructive or necessary to manually contact the mechanism when a controller is active. This should always be done with caution and never in such a way that clothing or hair may be caught in the apparatus. By staying clear of the mechanism when it is moving or when a trajectory has been commanded, the risk of injury is greatly reduced. Being motionless, however, is not sufficient to assure the system is safe to contact. In some cases an unstable controller may have been implemented but the system may remain motionless until perturbed – then it could react violently.

In order to eliminate the risk of injury in such an event, you should always safety check the controller prior to physically contacting the system. This is done by lightly grasping a slender, light object with no sharp edges (e.g. a ruler without sharp edges or an unsharpened pencil) and using it to slowly move the first mass (the one connected to the motor drive) from side to side. Keep hands clear of the mechanism while doing this and apply only light force to the mass. If the mass does not race away up or oscillate then it may be manually contacted – but with caution. This procedure must be repeated whenever any user interaction with the system occurs (either via the Executive Program or the Controller Box) if the mechanism is to be physically contacted again.

2.3.4 Warnings

WARNING #1: Stay clear of and do not touch any part of the mechanism while it is moving, while a trajectory has been commanded (via Execute, Command menu), or before the active controller has been safety checked – see Section 2.3.3.

WARNING #2: The following apply at all times except when motor drive power is disconnected (consult ECP if uncertain as to how to disconnect drive power):

- a) Stay clear of the mechanism while wearing loose clothing (e.g. ties, scarves and loose sleeves) and when hair is not kept close to the head.
- b) Keep head and face well clear of the mechanism.

WARNING #3: Do not apply power to the mechanism while the clear cover about the drive rack and pinion is not securely in place.

WARNING #4: Verify that the masses and stops are secured per Section 2.2 of this manual prior to powering up or transporting the Control Box.

WARNING #5: Do not take the cover off or physically touch the interior of the Control Box unless its power cord is unplugged (first press the "Off" button on the front panel) and the PC is unpowered or disconnected.

WARNING #6: The power cord must be removed from the box prior to the replacement of any fuses.

3. Start-up & Self-guided Demonstration

This chapter provides an orientation "tour" of the system for the first time user. In Section 3.1 certain hardware verification steps are carried out. In Section 3.2 a self-guided demonstration is provided to quickly orient the user with key system operations and Executive program functions. Finally, in Section 3.3, certain system behaviors which may be nonintuitive to a first time user are pointed out .

All users must read and understand Section 2.3, Safety, Before performing any procedures described in this chapter.

3.1 Hardware Setup Verification

At this stage it is assumed that

- a) The ECP Executive program has been successfully installed on the PC's hard disk (see Section 2.1.2).
- b) The actual printed circuit board (the real-time Controller) has been correctly inserted into an empty slot of the PC's extension (ISA) bus (this applies to the PC bus version only).
- c) The supplied 60-pin flat cable is connected between the J11 connector (the 60-pin connector) of the real-time Controller and the JMACH connector of the Control Box^[20].
- d) The other two supplied cables are connected between the Control Box and the mechanical apparatus;
- e) The apparatus has two brass masses on each carriage, the least stiff spring connected between the base plate and the first mass, and the stiffest spring between the first and second masses (also between the second and third for Model 210a). The damper should be attached to the third (Model 210a) or second (Model 210) mass and have its air valve opened two full turns from the fully closed position. (i.e. the plant should be set up as shown in Figure 2.2-1 except with no spring connected between the right-most mass and the base plate.) This is the configuration that the apparatus is set up in when shipped from the factory except that the stops should have been moved to allow approximately ± 3 cm of travel of each mass. (See Section 2.2 for setting the stops.)
- f) You have read the safety Section 2.3. All users should read and understand that section before proceeding.

Please check the cables again for proper connections.

3.1.1 Hardware Verification (For PC-bus Installation)

Step 1: Switch off power to both the PC and the Control Box.

Step 2: With power still switched off to the Control Box, switch the PC power on. Enter the ECP program by double clicking on its icon (or type ">ECP" in the appropriate directory under DOS). You should see the Background Screen (see Section 2.1.3). Gently move the first mass carriage (the one closest to the motor) by hand. You should observe some following errors and changes in encoder counts. The Control Loop Status should indicate "CLOSED" and the Controller Status should indicate "ACTIVE". If this is the case skip Step 3 and go to Step 4.

Step 3: If the ECP program cannot find the real-time Controller (a pop-up message will notify you if this is

the case), try the Communication dialog box under the Setup menu. Select PC-bus at address 528, and click on the test button. If the real-time Controller is still not found and you are using the ECP Executive for DOS, try increasing the time-out in increments of 20 up to a maximum of 300. If you are using the ECP Executive for Windows, try increasing the time-out in increments of 5000 up to a maximum of 80000. If this doesn't correct the problem, switch off power to your PC and then take its cover off. With the cover removed check again for the proper insertion of the Controller card. Switch the power on again and observe the two LED lights on the Controller card. If the green LED comes on all is well; if the red LED is illuminated, you should contact ECP for further instructions. If the green LED comes on, turn off power to your PC, replace the cover and turn the power back on again. Now go back to the ECP program and you should see the positions change as you gently slide the masses.

Step 4: Make sure that the masses slide freely. Now connect the power cord to the Control Box and press the black "ON" button to turn on the power to the Control Box. You should notice the green power indicator LED lit, but the motor should remain in a disabled state. Do not touch the masses whenever power is applied to the Control Box since there is a potential for uncontrolled motion (see the warnings in Section 2.3) unless the controller has been safety checked.

This completes the hardware verification procedure. Please refer again to Section 2.3 for future start up and shut down procedures.

3.2 Demonstration of ECP Executive Program

This section walks the user through the salient functions of the system. By following the instructions below you will actually implement a controller, maneuver the system through various trajectories, and acquire and plot data. The procedures given in this section are for Model 210a (3 mass) and also apply to Model 210 except where noted. The data plots shown are for Model 210a. The corresponding plots for Model 210 are given in Section 3.2.1 immediately following this section.

Step 1: Loading A Configuration File. With the power to the Control box turned off, enter the ECP Executive program. You should see the Background Display. Turn on power to the Control Box (press on the black button). Now enter the File menu, choose Load Settings and select the file `default.cfg`. This configuration file is supplied on the distribution diskette and should have been copied under the ECP directory by now. Note that this file would have been loaded into the Executive automatically (see section 2.1.4.1). This particular `default.cfg` file contains the controller gain parameters and other trajectory, data gathering and plotting parameters specifically saved for the activities within this section.

Note again that this file has been created to operate with the plant configuration as described in item e of Section 3.1.

Step 2: Implementing The Controller. Now enter the Setup menu and choose Setup Control Algorithm. You should see the sampling time $T_s = 0.004420$ seconds, Continuous Time control type, and State Feedback control selected. Select Setup Algorithm button and you should see for Model 210a the following gains: $k_{pf}=0.885$, $k_1=0.6084$, $k_2=0.0138$, $k_3=0.0642$, $k_4=0.0307$, $k_5=0.2130$ and $k_6=0.0281$. (For Model 210, the gains are $k_{pf}=0.6708$, $k_1=0.4383$, $k_2=0.01026$, $k_3=0.2325$, $k_4=0.0214$.) This controller was designed using the linear quadratic regulator (LQR) technique with error weighting on encoder 3 (encoder 2 for Model 210, i.e. the noncollocated encoder at the inertia furthest away from the drive motor).

Exit this dialog box and select Implement Algorithm. The control law is now downloaded to the real-time Controller. Use a ruler to verify that control action is in effect on the first mass (see Sect. 2.3.3, "Safety Checking The Controller") If you do not notice any motor power, click on the Implement Algorithm button again until you notice the servo loop closed. Now select Upload Algorithm, then General Form and Setup Algorithm. You should see the general form of the real-time algorithm with some non-zero coefficients. These coefficients correspond to the state feedback gains mapped to a form suitable for execution via the general form algorithm. Note that the general form algorithm is the only structure that actually runs in real-time. All other structures (e.g. PID, dynamic filters, etc.) are translated to the general form by the Executive prior to implementation.

Step 3: Setting Up Data Acquisition. Enter the Data menu and select Setup Data Acquisition. In this box make sure that the following four items are selected: Commanded Position and Encoders 1,2, & 3 (Encoders 1 & 2 only for Model 210). *Data sample period* should be 2 which means that data will be collected every second servo cycle (in this case every $2*0.00442=0.00884$ seconds).

Step 4: Executing A Step Input Trajectory & Plotting. Enter the Command menu and select Trajectory. In this box select Step and then Setup. You should see *step size = 2500*, *dwel time =1000 ms* and *no. of repetitions = 1*. If not, change the values to correspond to this parameter set. Exit this box and go to the Command menu. This time select Execute and with the Sample Data check box checked run the trajectory and wait for the data to be uploaded from the real-time Controller. You

should have noticed a step move of 1000 counts, a dwell of 1.45 second and a return step move. Now enter the Plotting menu and choose Setup Plot. Select Encoder 3 (Encoder 2 for Model 210) and Commanded Position for plotting (left axis) and select Plot Data.

You should see a plot similar to the one shown in Figure 3.2-1. Replace Encoder 3 with 2 via the Setup Plot box and then with Encoder 1, plotting the data each time. Similar plots to the ones shown in Figures 3.2-2 and 3.2-3 should be seen. Note that all plot screens may be repositioned by dragging the plot frame bar using the mouse. A plot screen may be minimized or "shrunk" by clicking on the left most button on the top right hand corner. By double clicking on a minimized plot screen it may be expanded again. In general, to limit the program memory usage, one should avoid minimizing more than 8 to 10 plot screens on the Background screen. It is best to close unwanted plot screens rather than minimize them. You can re-size the plots using the cursor to "drag" any edge of the plot frame (position cursor at corner and begin "dragging" when cursor becomes double arrow).

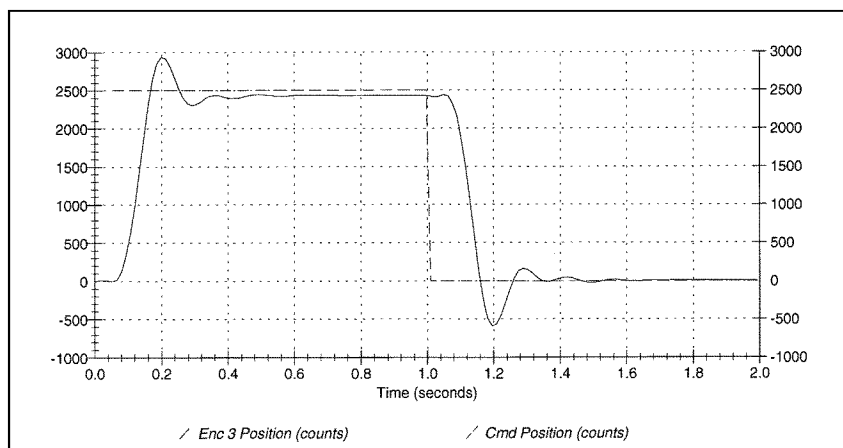


Figure 3.2-1 Step Response at Encoder 3

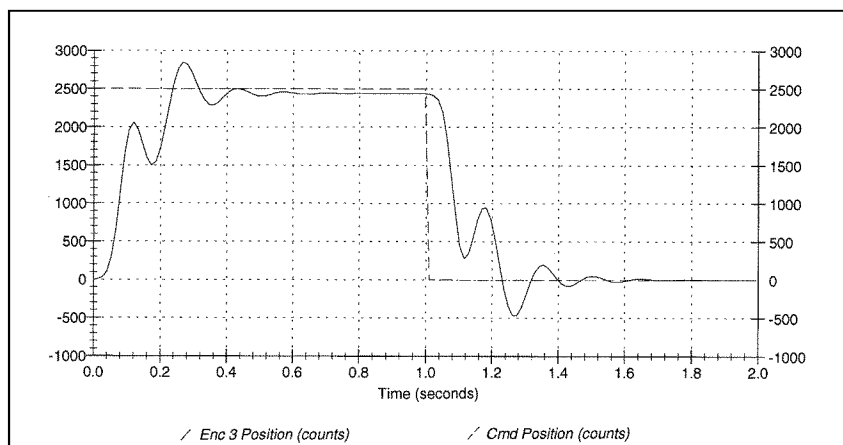


Figure 3.2-1 Step Response at Encoder 2



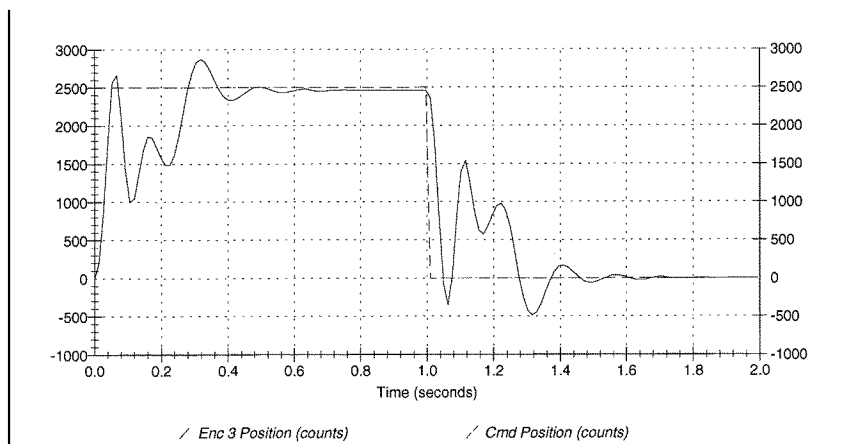


Figure 3.2-1 Step Response at Encoder 1

Step 5: Tracking Response. Return to Trajectory and select Ramp and then Setup to enter the *Ramp* dialog box. You should see *distance* = 5000 counts, *velocity* = 10000 counts/s, *dwel time* = 500 ms and *no. of repetitions* = 1. If not, change the values to this set. Exit this box and go to the Command menu. Now select Execute and with Sample Data checked, run the trajectory and wait for the data to be uploaded from the real-time Controller. You should have noticed the ramp move of 5000 counts, a one-half second dwell, a return ramp, then the cycle repeating. Now enter the Plotting menu and choose Setup Plot. Select Encoder 3 and Commanded Position for plotting and then plot the data. You should see a plot similar to the one shown in Figure 3.2-4. Replacing Encoder 3 with 1 and replotting should result in a response similar to Figure 3.2-5. (Encoder 3 tracks best due to the lqr design objective with error weighting on mass 3). You may save the ramp response under *anyname.plt* using the Save Plot Data option. Any plot data thus saved may be reloaded from the disk using the Load Plot Data option for future inspection, plotting or printing. To print simply choose the Print Data menu option and select the appropriate printer before the printing command. Alternatively, any set of collected data may be exported as an ASCII text file by the use of the Export Data option of the Data menu. Now close all plot windows.

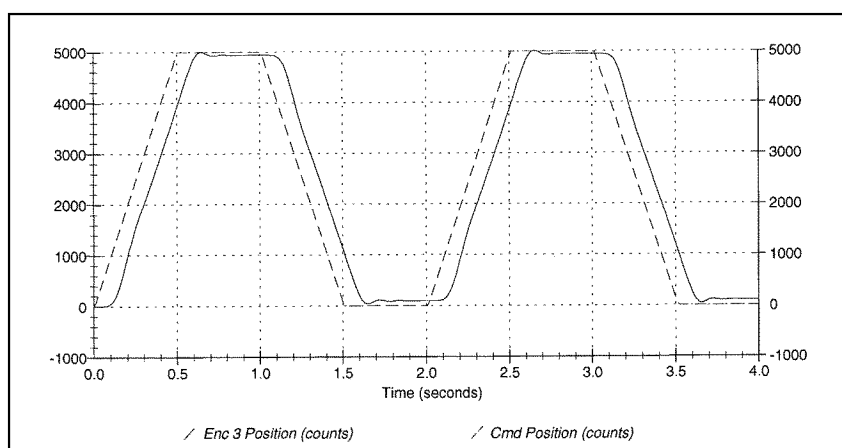


Figure 3.2-4 Ramp Response at Encoder 3

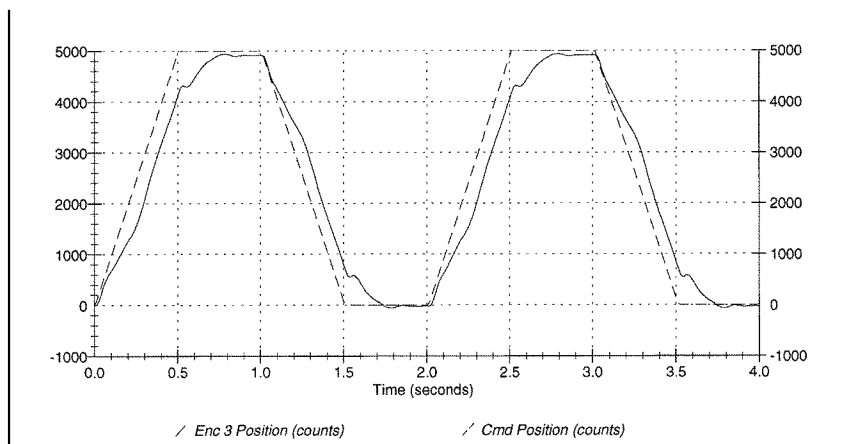


Figure 3.2-5 Ramp Response at Encoder 1

Step 6: Frequency Response. Again enter Trajectory and select Sine Sweep then Setup. You should see the *amplitude = 400 counts, max. freq. = 9.9 Hz, min. freq. = 0.1 Hz* and the *sweep time = 29.5 sec.* Again, if different, change the values to correspond this set. Exit this box and go to the Command menu and Execute with Sample Data checked. While running this trajectory, you should notice sinusoidal motion with increasing frequency for about thirty seconds. Now enter the Plotting menu and choose Setup Plot. This time select only Encoder 3 (Encoder 2 for Model 210) for plotting and choose Linear Time and Linear amplitude scaling for the horizontal and vertical axes; then plot the data. You should see a plot similar to the one shown in Figure 3.2-6a. You may wish to select to reduce the small amplitude drift at high frequency. Now replace Encoder 3 with 2 and then with 1. Plots similar to the ones shown in Figures 3.2-6b&c should be seen. Now return to the Sine sweep trajectory box and select with all else unchanged. Execute this maneuver and plot the data using Logarithmic Frequency and Db axis scaling. Now the plots should appear similar to those shown in Figure 3.2-6b.^[21] In general, an improved plot appearance results from selecting Remove DC Bias in the Setup Plot window. This reduces the affect of small amplitude drift at high frequency.

NOTE: For sine sweeps to accurately reflect the system response, the Model 210 apparatus must be setting on a rigidly supported surface. Many tables and counter tops have significant flexibility (not initially obvious) and will result in altered shape of the response. This may usually be checked by redoing the sweep with the apparatus setting on the floor to see if there is a change in its response signature.

The linear time / amplitude depiction shows the data in a manner that for most students more intuitively represents the physical motion of the system as witnessed. The $\text{Log}(\omega)$ / Db scaling presents the data in a way that leads to a physical understanding of the Bode magnitude plots commonly found in the literature. The resonances and anti-resonances (lightly damped poles and zeros) seen in these responses agree with their transfer function models. That is, the resonances (poles) are common to all outputs, and masses 1, 2, & 3 have zero, one, and two zero pairs respectively. The high frequency attenuation is greater with mass number reflecting the respective two, four, and six pole excesses. Similarly insightful properties of open loop systems are found by measuring their sine sweep responses.

Linear Time / Linear Amplitude Scaling	Logarithmic Frequency / Db Scaling

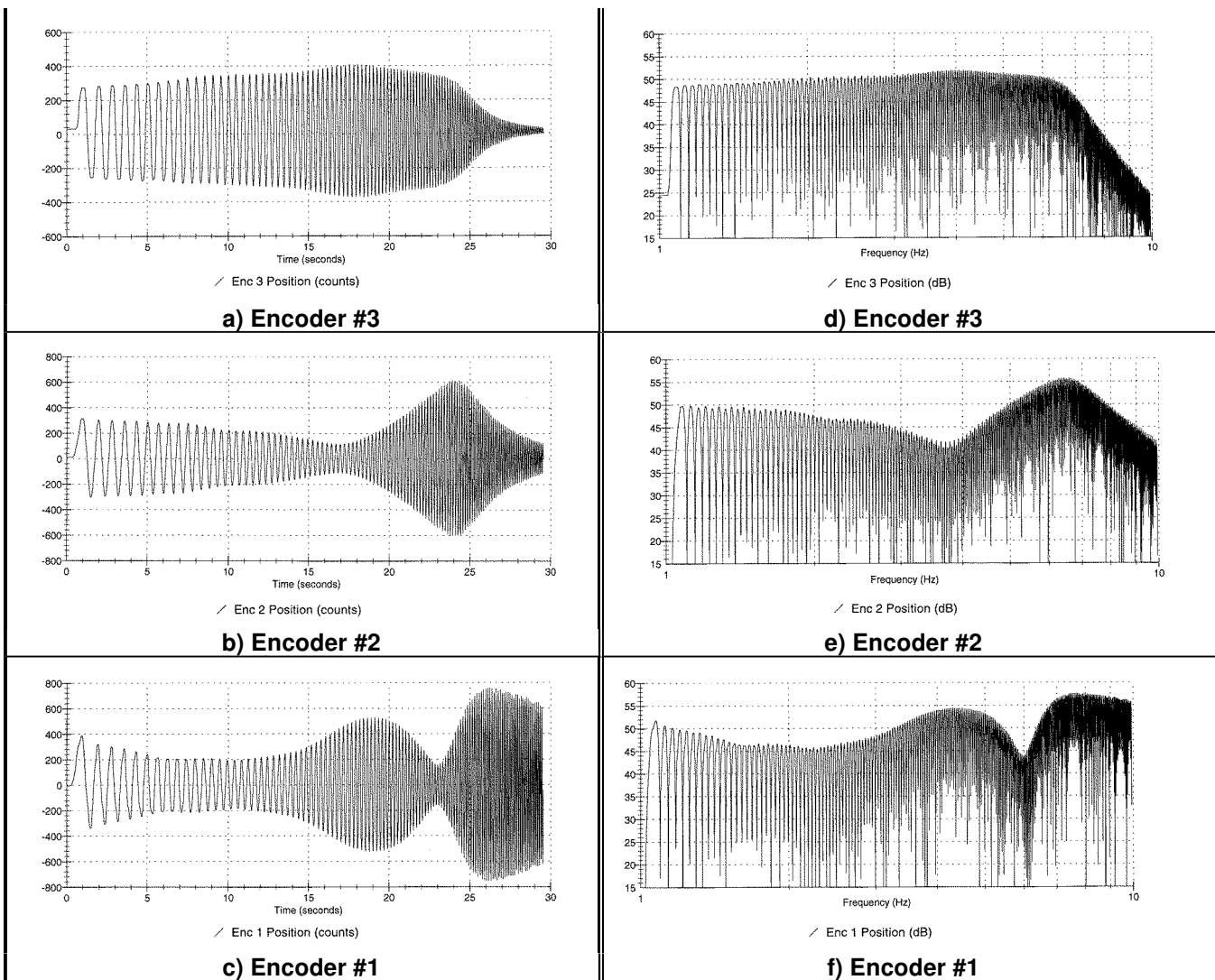


Figure 3.2-6i Sine Sweep (Frequency) Response Data With Various Trajectory / Plotting Options

This completes the self-guided tour of the Model 210a system. It is strongly advised that the user read all material in Chapters 2 and 3 before further operation of the equipment.

3.2.1 Plot Results For Model 210

The plots resulting from the self-guided tour for Model 210 (2 mass carriage) are as follows.

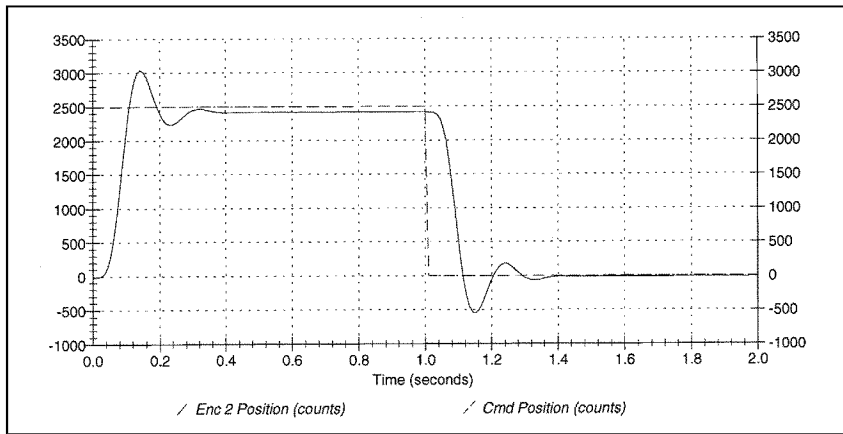


Figure 3.2-7 Step Response at Encoder 2

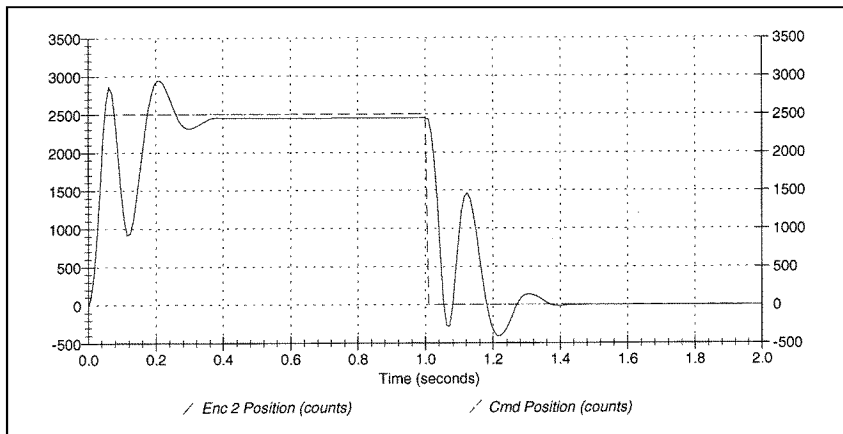


Figure 3.2-8 Step Response at Encoder 1

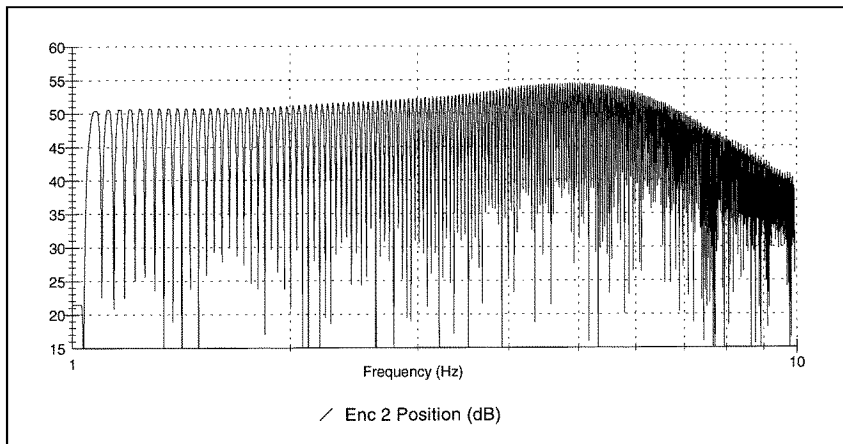
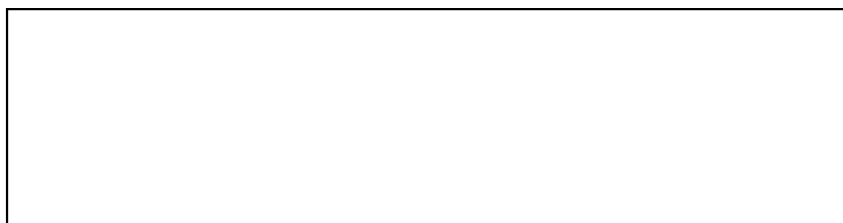


Figure 3.2-9 Sine Sweep Response at Encoder 2



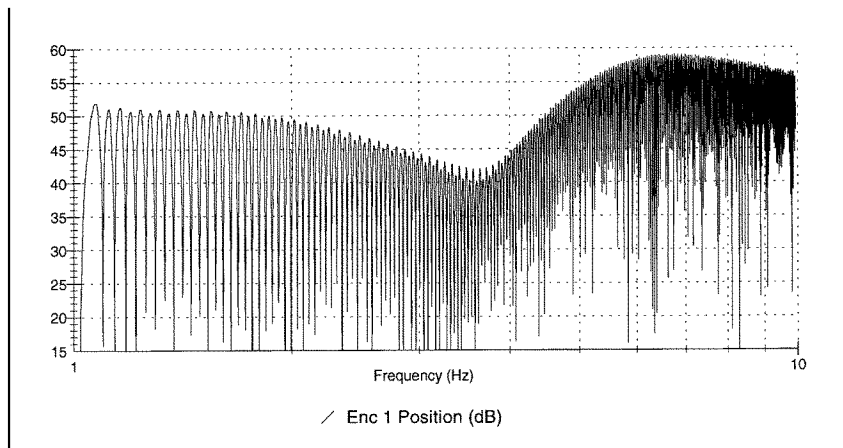


Figure 3.2-10 Sine Sweep Response at Encoder 1

The ramp following results appear similar to those in Figures 3.2-4,5 and are not shown here.

This completes the self-guided tour of the Model 210 system. It is strongly advised that the user read all material in Chapters 2 and 3 before further operation of the equipment.

3.2.2 Using The Optional Disturbance Drive

The optional disturbance drive is useful in studying the important system properties of regulation performance and disturbance rejection at any of the system outputs (inertia disks). It may also be used to apply viscous friction at any of these locations. The data shown below is for Model 210a (three mass) but is also representative of that for the basic Model 210.

Step 1: Disturbance Rejection During System Regulation

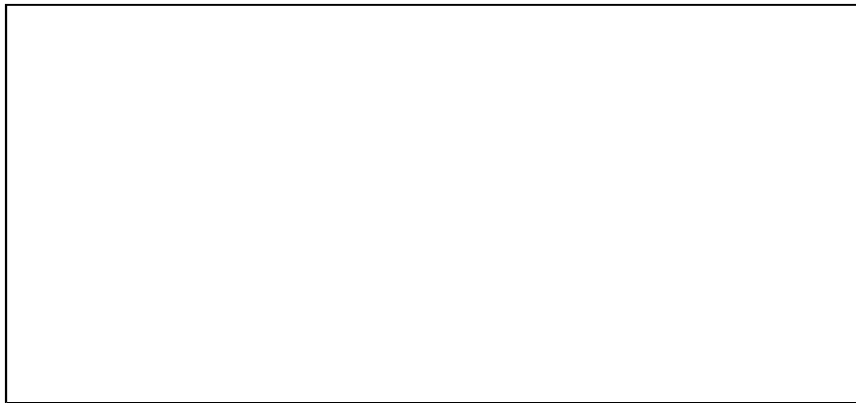
Attach the disturbance Drive to the third carriage (Encoder #3) as per Section 2.2.3. Implement the controller according to Section 3.2 Step #2. (This is the default controller shipped with the system. If you have been following the Self Guided Tour until now, you should simply need to select Implement Algorithm in the Setup Control Algorithm dialog box.) The system should now be regulating (in closed loop control with a zero valued reference input). Enter the Disturbance Configuration dialog box and select Sinusoidal (time). Specify an amplitude of 1 V, frequency of 1 Hz, and 5 Repetitions. Enter Trajectory Configuration. Setup a Step input with *distance* = 0 counts, *dwel time* = 2000 ms and *no. of repetitions* = 1. This will provide a zero amplitude input for gathering data during the disturbance. Execute this trajectory via the Command menu making sure that Include Sinusoidal Disturbance is checked. Plot the data to find a disturbance response similar to Figure 3.2-11. Note that the response is greater at x_3 , and that the x_1 and x_3 responses are roughly 90° out of phase.



Figure 3.2-11 Disturbance Transmission During Regulation At Encoders 1 & 3**Step 2: Disturbance Transmission During Tracking**

Enter the Disturbance Configuration dialog box and select Sinusoidal (time). Specify an amplitude of 0.25 V, frequency of 3 Hz, and 10 Repetitions. Return to Trajectory Configuration and setup a Ramp input with *distance* = 2000 counts, *velocity* = 2000 counts/s, *dwell time* = 10 ms and *no. of repetitions* = 1. Execute this shape and plot the data to find a response similar to Figure 3.2-12. The effect of the disturbance is clearly seen in the Encoder 3 output.

You may wish to try other combinations of disturbances and trajectories and at the mass carriages. Viscous friction may be applied simultaneously with disturbances.

**Figure 3.2-12 Disturbance Transmission at Encoder 3 During Tracking**

3.3 Nonintuitive System Behavior

An explanation of system functionality which may be nonintuitive to first time (or first few time) users is given below.

- Point 1:** Upon PC power up, the control loop is closed but all gains are cleared to zero. As a result, even though the status display indicates CLOSED LOOP, the control effort is zero meaning there is no force command to the actuator. With zero gains, the closed loop system behaves like an open loop one. This initialization of the gains to zero also takes place every time the real-time controller is Reset under the Utility menu
- Point 2** Even though the parameters for a particular control algorithm may exist within the Executive, the real-time Controller does not carry them until the Implement Algorithm button of the Setup Control Algorithm dialog box is activated. Then control is implemented immediately
- Point 3-** The **Limit Exceeded** message comes on as a result of either motor over-speed, a travel limit^[22], or drive over-current. In either case the message does not clear until a stable set of control gains are downloaded and if necessary a less abrupt trajectory is requested (see Section 2.1.3). This condition may also occur when the following takes place. With the control loop closed, the power to the Control Box is turned off and the mass carriages are moved manually and then the power is turned back on again. In this case the sudden motion of the carriages may cause the limiting conditions.

4. Real-Time Control Implementation

A functional overview of the control system is shown in Figure 4.0-1. This system is comprised of three subsystems: The mechanism including motor and sensors, the real-time controller / drive electronics, and the user/system (“Executive”) interface software.

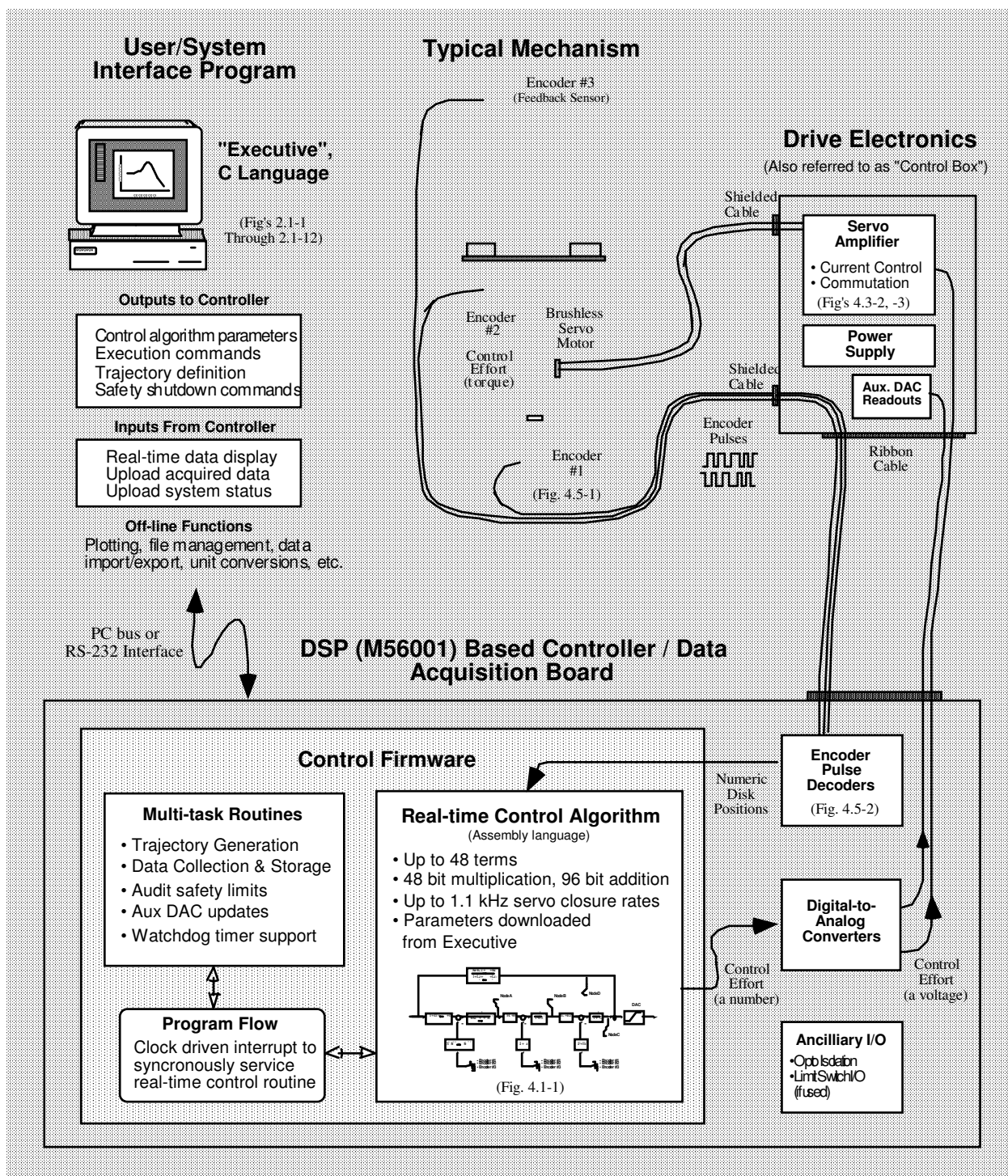


Figure 4.0-1. Overview of Real-time Control System.
This architecture is consistent with modern industrial control implementation.

An brief survey of the system architecture is afforded by tracing the data flow as the system is operated. The user specifies the control algorithm in the Executive program and downloads it (via "Implement Algorithm") to the DSP based real-time controller board. The DSP immediately executes the algorithm at the specified sample rate. This involves reading the reference input^[23] and feedback sensor (optical encoders) values, computing the algorithm, and outputting the digital control effort signal to the digital-to-analog converter (DAC).

The DAC converts the resulting stream of digital words to an analog voltage which is transformed to a current by the servo amplifier and then to a torque by the motor. The rack and pinion drive converts the torque to a force. The mechanism transforms the motor input to motion at the desired output according to the plant dynamics (i.e. equations of motion). These plant outputs are sensed by the encoders which output a stream of pulses^[24]. The pulses are decoded by a counter on the DSP board and made available as a digital position word to the real-time control algorithm.

When the user specifies a trajectory and subsequently commands the system to “Execute” the maneuver, the trajectory parameters are downloaded to the controller board. The DSP generates corresponding reference input values for use by the real-time control algorithm. Throughout the maneuver, any data specified by the user is captured and stored in memory on the board. On completion of the maneuver, the data is uploaded to PC memory where it is available for plotting and storage.

Details of these and other significant system functions are given in the remainder of this chapter.

4.1 Servo Loop Closure

Servo loop closure involves computing the control algorithm at the sampling time. The real-time Controller executes the General Form equation of the control law at each sample period T_s . This period can be as short as 0.000884 seconds (approx. 1.1 KHz) or any multiple of this number^[25]. The Executive program's Setup Control Algorithm dialog box allows the user to alter the sampling period. All forms of control laws are automatically translated by the Executive program to the General Form prior to downloading ("implementing") on the Controller. The General Form uses 96-bit real number (48-bit integer and 48-bit fractional) arithmetic for computation of the control effort. The control effort is saturated in software at +/- 16384 to represent +/- 5 volts on the 16-bit DACs whose range is +/- 10 volts. The +/-5 volt limitation is due to the actuator 's amplifier input voltage scaling.

Referring to Figure 4.1-1, the control equations are as follows:

$$R(q^{-1})\text{nodeA}(k) = T(q^{-1}) * c_p(k) - S(q^{-1}) * f_{b1}(k)$$

Here T , S , and R are seventh order polynomials of the unit time shift operator q , and the letter k represents the k^{th} sampling period for $k=0, 1, 2, \dots$. The variable nodeA is an intermediate value

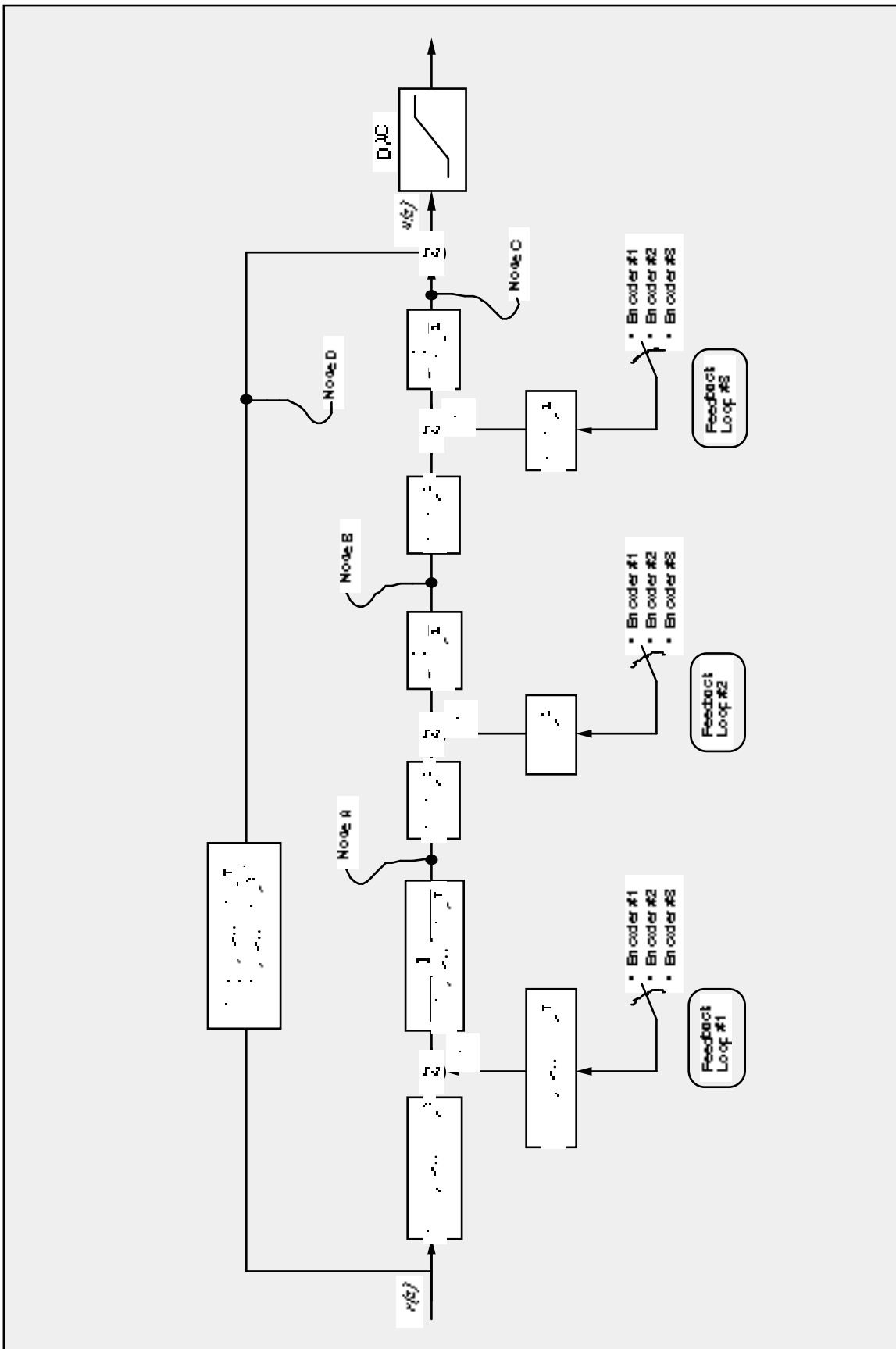


Figure 4.1-1. General Form Of The ECP Control Algorithm

of the overall control law equation which is stored in the memory and may be acquired by the user through the Data Acquisition feature of the Executive program. The variable $c_p(k)$ is the commanded position which is generated by the real-time Controller as described in the next section. The variable $f_{b1}(k)$ may be from any of the three possible feedback sensors depending on the state of software loop switch 1.

Next the intermediate loop is computed as follows

$$J(q^{-1}) * \text{nodeB}(k) = H(q^{-1}) * \text{nodeA}(k) - I(q^{-1}) * f_{b_2}(k)$$

Here again nodeB is an intermediate value stored in the memory and f_{b_2} is the sensor feedback selected via loop switch 2. J, H, and I are second order polynomials. For the inner loop we have

$$G(q^{-1}) * \text{nodeC}(k) = E(q^{-1}) * \text{nodeB}(k) - F(q^{-1}) * f_{b_3}(k)$$

NodeC is the contribution to the value of the control effort generated by the overall regulator and f_{b_3} is the sensor feedback selected via loop switch 3. G, E, and F are second order polynomials. For the feedforward loop we have

$$L(q^{-1}) * \text{nodeD}(k) = K(q^{-1}) * c_p(k)$$

In this case nodeD is the contribution to the value of the control effort generated by the feedforward terms, L and K are sixth order polynomials.

The combined regulatory and tracking controller generates the control effort as:

$$\text{control effort}(k) = \text{nodeC}(k) + \text{nodeD}(k)$$

The general control structure described above supports the implementation of a broad range of specific control forms.

4.2 Command Generation

Command generation is the real-time generation of motion trajectories specified by the user. The parameters of these trajectories are downloaded to the real-time Controller through the Executive program via the Trajectory Configuration dialog box. This section describes the trajectories generated in the current control version.

4.2.1 Step Move

Figure 4.2-1a shows a step move demand. The desired trajectory for such a move can be described by

$$\begin{aligned} c_p(t) &= c_p(0) + C && \text{for } t > 0 \\ c_v(t) &= 0 && \text{for } t > 0 \\ c_v(0) &= \infty \end{aligned}$$

Where $c_p(t)$ and $c_v(t)$ represent commanded position and velocity at time t respectively and C is the constant step amplitude. Such a move demand generates a strong impulsive force from the control actuator. The response of a mechanical system connected to the actuator would depend on the dynamic characteristics of the controller and the system itself. However, in a step move, the instantaneous velocity and its derivatives are not directly controllable. Usually step moves are used only for test purposes; more gentle trajectories are nearly always used for practical maneuvers.

4.2.2 Ramp Move

A ramp demand is seen in Figure 4.2-1b. The trajectory can be described by

$$\begin{aligned}c_p(t) &= c_p(0) + V*t && \text{for } t > 0 \\c_v(t) &= V && \text{for } t > 0 \\c_a(0) &= \infty\end{aligned}$$

where $c_a(0)$ represents commanded acceleration at time zero and V is a constant velocity. Relative to a step demand, a ramp demand is more gentle, however the acceleration is still impulsive. The commanded velocity is a known constant during the maneuver.

4.2.3 Parabolic Move

Figure 4.2-1c shows a parabolic move demand. Its trajectory can be expressed as:

$$\begin{aligned}c_p(t) &= c_p(0) + c_v(0)*t + 1/2 A*t^2 && \text{for } t > 0 < 1/2 t_f \\c_v(t) &= c_v(0) + A*t && \text{for } t > 0 < 1/2 t_f \\c_a(t) &= A && \text{for } t > 0 < 1/2 t_f \\c_j(0) &= \infty\end{aligned}$$

where $c_j(t)$ represents commanded jerk at time t and A is a constant acceleration, and t_f is the final destination time. Relative to a ramp demand, a parabolic demand is more gentle, however the rate of change of acceleration (jerk) is still impulsive. Note that the commanded acceleration is a known constant during the maneuver. The second half of a parabolic demand uses $-A$ for deceleration.

4.2.4 Cubic Move

Figure 4.2-1d shows a cubic demand which can be described by

$$\begin{aligned}c_p(t) &= c_p(0) + c_v(0)*t + 1/2 c_a(0)*t^2 + 1/6 J*t^3 && \text{for } t > 0 < 1/4 t_f \\c_v(t) &= c_v(0) + c_a(0)*t + 1/2 J*t^2 && \text{for } t > 0 < 1/4 t_f \\c_a(t) &= c_a(0) + J*t && \text{for } t > 0 < 1/4 t_f \\c_j(0) &= J\end{aligned}$$

where J represents a constant jerk. Relative to all the above demands, a cubic demand is more gentle. The commanded acceleration is linearly changing during the three sections of the maneuver. The second half of a cubic demand uses $-J$ and the third part uses J again for the jerk input.



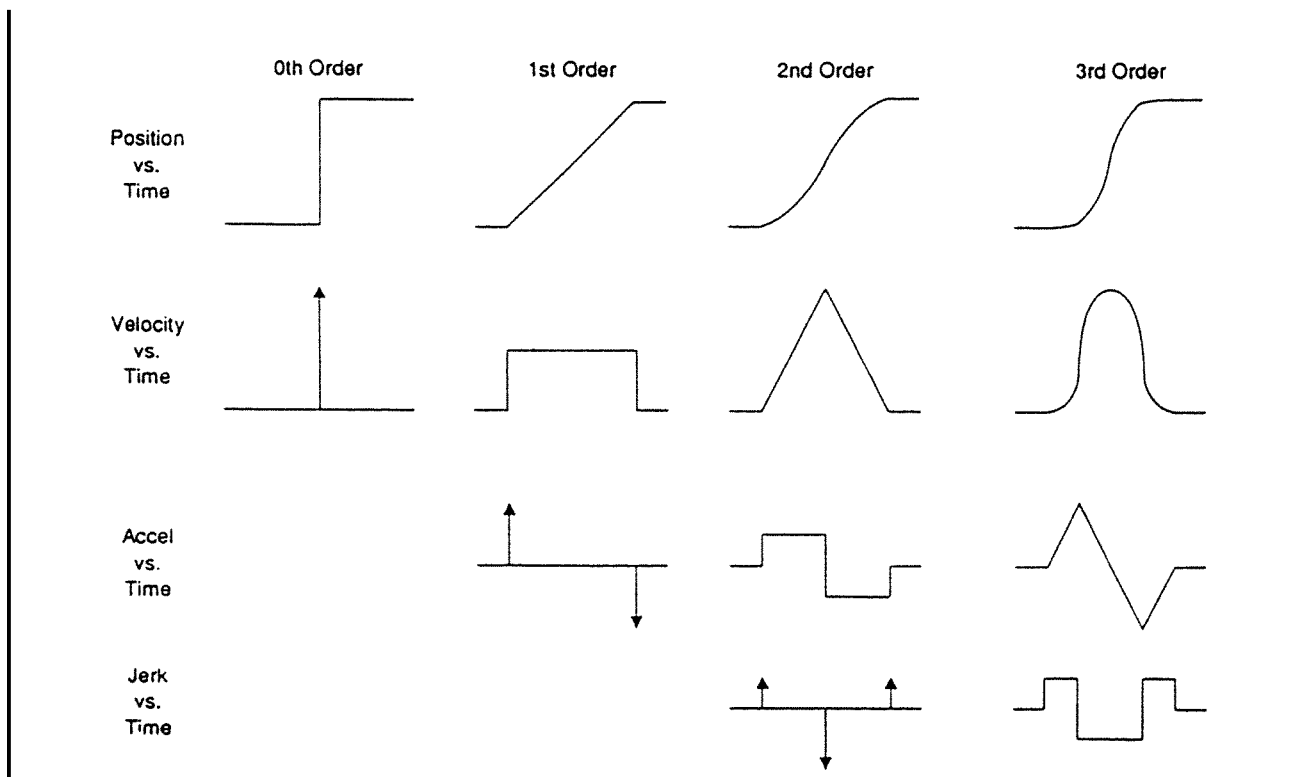


Figure 4.2-1. Geometric Command Trajectories Of Increasing Order

4.2.5 The Blended Move

Any time a ramp, a parabolic or a cubic trajectory move is demanded, the real-time Controller executes a general blended move to produce the desired reference input to the control algorithm. The move is broken into five segments as shown in the velocity profile of Figure 4.2-2. For each section a cubic (in position) trajectory is planned. Five distinct cubic equations can describe the forward motion. After the dwell time, the reverse motion can be described by five more cubic trajectories. Each cubic has the form:

$$c_{p_i}(t) = c_{p_i}(0) + V_i * t + 1/2 A_i * t^2 + 1/6 J_i * t^3 \quad i = 1, \dots, 5$$

Using a known set of trajectory data (i.e. the requested total travel distance, acceleration time t_{acc} , and the maximum speed v_{max} , for each move), the constant coefficients V_i , A_i , and J_i are determined for each segment of the move by the real-time Controller. This function is known as the "motion planning" task. Note that for a parabolic profile $J_i=0$, and for a ramp profile A_i is also zero which further simplifies the task. Having determined the coefficients for each section, the real-time Controller uses these values at the servo loop sampling periods to update the commanded position (reference input). For example if the segment is a cubic ($J \neq 0$):

$$c_a(k) = c_a(k-1) + J * T_s$$

$$c_v(k) = c_v(k-1) + c_a(k) * T_s$$

$$c_p(k) = c_p(k-1) + c_v(k) * T_s$$

where T_s is the sampling period and $c_a(k)$, $c_v(k)$, $c_p(k)$ represent commanded acceleration, velocity and position at the k^{th} sampling period.

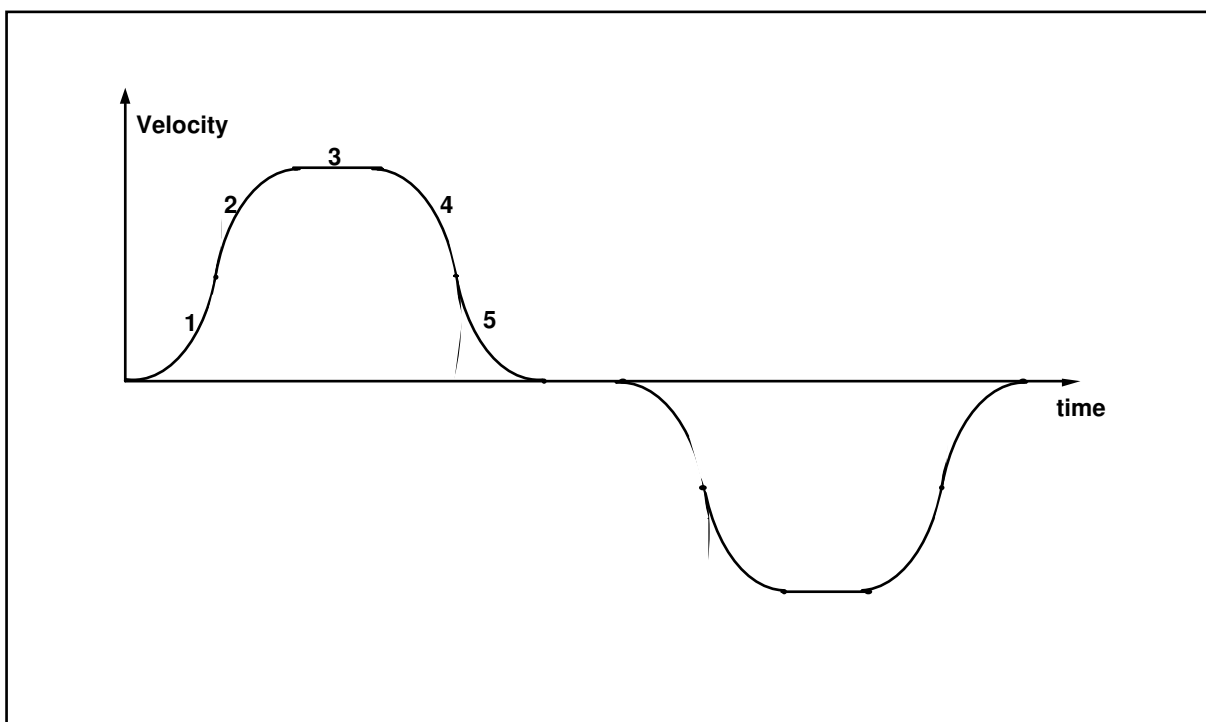


Figure 4.2-2 Velocity Profile for General Blended Move

4.2.6 Sinusoidal Move

The sinusoidal move is generated using the following equation:

$$c_p(k) = R * \sin(\theta(k))$$

where R is the amplitude, $\theta(k) = \omega * k * T_p$ for $k=0,1,\dots$, and ω is the commanded frequency in Hz. T_p is set to five milliseconds. To further smooth out the trajectory, a cubic spline is fitted between the points as follows:

$$c_p'(k) = (c_p(k-1) + 4 * c_p(k) + c_p(k+1)) / 6$$

For the linear sine sweep, $\omega(k) = \alpha * T_p$, where α is a constant determined by the difference between the maximum and the minimum frequency divided by the sweep time

$$\alpha = (\omega_{\max} - \omega_{\min}) / \text{sweep time}$$

4.3 Brushless Motor Commutation and Torque Control

The main advantage of a DC brushless motor (otherwise known as a *permanent magnet synchronous* motor) over the conventional DC brush motor is the elimination of both brush friction and associated wear.

Figure 4.3-1 shows a cross sectional view of a typical DC brushless motor. In contrast to the conventional DC brush motor, the permanent magnets are fixed to the rotor. The phase windings (typically 3 phases) are distributed in slots of the stator. This arrangement also provides for greater heat dissipation (i^2R), which in turn leads to improved life and typically greater volume-to-power ratios for brushless motors than for brush motors.

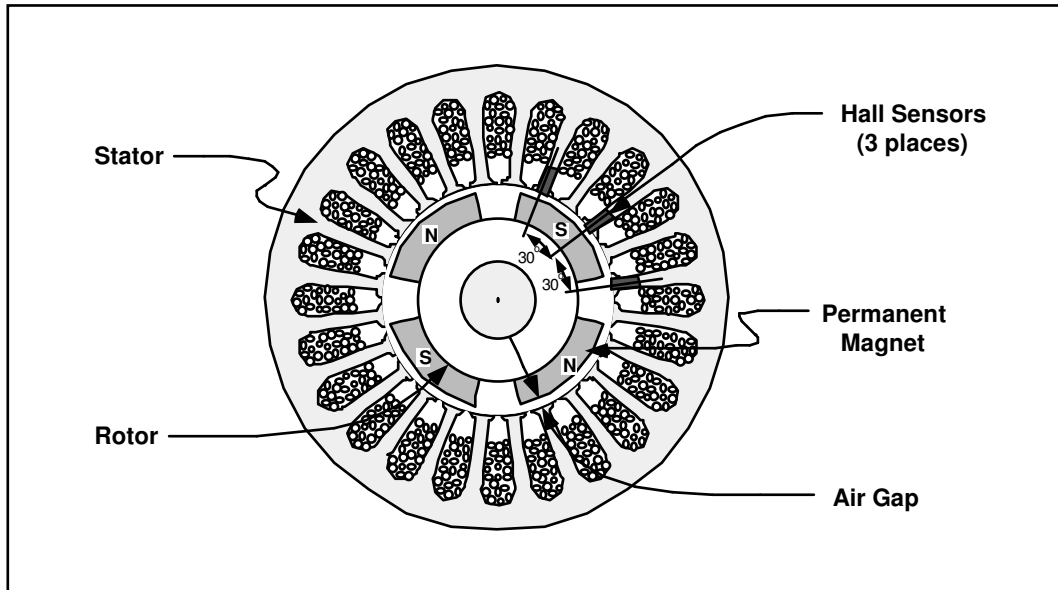


Figure 4.3-1. Cross-section of a Typical DC Brushless Motor. (Four pole type shown)

In any continuously rotating motor, to provide continuous torque, the current must be successively altered or switched depending on the absolute position of the rotor. In a DC brush motor, this switching is implemented mechanically by the commutator. In a DC brushless motor, a rotor positioning sensor is used and the commutation procedure is done electronically. We will consider two different types of commutation procedures for DC permanent magnet brushless motors: rectangular and sinusoidal.

4.3.1 Rectangular (Hall-Effect) Commutation

Figure 4.3-2 shows a simplified schematic of the drive system(s) used in this mechanism. This is a typical drive scheme for a 4-pole 3-phase star-wound brushless motor with Hall-effect (rectangular) commutation. The three Hall-effect sensors are positioned on the stator. The sensed magnetic field switches between the adjacent north/south poles as the rotor rotates. This switching sequence is then used to direct the commanded motor current to individual winding phases (Figure 4.3-3). A simple electronic logic circuit is used for the generation of the switching steps (six steps per electrical cycle or 12 steps per mechanical cycle for a 4-pole motor). Analog high bandwidth *proportional plus integral* (PI) current feedback loops are then used on two of the three

phases to assure almost instantaneous response of the actual winding current following any commanded current changes. The third phase does not require a current feedback loop because (as per Kirchhoff's law) the current in the third phase is the negative of the sum of the current in the other two phases.

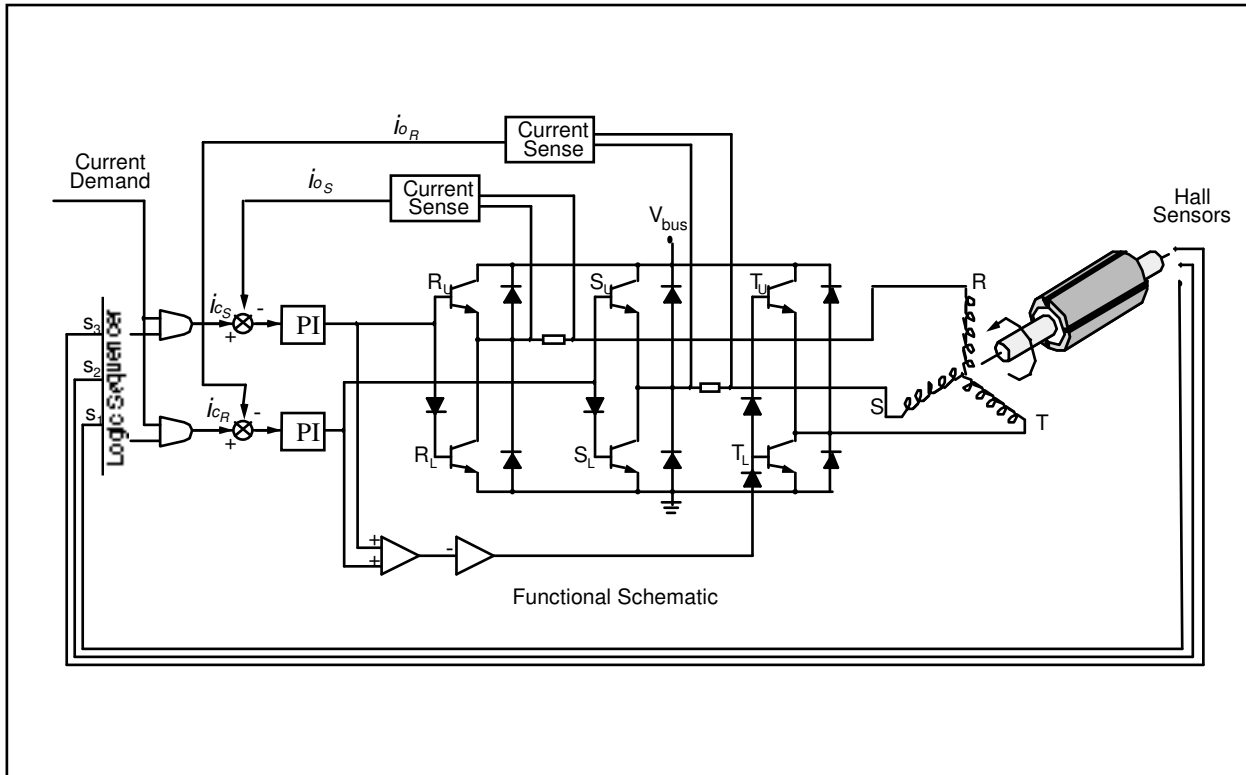


Figure 4.3-2 Simplified Schematic of Hall-effect Commutated DC Brushless Motor Drive System

We assume that, due to the relatively high bandwidth of the current loops, nearly instantaneous matching of the commanded current (torque) and the actual current occurs.^[26] With the rectangular drive scheme shown in Figure 4.3-3b, at any position only two of the three phases are operational so that the current in the third phase is zero. Consider a 60 electrical degree interval $0 \leq \theta \leq 60$. The torque input from phase R is

$$T_R = K_t I_R \sin\theta \quad (4.3-1)$$

and

$$I_R = -I_T \text{ with } I_S = 0$$

where $K_t = Z \cdot B_p \cdot r \cdot l$ is the torque constant per phase in N-m. Z is the number of turns per winding, r is the inside radius of the stator in meters, and l is the active length of conductors in meters. B_p is the peak air gap flux density in Teslas.

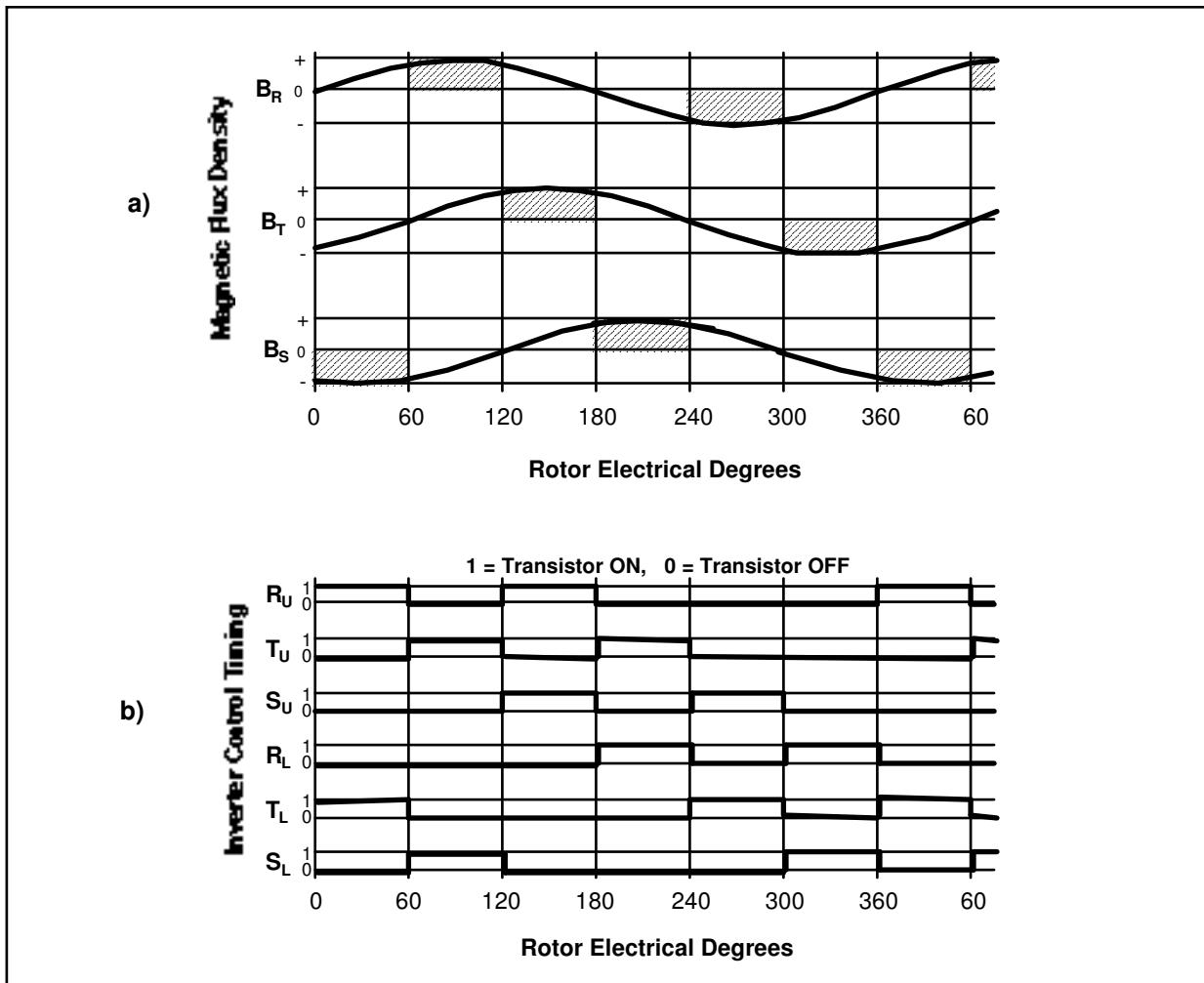


Figure 4.3-3 Hall-effect Commutation Timing Diagram

To produce positive torque, the flux density must be negative for phase T. This is the case for a sinusoidally wound motor (see Figure 4.3-3a):

$$\begin{aligned} T_T &= K_t I_R \sin(\theta+240) \\ &= K_t I_R (0.5\sin\theta+0.886\cos\theta) \quad \text{for } 0 \leq \theta \leq 60 \end{aligned} \quad (4.3-2)$$

The total torque is given by the addition of T_T and T_R over the interval $0 \leq \theta \leq 60$:

$$T = T_R + T_T = K_t (1.5\sin\theta + 0.886\cos\theta) I_R \quad (4.3-3)$$

Equation 4.3-3 shows that the effective torque constant, $K_t(1.5\sin\theta+0.886\cos\theta)$, changes as a function of rotor angle producing a torque ripple as much as 13% for rectangular commutation. This torque ripple can be treated

as a disturbance which, in most practical applications, is reduced by closing outer velocity and position loops around the current loop.

4.3.2 Sinusoidal Commutation

For a motor with sinusoidally wound stator coils, the air gap flux densities are:

$$\begin{aligned} B_R &= B_p \sin \theta \\ B_S &= B_p \sin(\theta+120) \\ B_T &= B_p \sin(\theta+240) \end{aligned} \quad (4.3-4)$$

Where B_R , B_S , and B_T are the per phase flux densities for the motor phases R, S, and T respectively and B_p is the peak value in Tesla. If we assume that the PI controllers around the current loop maintain the phase currents in phase with the air gap flux densities, then the current in each phase in terms of the peak current I_p are given by

$$\begin{aligned} I_R &= I_p \sin \theta \\ I_S &= I_p \sin(\theta+120) \\ I_T &= I_p \sin(\theta+240) \end{aligned} \quad (4.3-5)$$

From equation 4.3-4 and 4.3-5, the instantaneous torque being produced in each phase is

$$\begin{aligned} T_R &= K_t I_p \sin^2 \theta \\ T_S &= K_t I_p \sin^2(\theta+120) \\ T_T &= K_t I_p \sin^2(\theta+240) \end{aligned} \quad (4.3-6)$$

The total torque is the sum of the above torque components, which is given by

$$\begin{aligned} T &= T_R + T_S + T_T \\ &= K_t I_p (\sin^2 \theta + \sin^2(\theta+120) + \sin^2(\theta+240)) \\ &= \frac{3}{2} K_t I_p^2 \end{aligned} \quad (4.3-7)$$

Equation 4.3-7 shows that "perfect" sinusoidal commutation of a "perfectly" sinusoidally wound motor provides for a ripple free torque constant. In practice, however, a small amount of torque ripple may exist due to imperfections in the windings and the current loop elements and misalignment of the rotor position sensor. Note that for this method of commutation, a high resolution absolute position sensor is required.

4.3.3 Proportional Plus Integral (PI) Current Loop

Each phase of the motor winding coil is characterized by its resistance R and its inductance L . If we assume that the per-phase back emf of motor is negligible in the high torque / low speed region of operation, then the block diagram shown in Figure 4.3-4 represents the per-phase PI loop for two of the three phases. (There is no need for PI loop on the third phase, as the sum of the total current must be zero in a star winding circuit).

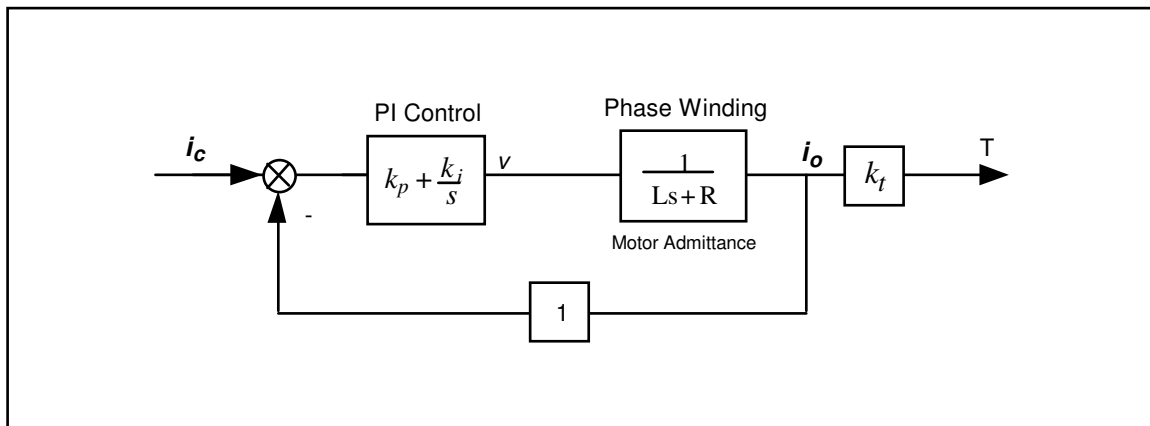


Figure 4.3-4 Simplified Block Diagram of Analog PI Controller (applies to each motor winding phase)

This is a second order system with the transfer function

$$\frac{i_o(s)}{i_c(s)} = \frac{(k_p/L)s + k_i/L}{s^2 + (R + k_p/L)s + k_i/L} \quad (4.3-8)$$

having natural frequency

$$\omega_n = \sqrt{\frac{k_i}{L}} \quad (4.3-9)$$

and damping ratio

$$\zeta = \frac{k_p + R}{2\sqrt{k_i L}} \quad (4.3-8)$$

By choosing the appropriate values for k_i and k_p , very high bandwidths for closed loop current (torque) response may be achieved (>500 Hz). In comparison to the achievable bandwidths for the outer velocity and/or position loops, the bandwidth of the current loop is generally about two orders of magnitude greater. This means that the current (torque) response is essentially instantaneous and therefore its dynamics may usually be ignored. Note also that the addition of the integrator has made this closed loop system a type one system. This, in turn, results in zero steady state error and a unity DC gain.

4.4 Multi-Tasking Environment

Digital control implementation is intimately coupled with the hardware and software that supports it. Nowhere is this more apparent than in the architecture and timing used to support the various data processing routines. A well prioritized time multi-tasking scheme is essential to maximizing the performance attainable from the processing resources.

The priority scheme for the ECP real-time Controller's multi-tasking environment is tabulated in Table 4.4-1. The highest priority task is the trajectory update and servo loop closure computation which takes place at the

maximum rate of 1.131 KHz (minimum sampling period is 0.000884 seconds). In this case, the user may reduce the sampling rate through the Executive Program via changes to T_s in the Setup Control Algorithm dialog box.

The trajectory planning task has the third highest priority and is serviced at a maximum rate of 377 Hz. Here the parameters for a new trajectory need not be calculated every time this task is serviced by the real-time Controller. Whenever a new trajectory is required (i.e. the current trajectory is near its completion) this task is executed. The lower priority tasks are system house keeping routines including safety checks, interface and auxiliary analog output.

Table 4.4-1 The Multi-Tasking Priority Scheme of the Real-Time Controller

Priority	Task Description	Service Frequency
1	Servo Loop Closure & Command Update	1.1 KHz
2	Trajectory Planning	377 Hz
3	Background Tasks including User Interface, Auxiliary DAC Update, Limit checks etc.	Background (In time between other tasks)

The higher priority tasks always prevail over lower ones in obtaining the computational power of the DSP. This multi-tasking scheme is realized by a real-time clock which generates processor interrupts.

4.5 Sensors

There are four incremental rotary shaft encoders used in the 3 mass version, Model 210a. (Three for Model 210). Three of these encoders are used to sense the position of the three masses. Each has a resolution of 4000 pulses per revolution. The fourth encoder is directly coupled to the brushless motor and is only used for commutation purposes. It has an output of 1000 pulses per revolution.

The encoders are all optical type whose principle of operation is depicted in Figure 4.5-1. A low power light source is used to generate two 90 degrees out of phase sinusoidal signals on the detectors as the moving plate rotates with respect to the stationary plate. These signals are then squared up and amplified in order to generate quadrate logic level signals suitable for input to the programmable gate array on the real-time Controller. The gate array uses the A and B channel phasing to decode direction and detects the rising and falling edge of each to generate 4x resolution – see Figure 4.5-2. ^[27] The pulses are accumulated continuously within 24-bit counters (hardware registers). The contents of the counters are read by the DSP once every servo (or commutation) cycle time and extended to 48-bit word length for high precision numerical processing.

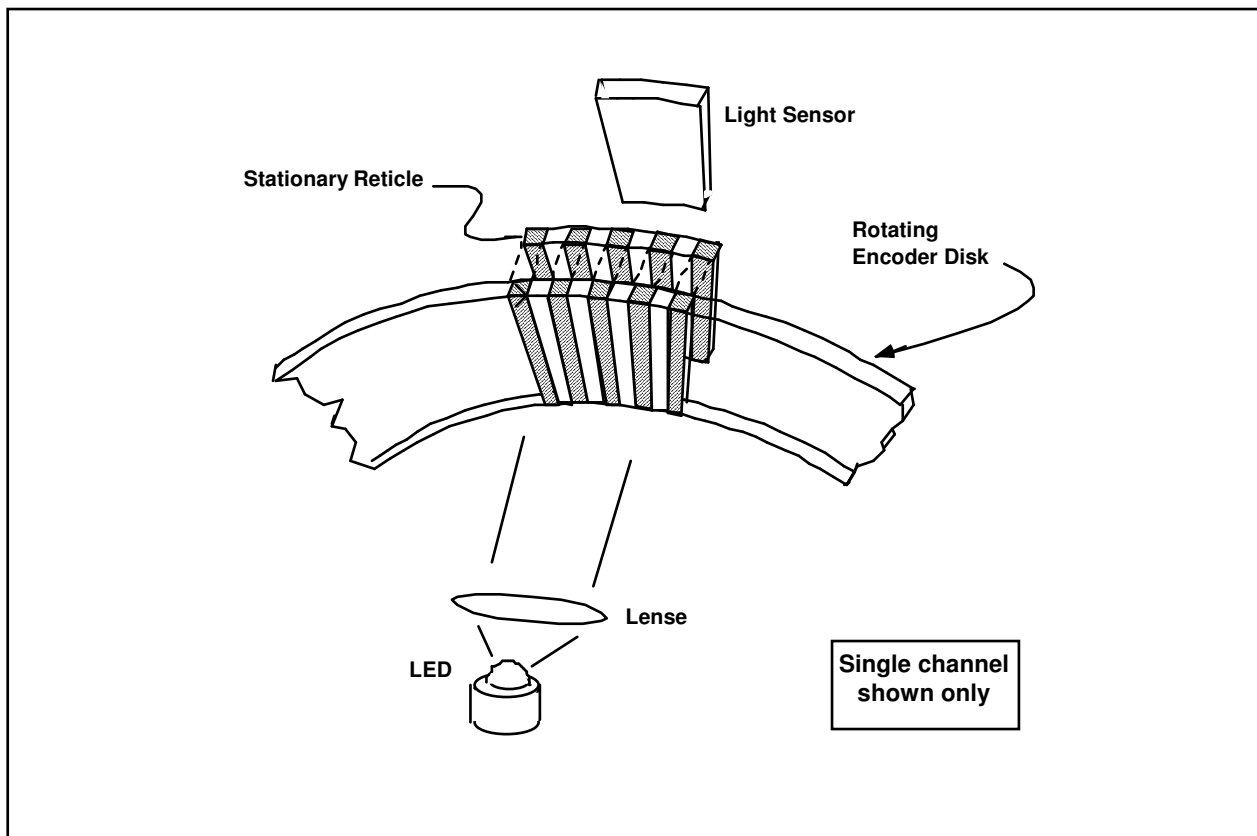


Figure 4.5-1 The Operation Principle of Optical Incremental Encoders

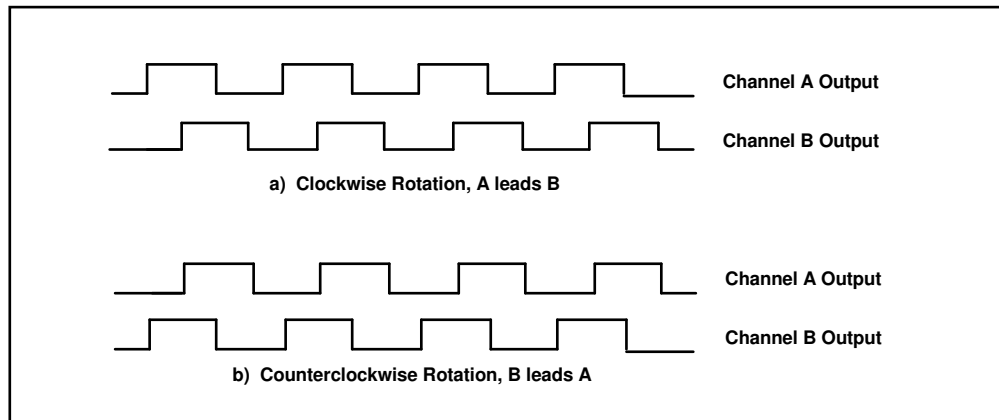


Figure 4.5-2. Optical Encoder Output

4.6 Auxiliary Analog Output (system option)

There are two analog output channels in the Control Box which are connected to two 16-bit DACs which physically reside on the real-time Controller. Each analog output has the range of +/- 10 volts (-32768 to +32767 counts) with respect to the analog ground. The outputs on these DACs are updated by the real-time Controller as a low priority task. However, for virtually all trajectories (e.g. for sine sweep up to approx. 25 Hz) the update rate is sufficiently fast for an oscilloscope or other analog equipment to inspect the various internal Controller signals. See the section on the Executive Program's Utility menu for the available signals to output on these DACs.

5. Plant Dynamic Models

This Chapter provides time and Laplace domain expressions which are useful for linear control implementation and are used in the experiments described later in this manual. Details of the development of these equations as well as their motivation and additional equation forms are provided in Appendix A.

5.1 Two Degree of Freedom Plants

The most general form of the two degree of freedom plant is shown in Figure 5-1a where friction is idealized as being viscous. Using the free body diagram of Figure 5-1b and summing forces acting on m_1 we have via Newton's second law:

$$m_1\ddot{x}_1 + c_1\dot{x}_1 + (k_1 + k_2)x_1 - k_2x_2 = F(t) \quad (5.1-1)$$

Similarly from Figure 5-1c for m_2 :

$$m_2\ddot{x}_2 + c_2\dot{x}_2 + (k_2 + k_3)x_2 - k_2x_1 = 0 \quad (5.1-2)$$

These may be expressed in a state space realization as:

$$\begin{aligned} \dot{x} &= Ax + BF(t) \\ Y &= Cx \end{aligned} \quad (5.1-3)$$

where:

$$X = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(k_1+k_2)/m_1 & -c_1/m_1 & k_2/m_1 & 0 \\ 0 & 0 & 0 & 1 \\ k_2/m_2 & 0 & -(k_2+k_3)/m_2 & -c_2/m_2 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & C_4 \end{bmatrix}$$

and $C_i = 1$ ($i = 1, 2, 3, 4$) when X_i is an output and equals 0 otherwise.

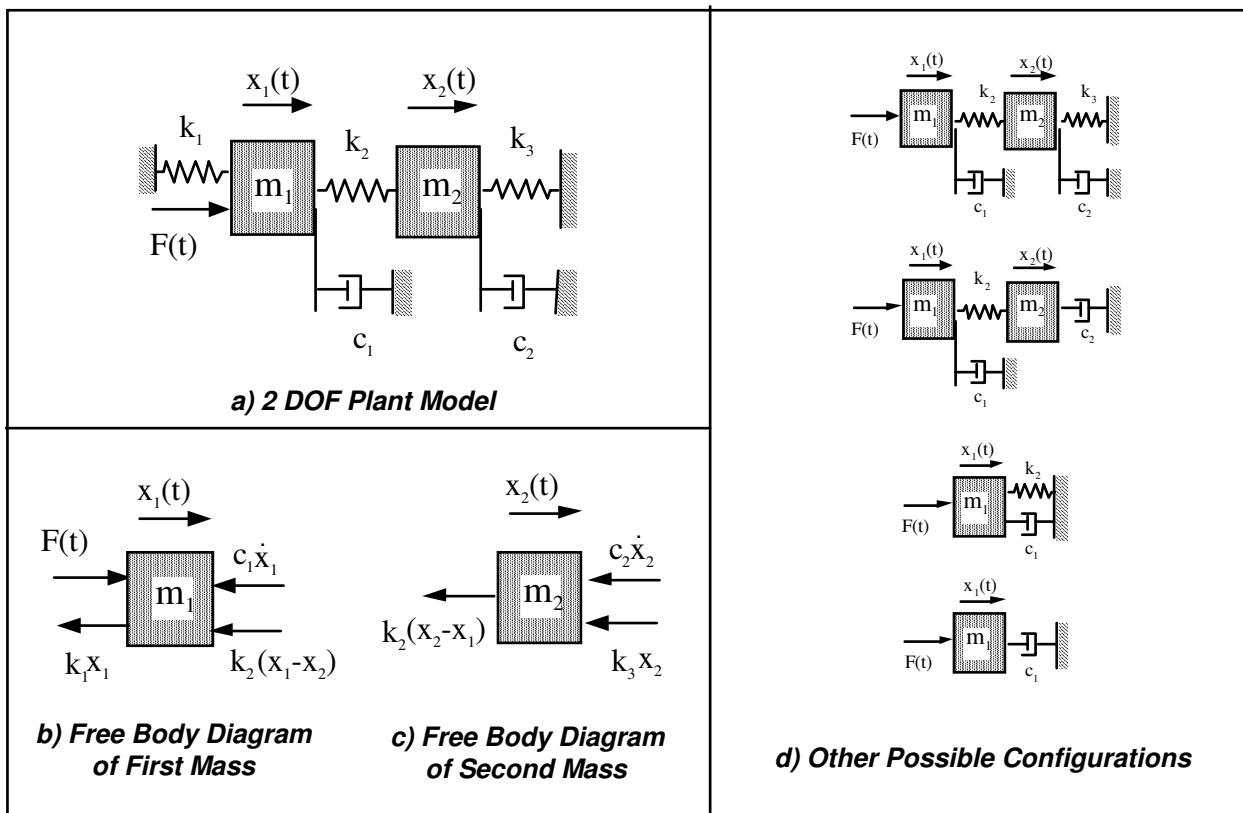


Figure 5.1-1 Two DOF Plant Models

By Laplace transform of Eq's (5.1-1,-2) and assuming zero valued initial conditions we may solve for the transfer functions:

$$\frac{x_1(s)}{F(s)} = \frac{m_2 s^2 + c_2 s + k_2 + k_3}{D(s)} \quad (5.1-4)$$

$$\frac{x_2(s)}{F(s)} = \frac{k_2}{D(s)} \quad (5.1-5)$$

where:

$$D(s) = m_1 m_2 s^4 + (m_1 c_2 + m_2 c_1) s^3 + (m_1(k_2 + k_3) + m_2(k_1 + k_2) + c_1 c_2) s^2 + (c_1(k_2 + k_3) + c_2(k_1 + k_2)) s + k_1 k_2 + k_1 k_3 + k_2 k_3 \quad (5.1-6)$$

which may also be expressed in the form:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (5.1-7)$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (5.1-8)$$

where the ω_p 's and ζ 's are the natural frequencies and damping ratios respectively, and the gains K_1 & K_2 , are nominally equal to $1/m_1$ and $k_1/m_1 m_2$ (but often may be measured more directly).

For the case $k_1 = k_3 = 0$ ($\omega_1 = 0$), a damped rigid body motion exists and Eq's (5.1-7,-8) become:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{s(s + c^*) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (5.1-9)$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2}{s(s + c^*) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (5.1-10)$$

The equations describing the single DOF plant may be obtained from the above or found in Appendix A.

5.2 Three Degree of Freedom Plants (Model 210a only)

The time domain equations of motion for the three DOF rectilinear plant are (see Appendix A, Figure A-4):

$$m_1 \ddot{x}_1 + c_1 \dot{x}_1 + (k_1 + k_2)x_1 - k_2 x_2 = F(t) \quad (5.2-1)$$

$$m_2 \ddot{x}_2 + c_2 \dot{x}_2 + (k_2 + k_3)x_2 - k_2 x_1 - k_3 x_3 = 0 \quad (5.2-2)$$

$$m_3 \ddot{x}_3 + c_3 \dot{x}_3 + (k_3 + k_4)x_3 - k_3 x_2 = 0 \quad (5.2-3)$$

which may be expressed in a state space realization:

$$\begin{aligned} \dot{x} &= Ax + BF(t) \\ Y &= Cx \end{aligned} \quad (5.2-4)$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -(k_1 + k_2)/m_1 & -c_1/m_1 & k_2/m_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ k_2/m_2 & 0 & -(k_2 + k_3)/m_2 & -c_2/m_2 & k_3/m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & k_3/m_3 & 0 & -(k_3 + k_4)/m_3 & -c_3/m_3 \end{bmatrix},$$

$$X = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \\ x_3 \\ \dot{x}_3 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} C_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_6 \end{bmatrix}$$

with the output notation as in Eq (5.1-3).

The Laplace transformation of Eq's (5.2-1, -2 & -3) yields (zero valued initial conditions):

$$\frac{x_1(s)}{F(s)} = \frac{N_1(s)}{D(s)} \quad (5.2-6)$$

$$\frac{x_2(s)}{F(s)} = \frac{N_2(s)}{D(s)} \quad (5.2-7)$$

$$\frac{x_3(s)}{F(s)} = \frac{N_3(s)}{D(s)} \quad (5.2-8)$$

where:

$$N_1(s) = m_2 m_3 s^4 + (m_2 c_3 + m_3 c_2) s^3 + (m_2(k_3 + k_4) + m_3(k_2 + k_3) + c_2 c_3) s^2 + (c_2(k_3 + k_4) + c_3(k_2 + k_3)) s + k_2 k_3 + k_2 k_4 + k_3 k_4 \quad (5.2-9)$$

$$N_2(s) = k_2 (m_3 s^2 + c_3 s + k_3 + k_4) \quad (5.2-10)$$

$$N_3(s) = k_2 k_3 \quad (5.2-11)$$

$$D(s) = m_1 m_2 m_3 s^6 + [m_1(m_2 c_3 + m_3 c_2) + m_2 m_3 c_1] s^5 + [m_1(m_2(k_3 + k_4) + m_3(k_2 + k_3) + c_2 c_3) + m_2(m_3(k_1 + k_2) + c_1 c_3) + m_3 c_1 c_2] s^4 + \left[\frac{m_1(c_2(k_3 + k_4) + c_3(k_2 + k_3)) + m_2(c_1(k_3 + k_4) + c_3(k_1 + k_2))}{m_3(c_1(k_2 + k_3) + c_2(k_1 + k_2)) + c_1 c_2 c_3} \right] s^3 + \left[\frac{m_1(k_2(k_3 + k_4) + k_3 k_4) + m_2((k_1 + k_2)(k_3 + k_4)) + m_3(k_1(k_2 + k_3) + k_2 k_3)}{+ c_1 c_2(k_3 + k_4) + c_1 c_3(k_2 + k_3) + c_2 c_3(k_1 + k_2)} \right] s^2 + [c_1(k_2(k_3 + k_4) + k_3 k_4) + c_2((k_1 + k_2)(k_3 + k_4)) + c_3(k_1(k_2 + k_3) + k_2 k_3)] s + k_1 k_2 k_3 + k_1 k_2 k_4 + k_1 k_3 k_4 + k_2 k_3 k_4 \quad (5.2-12)$$

which may also be expressed as:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_{z1} \omega_{z1} s + \omega_{z1}^2) (s^2 + 2\zeta_{z2} \omega_{z2} s + \omega_{z2}^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2) (s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \quad (5.2-13)$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2) (s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \quad (5.2-14)$$

$$\frac{x_3(s)}{F(s)} = \frac{K_3}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2) (s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \quad (5.2-15)$$

where again the ω_i 's and ζ_i 's are the respective natural frequencies and damping ratios, and the ordered gains K_i are nominally equal to $1/m_1$, $k_2/m_1 m_2$ and $k_2 k_3/m_1 m_2 m_3$ respectively.

For the case when $k_1 = k_4 = 0$ (rigid body mode present):

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_{z1} \omega_{z1} s + \omega_{z1}^2) (s^2 + 2\zeta_{z2} \omega_{z2} s + \omega_{z2}^2)}{s(s + c^*) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2) (s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \quad (5.2-16)$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{s(s + c^*) (s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2) (s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \quad (5.2-17)$$

$$\frac{x_3(s)}{F(s)} = \frac{K_3}{s(s + c^*)(s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2)(s^2 + 2\zeta_{p3}\omega_{p3}s + \omega_{p3}^2)} \quad (5.2-18)$$

Parameter identification to establish the explicit plant transfer functions or state space realizations for Models 210 & 210a is covered in Section 6.1.

6. Experiments

This chapter outlines experiments which identify the plant parameters, implement a variety of control schemes, and demonstrate many important control principles. The versatility of this software / hardware system allows for a much broader range of experimental uses than will be described here and the user is encouraged to explore whatever topics and methodologies may be of interest – subject of course to your school and laboratory guidelines and the safety notations of this manual. The safety portion of this manual, Section 2.4, must be read and understood by any user prior to operating this equipment.

The instructions in this chapter begin at a high level of detail so that they may be followed without a great deal of familiarity with the PC system interface and become more abbreviated in details of system operation as the chapter progresses. To become more familiar with these operations, it is strongly recommended that the user read Chapter 2 in its entirety prior to undertaking the operations described here. Remember here, as always, it is recommended to save data and control configuration files regularly to avoid undue work loss should a system fault occur.

6.1 System Identification

This section gives a procedure for identifying the plant parameters applicable to Eq's (5.1-1 through 5.1-8). The approach will be to indirectly measure the mass, spring, and damping parameters by making measurements of the plant while set up in a pair of classical spring-mass configurations.

Procedure:

1. Clamp the second mass to put the mechanism in the configuration shown in Figure 6.1-1a using a shim (e.g. 1/4 inch nut) between the stop tab and stop bumper so as not to engage the limit switch (see Section 2.2). Verify that the medium stiffness spring (nominally 400 N/m (2.25 lb/in.)) is connecting the first and second mass carriages.
2. Secure four 500g masses on the first and second mass carriages.
3. With the controller powered up, enter the Control Algorithm box via the Set-up menu and set $T_s = 0.00442$. Enter the Command menu, go to Trajectory and select Step, Set-up. Select *Open Loop Step* and input a step size of 0 (*zero*), a duration of 3000 ms and 1 repetition. Exit to the background screen by consecutively selecting OK. This puts the controller in a mode for acquiring 6 sec of data on command but without driving the actuator. This procedure may be repeated and the duration adjusted to vary the data acquisition period.

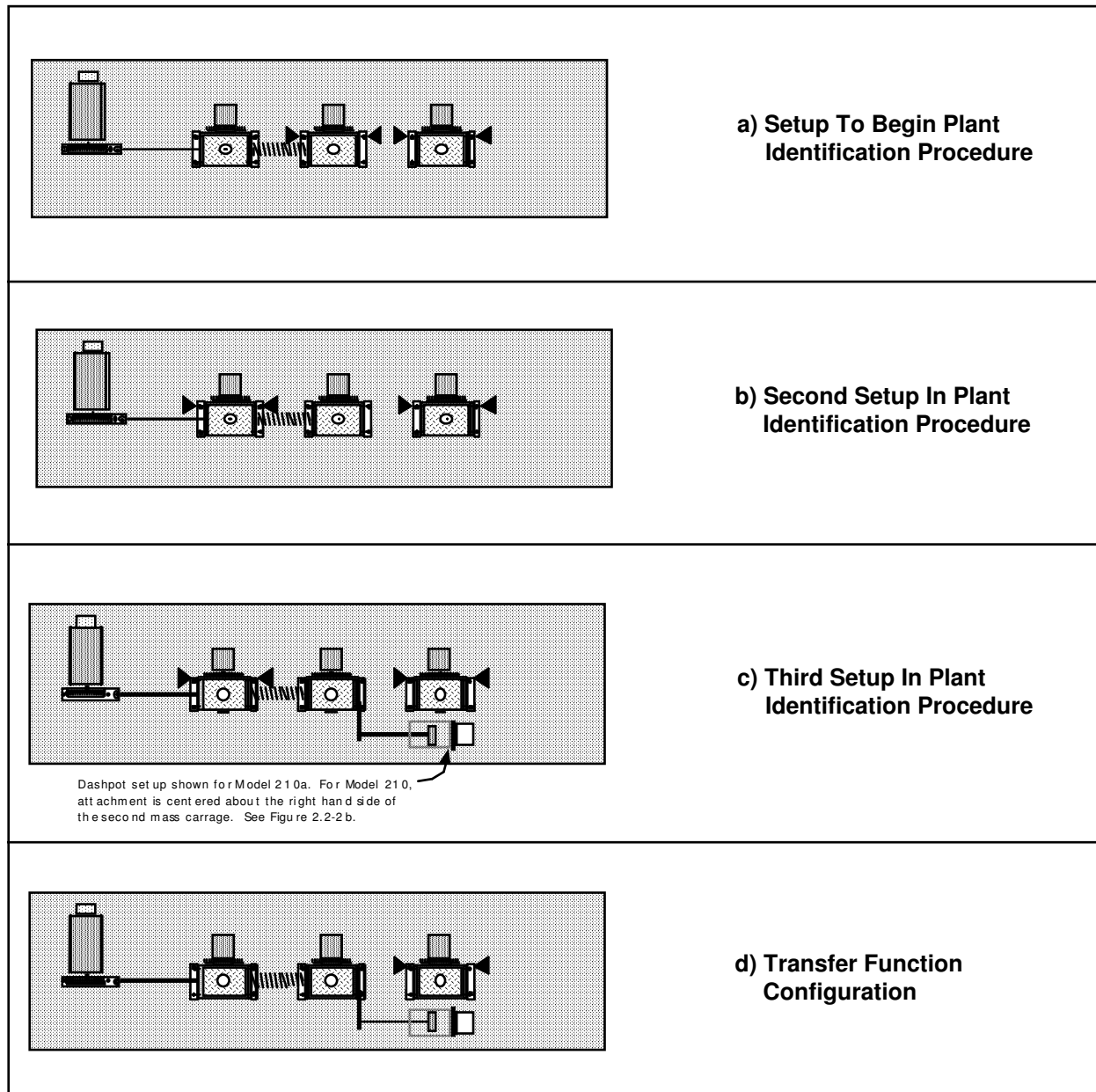


Figure 6.1-1 Configurations For Plant Identification
(Model 210a shown. Four 500 g. weights on each active carriage.)

4. Go to Set up Data Acquisition in the Data menu and select Encoder #1 and Encoder #2 as data to acquire and specify data sampling every 2 (two) servo cycles (i.e. every $2T_s$'s). Select OK to exit. Select Zero Position from the Utility menu to zero the encoder positions.
5. Select Execute from the Command menu. Prepare to manually displace the first mass carriage approximately 2.5 cm. Exercise caution in displacing the carriage so as not to engage the travel limit switch. With the first mass displaced approximately 2.5 cm in either direction, select Run from the Execute box and release the mass approximately 1 second later. The mass will oscillate and attenuate while encoder data is collected to record this response. Select OK after data is uploaded.
6. Select Set-up Plot from the Plotting menu and choose Encoder #1 Position then select Plot Data from the Plotting menu. You will see the first mass time response.
7. Choose several consecutive cycles (say ~5) in the amplitude range between 5500 and 1000 counts (This is representative of oscillation amplitudes during later closed loop control maneuvers. Much smaller amplitude responses become dominated by nonlinear friction

effects and do not reflect the salient system dynamics). Divide the number of cycles by the time taken to complete them being sure to take beginning and end times from the same phase of the respective cycles.^[28] Convert the resulting frequency in Hz to radians/sec. This *damped frequency*, ω_d , approximates the *natural frequency*, ω_n , according to:

$$\omega_{nm11} = \frac{\omega_{d_{m11}}}{\sqrt{1-\zeta_{m11}^2}} \approx \omega_{d_{m11}} \quad (\text{for small } \zeta_{m11}) \quad (6.1-1)$$

where the "m11" subscript denotes mass #1, trial #1. (Close the graph window by clicking on the left button in the upper right hand corner of the graph. This will collapse the graph to icon form where it may later be brought back up by double-clicking on it.)

8. Remove the four masses from the first mass carriage and repeat Steps 5 through 7 to obtain ω_{nm12} for the unloaded carriage. If necessary, repeat Step 3 to reduce the execution (data sampling only in this case) duration.
9. Measure the initial cycle amplitude X_o and the last cycle amplitude X_n for the n cycles measured in Step 8. Using relationships associated with the *logarithmic decrement*:

$$\frac{\zeta_{m12}}{\sqrt{1-\zeta_{m12}^2}} = \frac{1}{2\pi n} \ln\left(\frac{X_o}{X_n}\right) \rightarrow \zeta_{m12} \approx \frac{1}{2\pi n} \ln\left(\frac{X_o}{X_n}\right) \quad (\text{for small } \zeta_{m12}) \quad (6.1-2)$$

find the damping ratio ζ_{m12} and show that for this small value the approximations of Eq's (6.1-1, -2) are valid.

10. Repeat Steps 5 through 9 for the second mass carriage. Here in Step 6 you will need to remove Encoder #1 position and add Encoder #2 position to the plot set-up. Hence obtain ω_{nm21} , ω_{nm22} and ζ_{m22} . How does this damping ratio compare with that for the first mass? Be sure to save this plotted data as it will be used in the next experiment.
11. Connect the mass carriage extension bracket and dashpot to the second mass as shown in Figure 6.1-2c. Open the damping (air flow) adjustment knob 2.0 turns from the fully closed position. Repeat Steps 5, 6, and 9 with four 500 g masses on the second carriage and using only amplitudes ≥ 500 counts in your damping ratio calculation. Hence obtain ζ_d where the "d" subscript denotes "dashpot".
12. Each brass weight has a mass of 500 ± 10 g. (You may weigh the pieces if a more precise value is desired.) Calling the mass of the four weights combined m_w , use the following relationships to solve for the unloaded carriage mass m_{c2} , and spring constant k .^[29]

$$k/(m_w + m_{c2}) = (\omega_{nm21})^2 \quad (6.1-3)$$

$$k/m_{c2} = (\omega_{nm22})^2 \quad (6.1-4)$$

Find the damping coefficient c_{m2} by equating the first order terms in the equation form:

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = s^2 + (c/m)s + k/m \quad (6.1-5)$$

Repeat the above for the first mass carriage, spring and damping m_{c1} , c_{m1} and k respectively.

^[30] ^[31]

Calculate the damping coefficient of the dashpot, c_d .

13. Remove the carriage extension bracket and dashpot from the second mass carriage, replace the medium stiffness spring with a high stiffness spring (800 N/m nominally), and repeat Steps 5 and 6 to obtain the resulting natural frequency ω_{m23} . Repeat this frequency measurement

using the least stiff spring (nominally 200 N/m) to obtain ω_{m24} . Calling the value of stiffness obtained in Step 12 above $k_{med\ stiffness}$, calculate $k_{high\ stiffness}$ and $k_{low\ stiffness}$ from the frequency measurements of this step.

Now all dynamic parameters have been identified! Values for m_1 and m_2 for any configuration of masses may be found by adding the calculated mass contribution of the weights to that of the unloaded carriages^[32].

The following is necessary to establish the hardware gain for control modeling purposes.

Procedure:

13. Remove the spring connecting the first and second masses and secure four 500 g masses on the first mass carriage. (You should label this particular spring so that the identified parameter k_2 will be consistent when used in later experiments). Use the limit clamps to secure the second mass clear from the first. . Verify that the masses are secure and that the carriage slides freely. Hook up the drive power to the mechanism. Position the first mass approximately 3 cm to the left (negative x_1 position) of its center of travel.
14. In the Trajectory window deselect Unidirectional moves (i.e. enabling bi-directional inputs) select Step, Set-up. Choose *Open Loop Step*, and input *2.00 Volts, 75 ms, 2 reps*. Execute this open loop step via the Execute menu. (This move may trip a software speed limit or motion travel limit which disables the controller indicated by "Limit Exceeded" in the Controller Status box in the "desk top". Again, to reset, simply reselect Execute from the Execute menu.) Go to Set-up in the Plot Data menu and select Encoder #1 velocity for plotting.
15. Plot this data and observe four velocity profile segments with nominal shapes of: linear increase (constant acceleration), constant (zero acceleration^[33]), linear decrease (deceleration), and constant. Obtain the acceleration, \ddot{x}_{1e} , (counts/s²) by carefully measuring the velocity difference and dividing by the time difference (75 ms) through the positive-sloped linear segment.^{[34] [35]} Repeat this for the negative-sloped segment. Calculate the average magnitude of the positive and negative accelerations for use in obtaining k_{hw} below.
16. Save any files or plots of interest. Exit the executive program and power down the system.

Transfer Function Calculation

The so-called hardware gain, k_{hw} ,^[36] of the system is comprised of the product:

$$k_{hw} = k_c k_a k_t k_{mp} k_e k_{ep} k_s$$

(6.1-6)

where:

k_c , the DAC gain, = 10V / 32,768 DAC counts

k_a , the Servo Amp gain, = approx 2 (amp/V)

k_t , the Servo Motor Torque constant = approx 0.1 (N-m/amp)

k_{mp} , the Motor Pinion pitch radius inverse = 26.25 m⁻¹

k_e , the Encoder gain, = 16,000 pulses / 2 π radians

k_{ep} , the Encoder Pinion pitch radius inverse = 89 m⁻¹

k_s , the Controller Software gain, = 32 (controller counts / encoder or ref input counts)^[37]

In Step 15, we obtained the acceleration \ddot{x}_{1e} (counts/s²) of a known inertia, $m_I = m_w + m_{c1}$ with a known voltage applied at the DAC. This relates to the applied force during the acceleration according to:

$$\text{Applied Force} = m_I \ddot{x}_{1e} = m_I \ddot{x}_{1e} / (k_e k_{ep}) \quad (6.1-7)$$

7)

Thus we have a direct measurement of the five-term product $k_a k_t k_{mp} k_e k_{ep}$. i.e.:

$$2.00V k_a k_t k_{mp} = \text{Applied Force in Step 15} \quad (6.1-8)$$

Use (6.1-6 through 6.1-8) to solve for k_{hw} using the specified values for k_e and k_s .

For control purposes it is generally desirable to put the transfer function in denominator-monic form (leading term in D(s) has unity coefficient).

Questions / Exercises:

A. Report the measured properties and derived parameter values:

$m_w, m_{c1}, m_{c2}, m_{c3}, c_{m1}, c_{m2}, c_{m3}, c_d, k_{low\ stiffness}, k_{med\ stiffness}, k_{high\ stiffness}$ and k_{hw}

B. Construct a denominator-monic plant model suitable for control design for the case of four 500g weights on each of the first and second mass carriers with the 400 N/m (nominal) spring connecting the two and dashpot connected to the second mass – see Figure 6.1-1d. Use the above results and equations (5.1-4 through 5.1-6) to generate these transfer functions where the hardware gain multiplies the numerator for each transfer function. Construct the plant model for the same case except where there is no dashpot.

C. What are the units of k_{hw} ?

6.2 Rigid Body PD & PID Control

This experiment demonstrates some key concepts associated with proportional plus derivative (PD) control and subsequently the effects of adding integral action (PID). This control scheme, acting on plants modeled as rigid bodies finds broader application in industry than any other. It is employed in such diverse areas as machine tools, automobiles (cruise control), and spacecraft (attitude and gimbal control). The block diagram for forward path PID control of a rigid body is shown in Figure 6.2-1a where friction is neglected.^[38] Figure 6.2-1b shows the case where the derivative term is in the return path. Both implementations are found commonly in application and, as the student should verify, both have the identical characteristic roots. They therefore have identical stability properties and vary only in their response to dynamic inputs.

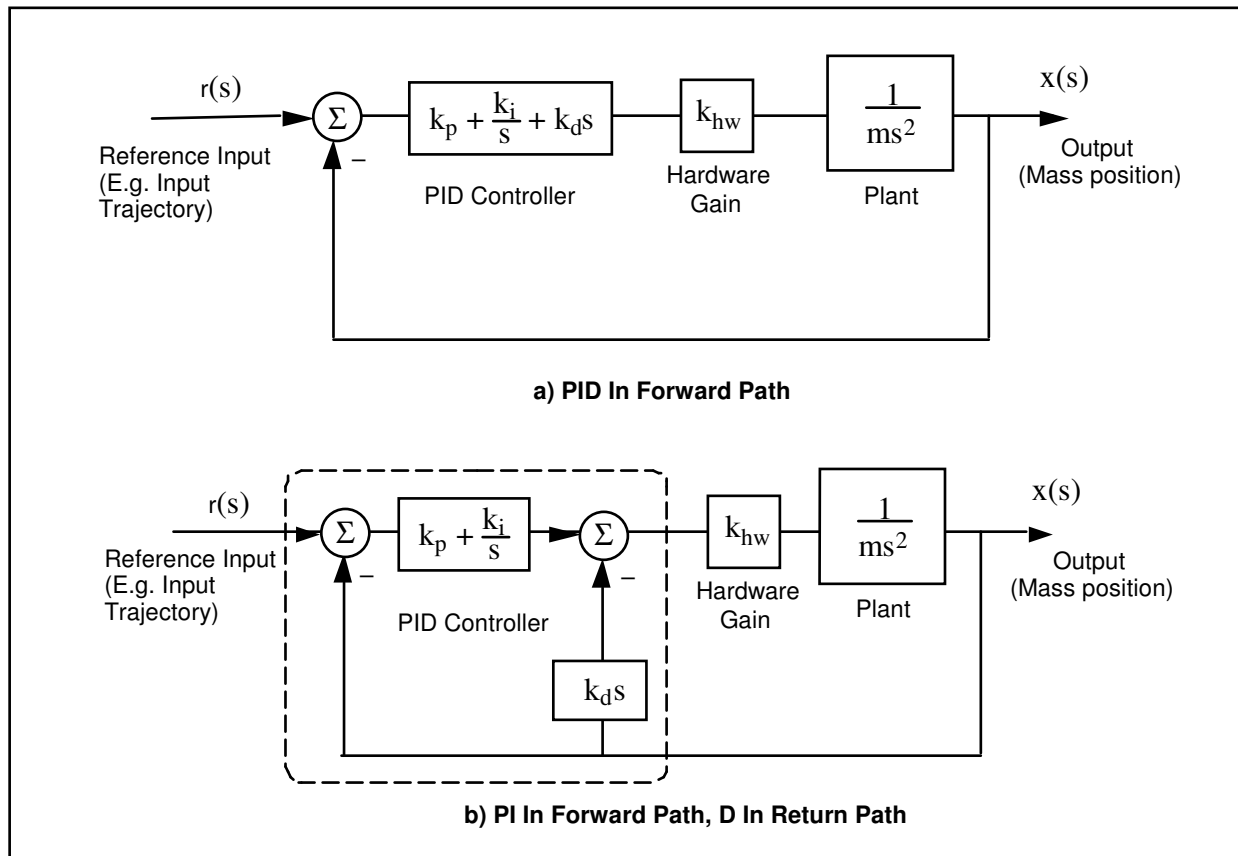


Figure 6.2-1. Rigid Body PID Control – Control Block Diagram

The closed loop transfer functions for the respective cases are:

$$c(s) = \frac{x(s)}{r(s)} = \frac{(k_{hw}/m)(k_d s^2 + k_p s + k_i)}{s^3 + (k_{hw}/m)(k_d s^2 + k_p s + k_i)} \quad (6.2-1a)$$

$$c(s) = \frac{x(s)}{r(s)} = \frac{(k_{hw}/m)(k_p s + k_i)}{s^3 + (k_{hw}/m)(k_d s^2 + k_p s + k_i)} \quad (6.2-1b)$$

For the first portion of this exercise we shall consider PD control only ($k_i=0$). For the case of k_d in the return path the transfer function reduces to:

$$c(s) = \frac{(k_{hw}/m)(k_d s + k_p)}{s^2 + (k_{hw}/m)(k_d s + k_p)} \quad (6.2-2)$$

By defining:

$$\omega_n \triangleq \sqrt{\frac{k_p k_{hw}}{m}} \quad (6.2-3)$$

$$\zeta \triangleq \frac{k_d k_{hw}}{2m \omega_n} = \frac{k_d k_{hw}}{2\sqrt{m k_p k_{hw}}} \quad (6.2-4)$$

we may express:

$$c(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (6.2-5)$$

The effect of k_p and k_d on the roots of the denominator (damped second order oscillator) of Eq (6.2-2) is studied in the work that follows.

Procedure :

Proportional & Derivative Control Actions

1. Using the results of Section 6.1 construct a model of the plant with four 500g mass pieces on the first mass carriage with no springs or damper attached. You may neglect friction.
2. Set-up the plant in the configuration described in Step 1. There should be no springs or damper connected to the first carriage and the other carriages should be secured away from the range of motion of the first carriage.
3. From Eq (6.2-3) determine the value of k_p ($k_d=0$) so that the system behaves like a $\sqrt{2}$ Hz spring-mass oscillator.
4. Set-up to collect Encoder #1 and Commanded Position information via the Set-up Data Acquisition box in the Data menu. Set up a closed-loop step of **0 (zero)** counts, dwell time = **3000 ms**, and **1 (one)** rep (via Trajectory in the Command menu).
5. Enter the Control Algorithm box under Set-up and set **$T_s=0.0042$ s** and select Continuous Time Control. Select PID and Set-up Algorithm. Enter the k_p value determined above for $\sqrt{2}$ Hz oscillation (k_d & $k_i = 0$, do not input values greater than $k_p = 0.08$ ^[39]) and select OK.

Place the first mass carriage at approximately the -0.5 cm (negative is toward the motor) mark.

In this and all future work, be sure to stay clear of the mechanism before doing the next step. Selecting Implement Algorithm immediately implements the specified controller; if there is an instability or large control signal^[40], the plant may react violently. If the system appears stable after implementing the controller, first displace it with a light, non sharp object (e.g. a plastic ruler) to verify stability prior to touching plant

Select Implement Algorithm, then OK.

6. Enter Execute under Command. Prepare to manually displace the mass carriage roughly 2 cm. Select Run, displace the mass approximately 3 cm and release it. Do not hold the mass position for longer than about 1 second as this may cause the motor drive thermal protection to open the control loop.
7. Plot encoder #1 output (see Step 6 Section 6.1). Determine the frequency of oscillation. What will happen when proportional gain, k_p , is doubled? Repeat Steps 5 & 6 and verify your prediction. (Again, for system stability, do not input values greater than $k_p = 0.06$).
8. Determine the value of the derivative gain, k_d , to achieve $k_d k_{hw} = 50$ N/(m/s).^[41] Repeat Step 5, except input the above value for k_d and set k_p & $k_i = 0$. (Do not input values greater than k_d

= 0.04).

9. After checking the system for stability by displacing it with a ruler, manually move the mass back and forth to feel the effect of viscous damping provided by k_d . Do not excessively coerce the mass as this will again cause the motor drive thermal protection to open the control loop.
10. Repeat Steps 8 & 9 for a value of k_d five times as large (again, $k_d \leq 0.04$). Can you feel the increased damping?

PD Control Design

11. From Eq's (6.2-3,-4) design controllers (i.e. find k_p & k_d) for a system natural frequencies $\omega_n = 4$ Hz, and three damping cases: 1) $\zeta = 0.2$ (under-damped), 2) $\zeta = 1.0$ (critically damped), 3) $\zeta = 2.0$ (over-damped).^[42]

Step Response

12. Implement the underdamped controller (via PI + Velocity Feedback) and set up a trajectory for a 2500 count closed-loop Step with 1000 ms dwell time and 1 rep.
13. Execute this trajectory and plot the commanded position and encoder position (Plot them both on the same vertical axis so that there is no graphical bias.)
14. Repeat Steps 12 & 13 for the critically damped and over-damped cases. Save your plots for later comparison.

Frequency Response

Review the discussion at the end of this section regarding the use of various sine sweep data scaling options prior to completing Step 16.

15. Implement the underdamped controller from Step 11. Set up a trajectory for a 600 count closed-loop Sine Sweep from 0.1 Hz to 20 Hz of 60 seconds duration with Logarithmic Sweep checked. (You may wish to specify Encoder #1 data only via Set-up Data Acquisition. This will reduce the acquired data size.)
16. Execute the trajectory and plot the Encoder 1 frequency response using Linear Time and Linear amplitude for the horizontal and vertical axes. The data will reflect the system motion seen as the sine sweep was performed. Now plot the same data using Logarithmic Frequency and Db amplitude. By considering the amplitude (the upper most portion of the data curve) you will see the data in the format commonly found in the literature for Bode magnitude plots. Can you easily identify the resonance frequency and the high frequency (>5 Hz) and low frequency (< 0.8 Hz) gain slopes? (i.e. in Db/decade).
17. Repeat Step 16 for the critically damped and overdamped cases.

Adding Integral Action

18. Now compute k_i such that $k_i k_{hw} = 7500$ N/(m-sec).^[43] Implement a controller with this value of k_i and the critically damped k_p & k_d parameters from Step 11. (Do not input $k_i > 3.0$).^[44] Be certain that the following error seen in the background window is within 20 counts prior to implementing.). Execute a 2500 count closed-loop step of 2000 ms duration (1 rep). Plot the encoder #1 response and commanded position.
19. Increase k_i by a factor of two, implement your controller (do not input $k_i > 3.0$) and plot its step response. Manually displace the mass by roughly 5 mm. Can you feel the integral action increasing the restoring control force with time? (Do not hold for more than about 2 seconds to avoid excessive force build-up and hence triggering the motor thermal protection.) What happens when you let go?

Questions:

- A. What is the effect of the system hardware gain, k_{hw} , the mass, m , and the control gains, k_p and k_d , on the natural frequency and damping ratio? Derive the transfer function for the mass/spring/damper system shown in Figure 6.2-2. How do the viscous damping constant, c , and the spring constant k correspond to the control gains k_d and k_p in the PD controlled rigid body of Figure 6.2-1b?
- B. Describe the effects of natural frequency and damping ratio on the characteristic roots of Eq's 6.2-1. Use an S-plane diagram in your answer to show the effect of changing ζ from 0 to ∞ for a given ω_n .
- C. Compare the step response and frequency response plots for the under-, critical, and overdamped cases ($k_i = 0$). Discuss how the resonance (if present), and *bandwidth* ^[45] seen in the frequency response data correlate with features of the step responses.
- D. What is the general shape of the frequency response amplitude (i.e. amplitude vs. time) of the three plots obtained in Step 16 (linear time / linear amplitude) at amplitudes well below and above ω ? What is the shape when viewing the same data plotted with $\log(\omega)$ / "Db scaling? Explain your answer in terms of the asymptotic properties of the closed loop transfer functions as ω tends to zero and infinity.
- E. Review the two step response plots obtained by adding integral action (Steps 18 & 19) with the previous critically damped and the critically damped plot ($k_i = 0$) of Step 14. What is the effect of the integral action on steady state error?

Static or Coulomb friction may be modeled as some disturbance force acting on the output as shown in Figure 6.2-3. Assume that this force to be step function ^[46], and use the final value theorem to explain the effect of such a step on the PD controlled system with and without the addition of integral action.

How does integral action effect overshoot (again, compare with the critically damped plot of Step 14). Why?

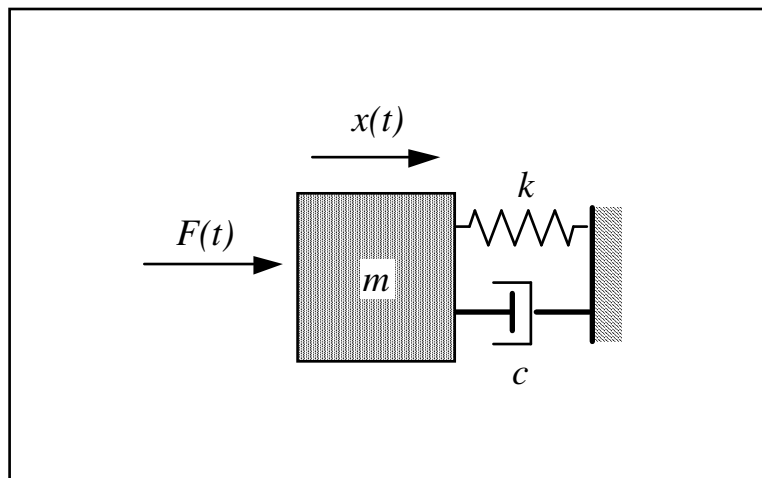


Figure 6.2-2. Mass/Spring/Damper

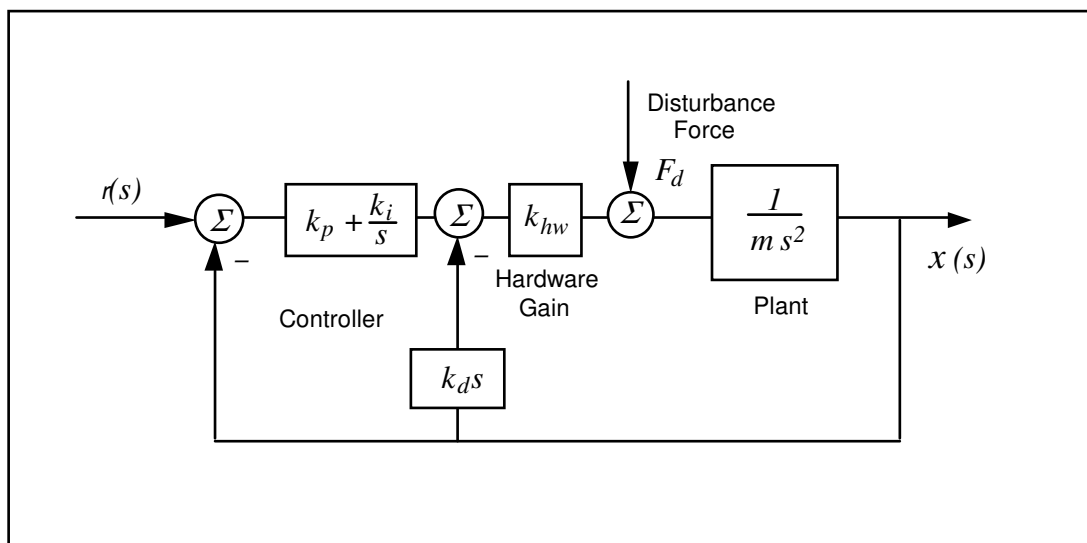


Figure 6.2-3. Friction Modeled as Disturbance Force at Output

Viewing Sine Sweep (Frequency Response) Plots

Much insight into frequency response behavior is afforded by viewing the above plots in the various data scaling functions available in the Setup Plot dialog box. By viewing the Linear Time - Linear Amplitude modes the data appears as the system motion was viewed during the sine sweep. Note the large range in amplitudes as frequency changes. In Linear Frequency - Linear Amplitude mode the amplitude is as before but the data is shown in with frequency of oscillation as the horizontal coordinate so that amplitude may be associated with a particular frequency.

In Logarithmic Frequency - Linear Amplitude mode the frequency is distributed logarithmically (the sine sweep was executed this way in our case). This is an appropriate frequency scaling function in many cases because magnitude changes of linear systems occur as powers (e.g. ω , ω^{-1} , ω^{-2} , etc.) of the excitation frequency and hence often occur over a large dynamic range of frequency. For example the change between 0.1 and 0.2 Hz may be greater than that between 9 and 10 Hz but would be difficult to ascertain in a linear frequency distribution between say 0.1 and 10 Hz because they would constitute only 1% of the plot length. Similarly, phase changes are symmetrically shaped in a logarithmic frequency distribution.

In Logarithmic Frequency - Db mode the frequency is distributed logarithmically and the magnitude is in Db. This method of data presentation, once the user is familiar with it, quickly affords much information about the system. The response magnitude asymptotically tends toward straight lines whose slope is associated with the salient system dynamics (i.e. the powers of ω mentioned above).

In ECP systems, the Db magnitude (see Eq. 6.2-6) of each data point is taken so that the upper bound of the trace represents the cycle-to-cycle maximum amplitude. In virtually all other plots that the engineer may encounter, peak Db magnitude is shown as a pure function of frequency - i.e. it is a single curve on the plot. Thus the mapping of actual test data into the frequency-Db format, as done here, affords physical insight into the meaning of these important analytical and design tools.

$$Db(x) = 20 \log(x) \quad (6.2-6)$$

The methods of Log ω - Db magnitude scaling, and Log ω - linear phase scaling are widely used in industrial and academic practice.

The Remove DC bias check box subtracts the average of the last 50 data points from all points on the curve to provide results that are centered about zero at high frequency. It generally provides better appearance to plots that have low amplitude in the high frequency section (e.g. all x_1 and x_2 sine sweeps in this manual) and is necessary in many cases to obtain useful Db data at high frequency. It may provide misleading results, however, if the bias of the original data or the amplitude in the final 50 points is large.

6.3 Disturbance Rejection of Various 1 DOF Plant Controllers

In this experiment, we consider the performance of three distinct controller designs in rejecting low and higher frequency disturbances. The plant should be setup as per the previous experiment (four 100 kg weights on the first carriage only, no springs or damper attached)

Procedure :

Setup Disturbance Apparatus

1. Connect the disturbance apparatus to the first mass carriage as per the instructions of Section 2.2
2. The three controllers are as follows:
 - a) PD control with $\omega_n \approx 2$ Hz and $\zeta \approx 1$, and feedback at Encoder #1. I.e. $k_p = 0.035$, $k_d = 0.0055$
 - b) Same as "a" plus added integral action: $k_i k_{hw} / m = 2500$.
 - c) Same as "a" plus a cascaded lead/lag filter with the following specifications: Zero at 0.2 Hz, Pole at 1.0 Hz, DC gain = 1.

Design the lead/lag filter for controller "c". (Hence find n_o , n_l , d_o , and d_l in $F(s) = (n_o + n_l s) / (d_o + d_l s)$)

3. Set up a time dependent sinusoidal disturbance (Sinusoidal (time)) of 1.0 volt, 0.1 Hz, 3 repetitions. (You should reduce the data acquisition sampling frequency to say once every 10-20 servo cycles to avoid an unnecessarily large data file). Input the k_p and k_d values for controller "a" under PID control using Encoder #1 for feedback with $T_s = 0.00442$ s. Perform a closed loop step of 0 (zero) count amplitude, 15000 ms duration, and 1 repetition. This sets the system up to acquire data during closed loop regulation for a total of 30 s. Execute this regulation for controllers "a" and "b" above making sure to check the box labeled "Include sinusoidal disturbance" when commanding the system. Plot and save the disturbance effort and encoder #1 data.
4. Input controller "c" via PID + Notch in the Setup Control Algorithm box. Execute the same regulation with disturbance as in Step 3, again plotting and saving your data.
5. Repeat Steps 3 and 4 with the disturbance frequency increased to 2.0 Hz with 30 repetitions. You may reduce the step duration to 10 s and increase the data acquisition frequency to every 5-10 servo cycles.

Questions:

- A. Consider the block diagram of Figure 6.3-1. Derive the open loop transfer function

$$\frac{N_{ol}(s)}{D_{ol}(s)} = K(s) k_{hw} P(s) \quad (6.3-1)$$

and closed loop transfer function $x_f(s) / F_d(s)$ for each controller.

- B. Plot the Bode response for the open and closed loop transfer functions in each of the three controller cases. Explain the disturbance attenuation characteristics of each controller in terms of their Bode response.

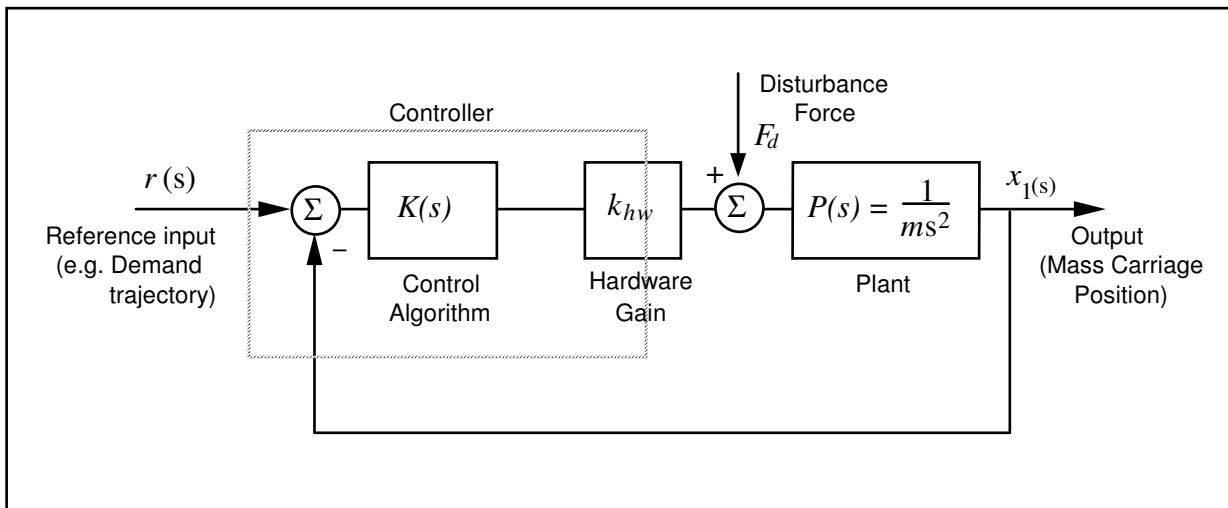


Figure 6.3-1 Disturbance Torque Input To Rigid Body Control Scheme

6.4 Collocated PD Control With 2 DOF Plant

In this experiment we consider PD control of a 2-mass system where the controlled output, x_1 , is of the first mass. Such a scheme is referred to as collocated since the sensor output is rigidly coupled to the actuator input.

The addition of the spring and second mass increases the plant order by two and adds an oscillatory mode to the plant dynamics. This may be thought of, in a sense, as a dynamic disturbance to rigid body plant studied in Section 6.2. The collocated control implemented here may be practically employed when there is flexibility between the actuator and some inertia, with the location of objective control being near the actuator. If the location of objective control is at the distant inertia, however, this method has its limitations.

The approach in this experiment will be to design the controller by interactively changing the PD gains and observing their effect on the physical system.

Procedure :

1. Set-up the system with four masses on the first and second carriage and with a medium stiffness spring (nominally 400 N/m) connecting the two as shown in Figure 6.1d but without the dashpot attached. I.e. in the same configuration (using the same spring) as was used in identifying the transfer function at the end of Experiment #1 without using the added dashpot damping.
2. Implement the critically damped controller from Step 11 of Section 6.2 being sure that encoder #1 is selected for control. Set-up data acquisition for encoders 1 & 2 and for commanded position and gather data every 5 servo cycles. Execute a 2000 count step response and plot the result for commanded position and encoder #1.
3. Now iteratively adjust the gains k_p & k_d and plot results to obtain an improved response. Make your gain adjustments gradually (not more than 50% at a time) and note the effects of increasing or reducing each of them. Do not input $k_p > 1.2$ or $0.01 < k_d < 0.05$. Attempt to achieve performance goals for the first mass of ≤ 200 ms rise time (0-90% amplitude) and $\leq 5\%$ overshoot without excessive oscillation. Save your best step response plot. Gently displace the first mass (manually) and note the relative stiffness of the servo system at the first mass.
4. For your last iteration in Step 3, plot the step response of the second mass. (There is no need to rerun the step, simply re-setup the plot for "encoder #2" & "commanded position" and plot data.) What is the predominant characteristic of the second mass motion? Can you give an explanation for the difference in the responses of the two masses in terms of their closed loop transfer functions?
5. Now using the existing values of k_p & k_d as starting points, iteratively reduce gains and plot x_2 results to provide a well-behaved step response with $\leq 10\%$ overshoot, without excessive oscillation, and as fast a rise time as possible. Save your final plot and record the corresponding gains. Manually displace the first and second masses and note their relative stiffness. Are they generally more or less stiff than for the controller of Step 3? How does the steady state error compare with the high gain controller from Step 3? Could this error be reduced by feedback about x_2 ?

Questions:

- A. Calculate the poles of the closed-loop transfer functions: $x_1(s)/r(s)$ and $x_2(s)/r(s)$ for your final controllers in Steps 3 & 5 respectively. How close

to the imaginary axis (& right half plane) are the most lightly damped poles in each case? How close are the complex poles of $x_1(s)/r(s)$ to its zeros in each case? Explain your answer in terms of the root loci for this system for gain ratios of $k_d/k_p = 0.05, 0.10, 0.17, \text{ and } 0.25$.

B. Calculate the closed loop transfer function in the form:

$$\frac{x_2(s)}{r(s)} = \frac{(N(s))_{\text{forward path}} / D_{ol}(s)}{1 + N_{ol}(s) / D_{ol}(s)} \quad (6.3-1)$$

Use N_{ol} and D_{ol} to obtain the open loop Nyquist or Bode responses resulting from your high and low gain controllers from Steps 3 & 5 respectively. What are the associated phase and gain stability margins? What are these margins for $x_1(s)/r(s)$? Explain.

C. Referring to Figure 6.4-1, the transfer function between the force disturbance and the first carriage displacement is:

$$\frac{x_1(s)}{F_d(s)} = \frac{N_1(s)}{D(s) + k_{hw}(k_p + k_d s) N_1(s)} \quad (6.4-1)$$

where $N_1(s)$ is the numerator in the right hand side of Eq(5.1-4).

Use the final value theorem to find this expression for a constant disturbance force. The inverse of this expression is called the *static servo stiffness*. What is the static servo stiffness of your final controllers from Steps 3 & 5 for the first mass carriage? What is the equivalent expression for the second carriage (i.e. defining static stiffness at the second carriage as the force required to displace the carriage one meter) and what is its value for the two controllers? Recalling Section 6.2, what is the static stiffness with integral action in the controller (i.e. $k_i \neq 0$)?

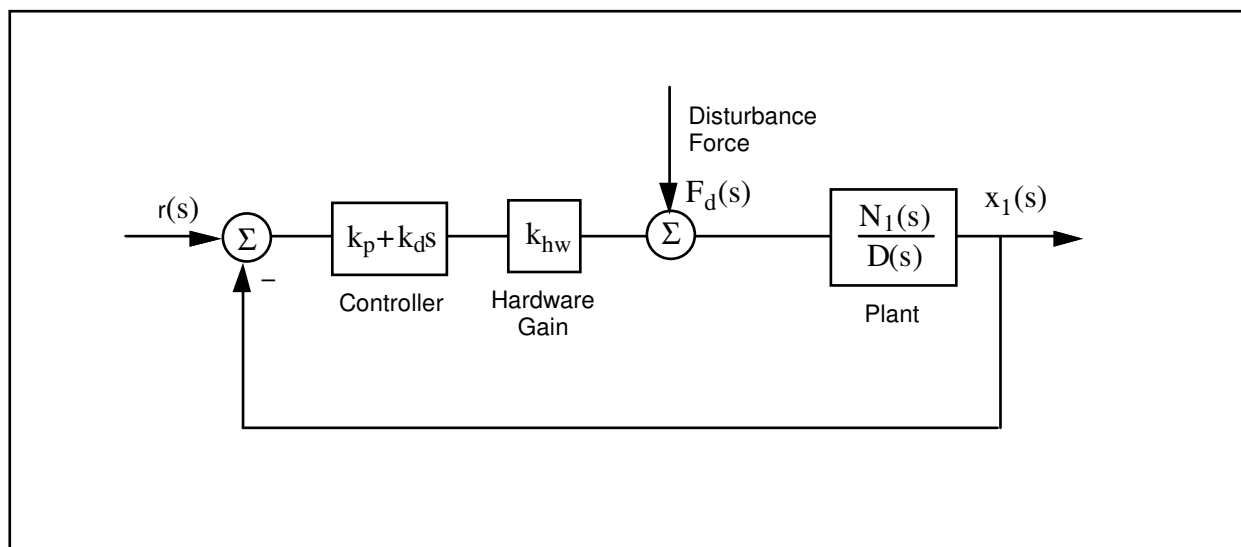


Figure 6.4-1. Disturbance Forces On PD Controlled 2-DOF Plant

6.5 PD Plus Notch Filter Control

In this experiment a control scheme is implemented which first closes the collocated loop with simple rate feedback to dampen the oscillatory mode. Then a notch filter is designed to further attenuate the transmission of signals at the damped mode frequency (i.e. nearly canceling the poles with a zeros). Finally, PD control is used to achieve certain performance goals..

Procedure :

1. The plant studied here is that of the previous exercise, i.e. the one shown in Figure 6.1-1d but without the dashpot. Consider the block diagram of Figure 6.5-1. Use root locus techniques to find the rate feedback gain, k_v , that provides the satisfactory damping of the complex roots of the inner loop $x_1(s)/r^*(s)$.
2. Implement this gain as the f_1 coefficient in the General Form controller specification box using $T_s = 2.652$ ms. Be sure that you select Encoder #1 for Feedback Loop #3 before exiting the box and implementing. Attempt to manually excite the oscillatory mode via the second mass and notice the damping effect of velocity feedback.
3. We may now design for the new "plant" $G^*(s)$ where $N_2(s)$ is as before, and:

$$D^*(s) = D(s) + k_v s N_1(s) \quad (6.5-1)$$

Design a notch filter, $N_n(s)/D_n(s)$ with two poles at $5.0 \pm 5.0i$ Hz and two additional higher frequency poles at $8.0 \pm 8.0i$ Hz^[47] and with two zeros at the poles of $D^*(s)$. Normalize the notch filter transfer function to have unity DC gain.^[48]

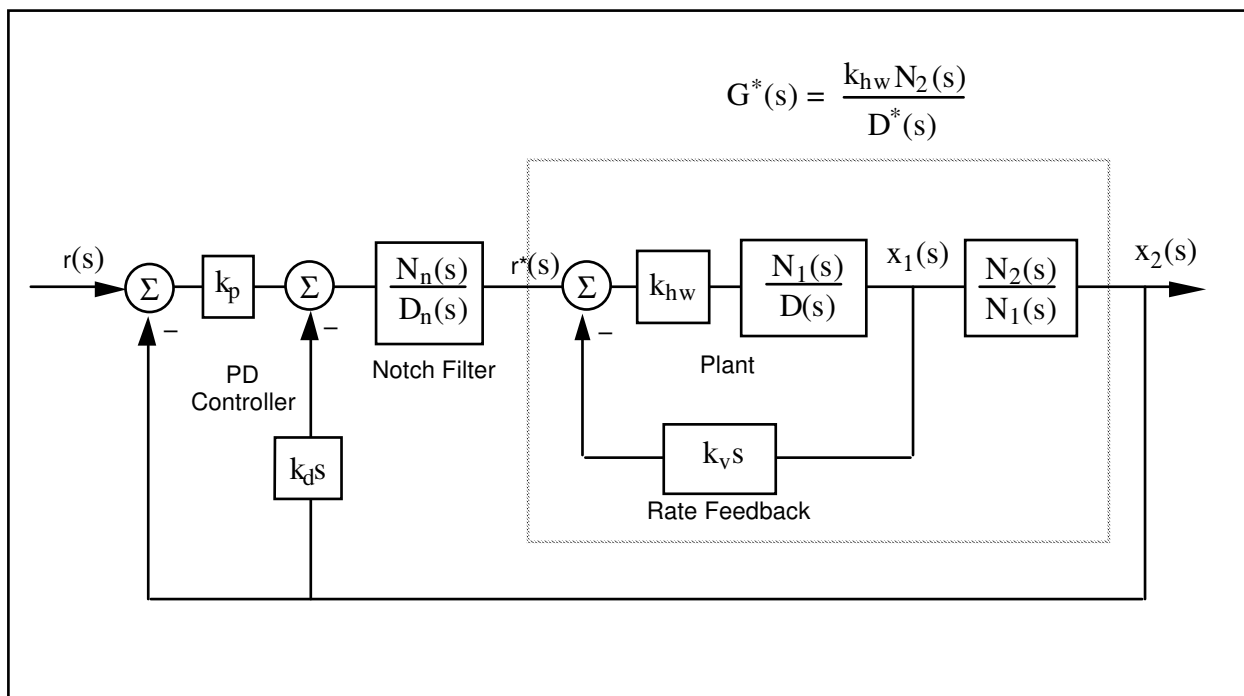


Figure 6.5-1.

Control Structure For Section 6.5 Experiment

4. Select some initial PD gains for control of x_2 . Use values of $k_p = 0.01$, $k_d = 0.001$ initially. Do not exceed $k_p = 0.05$ and $k_d = 0.05$ in any subsequent run. Since this control scheme uses multiple loop feedback, the Executive program requires that it be implemented via the general control algorithm form. A short-cut to controller coefficient entry is as follows: Input your notch filter coefficients and proportional controller gain, k_p , via the dialog box PID + Notch. Be certain to enter these coefficients in the proper order and to high numerical precision (e.g. 8 decimal places). Exit and select Preview In General Form. You will see the P + notch controller in the form that it is implemented in the generalized controller form.^[49]

In the General Form window, enter the f1 coefficient calculated in Step 1 (i.e. k_v), and enter as i1 the desired derivative gain, k_d . Make sure that Encoder #1 is selected for Loop #3 and Encoder #2 is selected for Loops #1&2. You should verify that these entries appropriately represent the control structure of the controller in Figure 6.5-1. Exit the box and make sure that General Form is selected; then Implement. Now check performance of the control using step trajectories of 500 - 1000 counts.^[50]

5. Iteratively change PD gains by repeating Step 4 to obtain performance goals of 0.4 sec rise time (0-90% of final amplitude) and overshoot less than 15%. Do not exceed $k_p = 0.10$ or $k_d = 0.01$. You may notice a trend here that for relatively high values of k_d the system transmits excessive noise and may appear "twitchy". As k_p becomes large, the system becomes oscillatory and further increases lead to instability. Does increase in noncollocated derivative action necessarily reduce the step response oscillations?
6. Record your best performance step response. Manually displace the masses and note the relative static stiffness of the noncollocated (second) carriage under this control.

Questions:

- A. Report your calculated values for k_p , $N_n(s)$, $D_v(s)$, and your selected values of k_p , k_d . Submit your step response plot. Does it meet the performance goals of Step 5?
- B. Calculate the closed-loop transfer function $x_2(s)/r(s)$ including all elements in the block diagram of Figure 6.6-1. You may express this in terms of the polynomials $D(s)$, $N_1(s)$, $N_n(s)$, etc. rather than expanding each term fully. Use the equation

$$\frac{x_2(s)}{r(s)} = \frac{(N(s))_{\text{forward path}} / D_{ol}(s)}{1 + N_{ol}(s) / D_{ol}(s)} \quad (6.5-2)$$

and determine the phase and gain margins of your system design via Bode or Nyquist plots. What would happen if you doubled your gains k_p and k_d ? What if you quadrupled them?

- C. For this design, determine the static stiffness at x_1 and x_2 for F_{c1} and F_{c2} applied at m_1 and m_2 respectively (i.e. F_{c1}/x_1 and F_{c2}/x_2). What are the "cross" static stiffnesses (i.e. F_{c2}/x_1 and F_{c1}/x_2)? How do the relative stiffness at the first and second masses compare in each case? How does the static stiffness of this design compare with those of the last section?

Based on these results would you expect relatively high or low steady state errors for this design.

6.6 Successive Loop Closure / Pole Placement Design For 2 DOF Plant

In this experiment we first close a position loop about the collocated (x_1) position with a relatively high bandwidth (close tracking) PD control. We then make the assumption that the first carriage closely follows its internal demand $r^*(s)$ so that for designing a controller for x_2 , the "plant" is approximated by the transfer function $x_2(s)/x_1(s)$ (i.e. $N_2/N_1(s)$). The block diagram for this approach is given in Figure 6.6-1.

The design and control implementation in this section proceeds as follows

1. High bandwidth PD control of x_1
2. Low pass filter augmentation to attenuate signal noise due to high PD gains
3. Outer loop control of x_2 via pole placement methodology

6.6.1 PD Control Of The First Mass Carriage

Set-up the system with four masses on the first and second carriage and with a medium stiffness spring (nominally 400 N/m) connecting the two. I.e. in the same configuration (using the same spring) as the previous experiment.

PD Control Design & Implementation

1. Using Eq's (6.2-3, -4) design the PD control gains such that $\omega_n = 10$ Hz, and $\zeta = 0.707$ when considering only m_1 acting as a rigid body.
2. Set the sampling rate to $T_s = 0.002652$ seconds and implement your gains via the PID control algorithm box. You may notice some system noise associated with the high derivative gain term. After safety checking the controller (See Section 2.3) displace the first and second mass carriages and notice their relative stiffness. Discontinue the control via Abort Control.

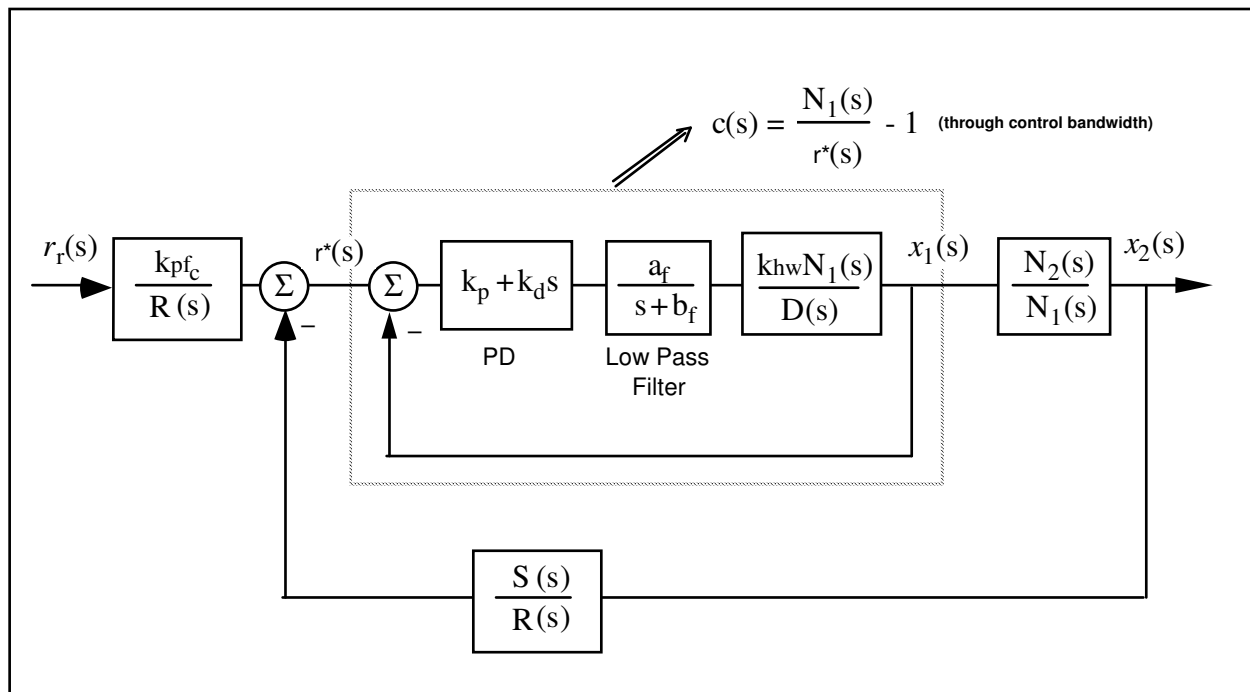


Figure 6.6-1. Control Structure For Successive Loop Closure with High Gain Inner LoopLow Pass Filter Design & Implementation

3. Solve for the constants a_f and b_f such the filter has a pole at $s = -240$ (approximately 40 Hz) and has unity DC gain.
4. Calculate the numerator and denominator of the controller associated with the cascade of the PD and low pass filter blocks. These will have the form

$$\text{PD*Filter} = \frac{e_0 + e_0s}{g_0 + g_1s} \quad (6.6-1)$$

5. In the Generalized Form box, enter the following.
 - a. Enter your calculated e_i 's and g_i 's
 - b. Select Encoder #1 for Loop #2. (the other loops may have any encoder selected at this point)
 - c. Set the following equal to 1: t_o, h_o, i_o, j_o
 - d. Verify that $r_1 = 0.0000002$ ^[51]
 - e. Verify that all other coefficients = 0

It is important to take care and assure that all parameters are properly set before leaving the Generalized Form dialog box and implementing control.

6. Exit the dialog box, verify that the Generalized Form button is selected, and Implement control. You should notice a reduction in any audible signal noise that may have been present without the low pass filter. Safety check the controller and manually displace the carriages. They should behave as before in Step 2.

6.6.2 Pole Placement Control of $x_2(s)/x_1(s)$

Having closed a relatively high bandwidth (≈ 10 Hz) loop about the first mass, we utilize the fact that the transfer function of Eq. 6.2-5 has near unity input/output gain (and near zero phase) through the bandwidth (≈ 2.5 Hz) that we will attempt to attain in the overall control of x_2 . Thus for the control of x_2 we consider the outer loop in the block diagram of Figure 6.6-1.

Now the plant to be controlled is:

$$\frac{x_2(s)}{x_1(s)} = \frac{N_2}{N_1} \triangleq \frac{N^*(s)}{D^*(s)} \quad (6.6-2)$$

The numerical values of the parameters in this expression were determined in Experiment #1.

We now seek to find a controller $S(s)/R(s)$ which will result in a prescribed set of closed loop poles. The closed loop denominator will have the form:

$$D_{cl}(s) = D^*(s)R(s) + N^*(s)S(s) \quad (6.6-3)$$

which may be expressed as^[52]

$$D_{cl}(s) = (d_2s^2 + d_1s + d_0)(r_1s + r_0) + (n_0)(s_1s + s_0) \quad (6.6-4)$$

where the d_i 's and n_i 's are the respective coefficients of the denominator and numerator of the right hand side of Eq. (6.6-2). Their values are known from the plant model Eq's (5.1-4,-5) (or Eq's (5.1-7,-8)).

By linear system theory, for coprime $N^*(s)$, $D^*(s)$ with $N^*(s)/D^*(s)$ proper, there exists an $(n-1)^{\text{th}}$ order $S(s)$, $R(s)$ which when convolved as per Eq. (6.6-3) form an arbitrary $(2n-1)^{\text{th}}$ order $D_{cl}(s)$ where n is the order of $D^*(s)$.

Here we shall solve for the desired denominator:

$$D_{cl}(s) = \left(s + 5\pi \frac{(1+j)}{\sqrt{2}}\right) \left(s + 5\pi \frac{(1-j)}{\sqrt{2}}\right) (s + 5\pi) \quad (6.6-5)$$

I.e. closed loop poles at -2.5, and $-1.77 \pm j1.77$ Hz.^[53]

-

Pole Placement Design

7. Determine the coefficients of the controller polynomials $S(s)$ and $R(s)$ by equating coefficients in the expanded forms of Eq's 6.6-4 and 6.6-5.
8. Calculate the scalar prefilter gain k_{pf} by referring to Figure 6.6-1. The goal is to have the output $x_2(s)$ scaled equal to the input $r_r(s)$. Hint: Consider the system in static equilibrium. Set $x_2 = 1$ and $r_r = 1$ and solve for k_{pf} using only the constant terms in all control blocks.

Control Implementation

9. In the Executive program, set-up to collect Encoder #1', Encoder #2 and Commanded Position information^[54] via Set-up Data Acquisition with data sampling every two sample periods. Setup a closed loop step trajectory of 2500 counts, 2000 ms duration and 1 repetition.
10. Return to the General Form Algorithm box and verify that the PD controller, low pass filter and all other coefficients are entered as given in Step 5 above. Enter the coefficients for $S(s)$ and $R(s)$ determined in Step 7. Enter the value k_{pf} calculated in Step 8 as to. Make sure that Encoder #2 is selected for Loop #1 and Encoder #1 for Loop #2. Select OK to exit to the controller selection dialog box.
11. While staying clear of the mechanism select Implement Algorithm. If the mechanism reacts violently you have implemented an unstable controller or otherwise improperly entered the control coefficients and you will need to repeat the above steps as appropriate. You should first Reset Controller (Utilities menu) before attempting to re-implement control. If the system is well behaved, and after safety checking the controller, you may disturb the mass carriages lightly. Notice the relative stiffness of the two carriages and how the first carriage moves in opposition to (i.e. regulation of) disturbances of the second carriage.

You may notice some "twitching" or buzzing due to noise in the system. This often occurs in such high gain systems, but has been mitigated via the low pass filter. If the noise is excessive or there is any possibility that the equipment is at risk discontinue control immediately.

12. Execute the Step input previously programmed, and plot the Encoder 1, Encoder #2, and Commanded Position data. Save your plot. How does the response at x_2 compare with designs previously tested? Describe the motion of x_1 and how it shapes the response at x_2 .

Questions:

- A. Report your calculated values for $k_p, k_d, a_p, b_p, e_p, e_r, g_p, g_r, s_p, s_r, r_p, r_r,$ & k_{pf}
- B. Determine an expression for the closed-loop transfer function $x_2(s)/r(s)$ including all elements in the block diagram of Figure 6.6-1. You may express this in terms of the polynomials $D(s), N_1(s), R(s)$, etc. rather than expanding each term fully. Determine $x_2(s)/r(s)$ using the assumption $c(s)=1$. Compare the simulated frequency response of the full and reduced order transfer functions. In which regions are the two similar in magnitude and phase and in which are they different? Are they similar throughout the final system closed loop bandwidth? Is the assumption of unity gain in $c(s)$ valid for the purposes here?
- C. Determine the phase and gain margins for the outer loop considering the full expression for the closed inner loop $c(s)$ (i.e. consider $c(s)$ to be a single block in the outer loop block diagram). Determine the phase and gain margins of the outer loop using the $c(s) = 1$ assumption. How do the two compare?
- D. For this design, determine the static stiffness at x_1 and x_2 for F_{c1} and F_{c2} applied at m_1 and m_2 respectively (i.e. F_{c1}/x_1 and F_{c2}/x_2). What are the “cross” static stiffnesses (i.e. F_{c2}/x_1 and F_{c1}/x_2)? How do the relative stiffnesses of the first and second carriage compare in each case?

6.7 LQR Control

In this experiment a linear quadratic regulator is implemented using full state feedback. The plant used is again the same 2 DOF configuration as used in the previous three experiments. In this experiment a linear quadratic regulator is implemented using full state feedback. The states chosen are the mass positions and rates according to the model of Eq (5.1-3) with the output taken primarily as x_2 , with a 2% weighting in the collocated rate i.e.

[55]

$$C = [0 \quad 0.02 \quad 1 \quad 0] \quad (6.7-1)$$

LQR Design:

1. Construct a state space model of the plant using the realization of Eq (5.1-3) and measured parameter values from Section 6.1.
2. The following notation shall be used for LQ optimization:

Feedback law:

$$u = -Kx \quad (6.7-2)$$

where

$$K = [K_1 \quad K_2 \quad K_3 \quad K_4] \quad (6.7-3)$$

Perform LQR synthesis via the Riccati equation solution^[56] or numerical synthesis algorithms

to find the controller K which minimizes the cost function (scalar control effort):

$$J = \int (x' Q x + u^2 r) dt \quad (6.7-4)$$

In this synthesis choose $Q=C'C$ so that the error at the intended output, x_2 , is minimized subject to the control effort cost. Perform synthesis for control effort weight values: $r = 1000, 100, 10, 1, \text{ and } 0.1$. Calculate the closed loop poles for each case as the eigenvalues of $[A-BK]$

- 3) From this data, select a control effort weight to put the lowest pole frequency between 3 and 4 Hz. Use one of the above obtained K values if it meets this criteria, or interpolate between the appropriate r values and perform one last synthesis iteration. Do not use K_1 or K_3 values greater than 0.75, or K_2 or K_4 values greater than 0.1. ^[57]

Control Implementation

- 4) Implement your controller via the State Feedback box under Control Algorithm. ^[58] For tracking, the prefilter gain K_{pf} must be set equal to $K_1 + K_3$. You may wish to select Preview In General Form to see how these parameters are mapped into the generalized algorithm.
- 5) Execute a 2500 count step and plot the result. How do the rise time, overshoot, servo stiffness, and steady state errors compare with previous controllers?

Questions / Exercises:

- A. Report your calculated values of the closed loop poles for the various values of r in Step 2 and for your final design. Report the values of K , for your final design.
- B. Calculate the closed-loop transfer function $x_2(s)/r_f(s)$ including all elements in the block diagram of Figure 6.7-1b (lower subfigure). You may express this in terms of the polynomials $D(s)$, $N_1(s)$, N_2 , and K_i ($i = 1,2,3,4$). Determine the phase and gain margins for the system by considering the open loop “numerator” to be all terms in the closed loop denominator except $D(s)$. ^[59]
- C. For this design, determine the static stiffness at x_1 and x_2 for F_{c1} and F_{c2} applied at m_1 and m_2 respectively (i.e. F_{c1}/x_1 and F_{c2}/x_2). What are the “cross” static stiffnesses (i.e. F_{c2}/x_1 and F_{c1}/x_2)? How do the relative stiffness of the first and second carriage compare in each case?

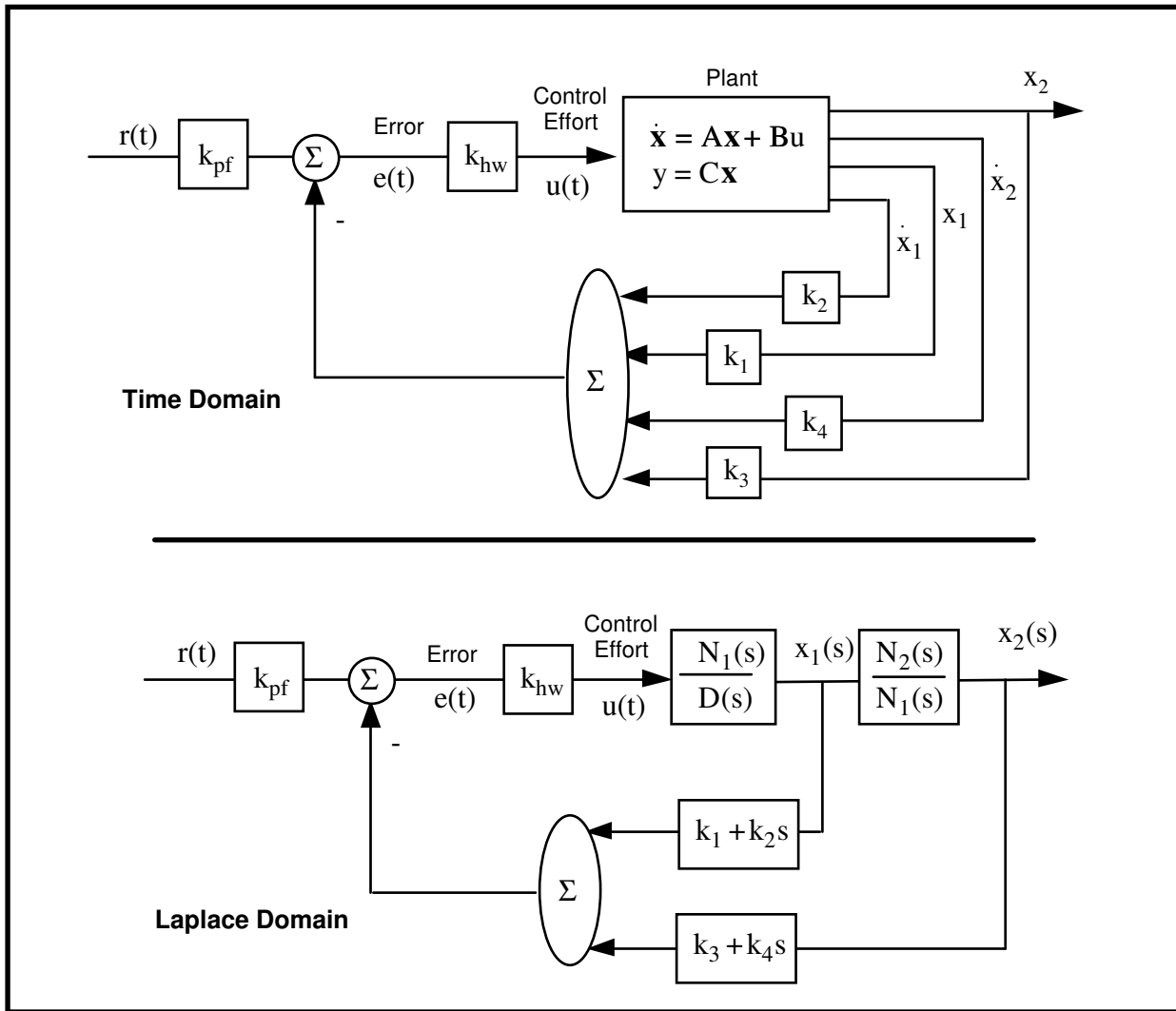


Figure 6.7-1. Representations of the Full State Feedback System

6.8 Practical Control Implementation

This section demonstrates non-ideal behaviors that affect many controls applications, and gives guidelines for mitigating their effects. These are: drive saturation, sensor quantization, and discrete time sampling. Other effects include drive flexibility which is addressed in detail in the above experiments, and various forms of disturbances which may be studied using the optional disturbance drive.

6.8.1 Effect of Drive Saturation

Saturation is when the output of a device is held at some positive (negative) value for all input values above (below) some limit. This is illustrated in Figure 6.8-1 which depicts the input/output relationship for an otherwise linear device. Saturation may occur due to the control effort signal exceeding the range or capability of the DAC, servo amplifier, or even the sustainable magnetic field of the actuator. In practice however the amplifier and actuator are usually selected to accept full scale DAC voltage or the control electronics are programmed such that the DAC output saturates below control magnitudes that would saturate or damage other drive components.^[60]

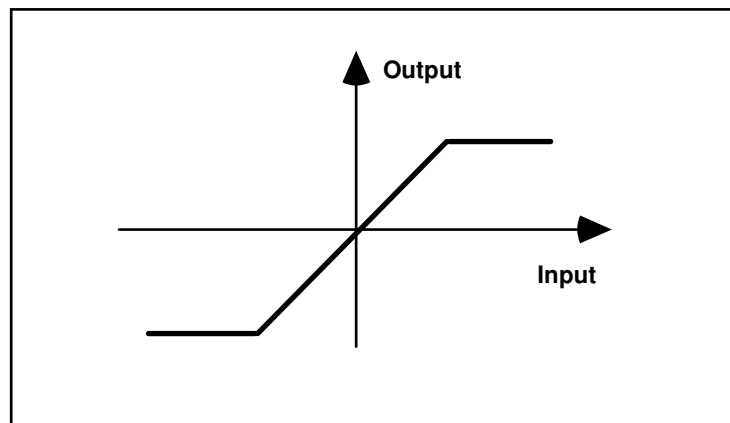


Figure 6.8-1. Linear Device Saturation

Qualitative Changes in Step Response For High Bandwidth Controller /

- 1 Secure four 500 g. brass weights to the first carriage and disconnect it from the second (i.e. remove the spring if it is present). Implement the following parameters under PID control (not PI with Velocity Feedback): $T_s = 0.00442$ s., $k_p = 0.04$, $k_d = 0.0060$, $k_i = 0$. The student should verify that these gains correspond to roughly $\omega_n = 2$ Hz and $\zeta = 1.0$ ^[61]. Setup to acquire Encoder 1, Commanded Position, and Control Effort every 1 servo cycle. Perform a 250 count step maneuver of 1, then a 10,000 count step. Plot and save your data.
- 2 Change the gains to $k_p = 0.04$, $k_d = 0.0060$, $k_i = 0$ and implement. These correspond to roughly $\omega_n = 10$ Hz and $\zeta = 1.0$, and finally a 16,000 count step. Again perform a 250 count step maneuver, then a 10,000 count step. Plot and save your data.

Questions / Exercises:

- A. Compare the 250 count and 10,000 count step responses and associated control effort for the forward path differentiator DAC saturation tests of Step 1. Plot the simulated ideal step response for a forward and return path differentiated PD controlled system (e.g. for the return path case plot the response of the system of Eq. 6.2-5 using $\omega_n = 2$ Hz, $\zeta = 1$). Which ideal response does the experimental data follow more closely for the

- 250 and 10,000 count amplitudes? (you may wish to view the data and simulated plots over rather short duration of say 0.5 or 1.0 sec. to more clearly view the initial control effort). Why?
- B. Compare the speed and shape of the 250 count steps for the controllers in Steps 1 and 2. Why are they different? Compare the shape of the 250 and 10,000 count maneuvers of Step 2 (here you may want to reduce the plot time to the first few hundred milliseconds). What is the shape of the motion in the first 80 ms (approximately)? Why?
- C. Describe in what way the following factors affect whether and to what extent a given set of drive hardware will reach its force limit or saturate: controller bandwidth, commanded acceleration, commanded amplitude.

6.8.2 Effect of Discrete-time Sampling

With modern high speed data processing capability, sampling rates are often high enough that control may be analyzed and implemented using continuous time analytical methods. In cases where either the objective control bandwidth is high, the control processor is otherwise heavily burdened, or economic considerations require lower speed processing capability, the sample rate may be sufficiently low that its effect is significant.

Of particular interest is the phase lag due to the time delay associated with sampled data systems. Such time delays propagate from signal sampling and holding in ADC conversion of analog feedback sensors (not the case for the Model 210 system) and in the control algorithm itself. Further time delay and phase lag results from DAC conversion of control effort and in the analog servo amplifier. While a detailed analysis of the associated time delay is beyond the scope of this exercise, the phase lag associated with discrete time sampling is on the order of that of a pure time delay of length equal to the sample period, T_s . Such a delay, denoted here as $\phi(s)$, is expressed in the Laplace domain as:

$$\phi(s) = e^{-sT_s} \quad (6.8-1)$$

In the frequency domain, its phase angle is:

$$\arg(\phi(j\omega)) = -T_s \omega \quad (6.8-2)$$

where T_s is in seconds, and $\arg(\phi(j\omega))$ and ω are in consistent angular units.

Procedure

- Set up the mechanism with the first carriage only attached and with no weights attached. Verify that the Data Acquisition function is set to sample data every $1 T_s$. Implement the following parameters under PI with Velocity Feedback: $k_p = 0.01$, $k_d = 0.0016$. Use Encoder #1 for feedback and set $T_s = 0.00442$ s. These gains correspond to approximately $\omega_n = 2$ Hz and $\zeta = 1.0$ for this reduced inertia. Perform a 5000 count step. Gradually increase the sample period and repeat the procedure until significant oscillations occur and the system exhibits near instability. Do not implement sampling periods greater than 75 ms. Do not increase sample period after once oscillation amplitudes of 2000 counts peak-to-peak have been observed. Record the highest value of T_s prior to instability. Plot the step response and note the oscillation frequency.
- Repeat Step 3 for $\omega_n \approx 10$ Hz and $\zeta \approx 1.0$ ($k_p = 0.25$, $k_d = 0.008$). Increase sample rates by one increment (0.000884 s.) at a time only. Do not implement sampling periods greater than 15 ms in this case. Do not implement again once a limit cycle oscillation has been identified.

Exercise:

- D. Compute the continuous-time Bode phase margin of the two (2 & 10 Hz) systems and note the associated critical frequency (the frequency where the open loop gain curve crosses 0 Db). Using this information and the approximation for sampling related phase lag from Eq. (6.8-2) to explain the instability. Explain the difference in instability inducing sample rate for the two systems. How does the oscillation frequency compare with the critical frequency? Explain.

6.8.3 Effect of Finite Wordlength & Sensor Quantization

Numerical quantization occurs by necessity in all control systems that contain digital control elements. Such effects (also known as finite word-length effects) include coefficient and data quantization, DAC and ADC resolution, signal dynamic range, and round-off noise. These can have profound consequences with respect to system stability and performance. High dynamic order controllers require particular care in providing robust algorithmic treatment and high data fidelity.^[62]

Consider the effect of the position measurement quantization of the feedback encoders. The discrete encoder position measurement may be operated on by high bandwidth control law terms such as the backwards differentiator in discrete-time PD control which can result in quantization noise. The backwards differentiator approximates angular rate as

$$\omega_n \cong \frac{x_n - x_{n-1}}{T_s} \quad (6.8-3)$$

where x_n is the current encoder measurement and x_{n-1} is the one immediately prior. Calling the encoder measurement resolution Δ_x ,^[63] the backwards difference rate quantum, Δ_v , becomes

$$\Delta_v = \frac{\Delta_x}{T_s} \quad (6.8-4)$$

The associated incremental control torque T_{Δ_v} (which is transformed to a force via the rack and pinion drive) is

$$T_{\Delta_v} = k_d \Delta_v k_{hw} = \frac{k_d \Delta_x k_{hw}}{T_s} \quad (6.8-5)$$

As T_{Δ_v} becomes large, a single encoder pulse (which can occur at infinitesimal rates) can cause relatively high instantaneous control effort. This propagates through the system causing high frequency motion in the sensed position further causing high instantaneous motor torque – i.e. noise is propagated.

From Eq.(6.8-5) torque (or control effort) quantization increases with reduced encoder resolution.^[64] For a given encoder, the torque quantization and hence torque noise increases with increased derivative^[65] gain and with reduced sampling period.

Procedure

- Secure four 500 g. brass weights to the first carriage and disconnect it from the second (i.e. remove the spring if it is present). Input the following parameters under PI with Velocity Feedback: $k_d = 0.01$, $k_p = 0$, $k_i = 0$ (i.e. the controller will implement a pure backwards

differentiator) Set $T_s = 0.00442$ s. Before proceeding, note the following:

If at any time in this experiment excessive noise occurs, shut down open the control loop immediately via Abort Control. Any sustained operation with high frequency noise (and/or high motor torque) can cause loosening or other damage of drive components.

6. Implement the control and displace the first carriage with a ruler and note if any noise occurs. Reduce T_s by one increment (0.000884 s) and re-implement, displace the carriage and check for noise. Repeat this procedure (by one increment at a time) until quantization noise is clearly detected. Abort control immediately upon hearing noise and do not repeat at a lower sample period. Note the last sample period.
7. Repeat Steps 1 and 2 for $k_d = 0.04$ (beginning at $T_s = 0.00442$ s. and reducing it by one increment at a time). Do not reduce T_s below 1.768 ms unless

Question:

- E. Is the relationship between derivative gain, sampling period, and torque noise implied by Eq.(6.8-5) generally consistent with the results of Steps 6 and 7?

Appendix A. Dynamic Modeling Details

A.1 Dynamics of Ideal Plant

A.1.1 Equations Of Motion

An idealized approximation of the rectilinear spring/mass plant is shown in Figure A-1a where friction is neglected. Using the free body diagram of Figure A-1b and summing forces acting on m_1 we have via Newton's second law

$$F(t) - x_1(k_1+k_2) + x_2k_2 = m_1\ddot{x}_1$$

or:

$$m_1\ddot{x}_1 + (k_1+k_2)x_1 - k_2x_2 = F(t) \quad (\text{A.1-1})$$

Similarly from Figure A-1c for m_2 :

$$m_2\ddot{x}_2 + (k_2+k_3)x_2 - k_2x_1 = 0 \quad (\text{A.1-2})$$

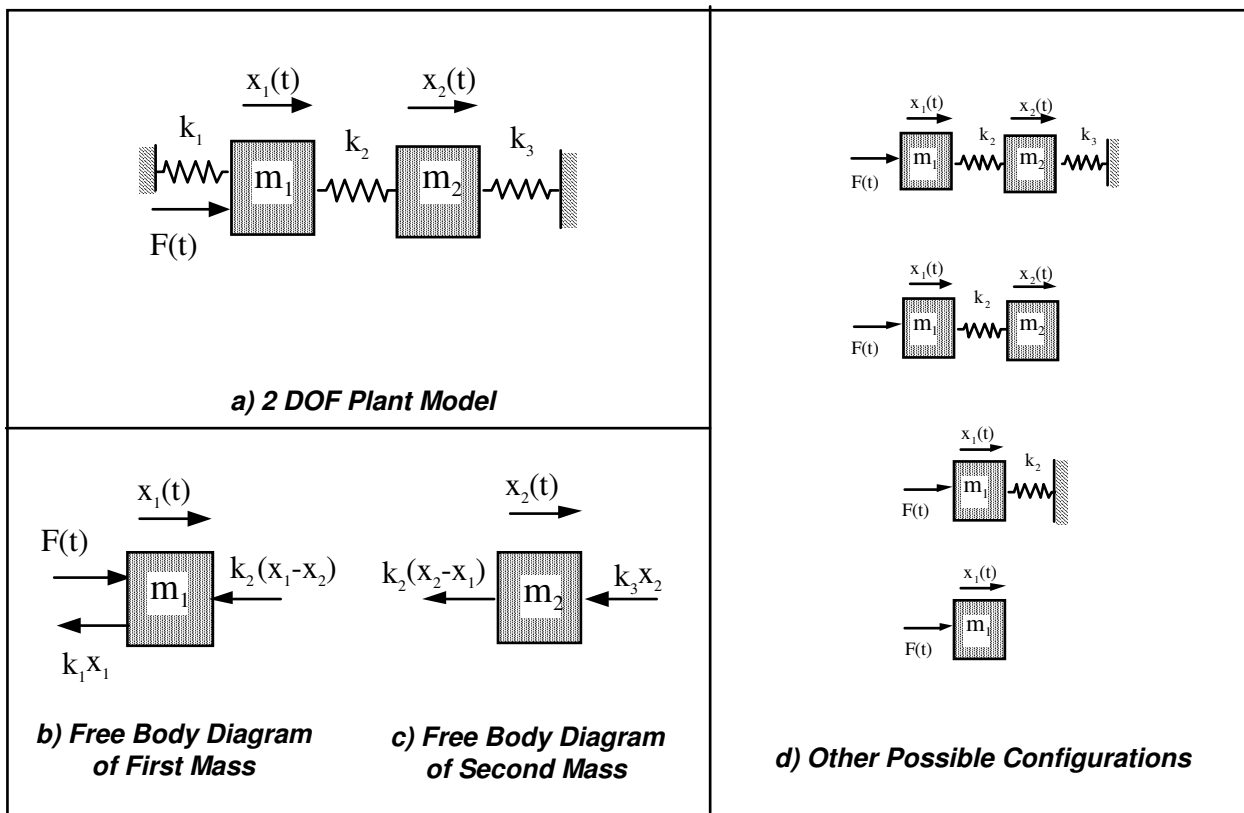


Figure A-1 Idealized Plant Models

or in matrix notation:

$$m\ddot{x} + kx = F \quad (\text{A.1-3})$$

where

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad m = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix}, \quad k = \begin{bmatrix} k_1+k_2 & -k_2 \\ -k_2 & k_2+k_3 \end{bmatrix}, \quad F = \begin{bmatrix} F \\ 0 \end{bmatrix} \quad (\text{A.1-4})$$

A.1.2 State Space Realization

A state space realization which is useful in several control approaches to this system follows directly from Eq A.1-4. i.e.:

$$\begin{aligned} \dot{x} &= Ax + BF(t) \\ Y &= Cx \end{aligned} \quad (\text{A.1-20})$$

where:

$$X = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(k_1+k_2)/m_1 & 0 & k_2/m_1 & 0 \\ 0 & 0 & 0 & 1 \\ k_2/m_2 & 0 & -(k_2+k_3)/m_2 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & C_4 \end{bmatrix}$$

and $C_i = 1$ ($i=1,2,3,4$) when X_i is an output and equals 0 otherwise.

A.1.3 Modal Frequencies and Shapes

Several important dynamic properties are best illustrated by considering a modal representation of the plant. The homogeneous ($F(t)=0$) solutions of Eq's A.1-1,-2 are harmonic and have the general form:

$$x_i = \chi_i C \cos(\omega t + \phi) \quad (i=1,2) \quad (\text{A.1-30})$$

Substituting Eq A.1-30 into Eq A.1-3 yields an *eigenvalue problem* in the form:

$$\begin{bmatrix} k_1+k_2-m_1\omega^2 & -k_2 \\ -k_2 & k_2+k_3-m_2\omega^2 \end{bmatrix} \begin{Bmatrix} \chi_1 \\ \chi_2 \end{Bmatrix} = 0 \quad (\text{A.1-31})$$

which has a nontrivial solution if and only if

$$\begin{vmatrix} k_1+k_2-m_1\omega^2 & -k_2 \\ -k_2 & k_2+k_3-m_2\omega^2 \end{vmatrix} = m_1 m_2 \omega^4 - [(k_2+k_3)m_1 + (k_1+k_2)m_2] \omega^2 + k_1 k_2 + k_1 k_3 + k_2 k_3 = 0 \quad (\text{A.1-32})$$

For the two degrees of freedom system this characteristic equation may be solved explicitly via the quadratic solution. E.g. for the case of $k_1=k_2=k_3=k$:

$$\omega^2 = \frac{k(m_1+m_2) \pm k \sqrt{m_1^2 - m_1 m_2 + m_2^2}}{m_1} \quad (\text{A.1-33})$$

For $m_1=m_3=m$, $k_1=k_3=k$, k_2 arbitrary:

$$\omega^2 = \left[\frac{k}{m}, \frac{k+2k_2}{m} \right] \quad (\text{A.1-34})$$

which of course becomes :

$$\omega^2 = \left[\frac{k}{m}, \frac{3k}{m} \right] \quad (\text{A.1-35})$$

when $k_2=k$.

Consider also the following cases – all for which (i.e. Eq's A.1-36 through A.1-41) $k_3=0$.

When $m_1=m_2$, $k_1=k_2$:

$$\omega^2 = \frac{k_2}{J_2} \left[\frac{3 \pm \sqrt{5}}{2} \right] \quad (\text{A.1-36})$$

For $m_1=2m_2$, $k_1=2k_2$:

$$\omega^2 = \frac{k_2}{m_2} [2 \pm \sqrt{3}] \quad (\text{A.1-37})$$

Or, in general:

$$\omega^2 = \frac{k_2}{m_2} \left[\frac{1}{2} + a\beta \pm \sqrt{\frac{1}{4} + b\beta + c\beta^2} \right] \quad (\text{A.1-38})$$

where:

$$a = \frac{1+\alpha}{2}, \quad b = \frac{1-\alpha}{2}, \quad c = \frac{1+\alpha^2}{4} \quad (\text{A.1-39})$$

and:

$$\alpha = k_1/k_2 \quad (\text{A.1-40a})$$

$$\beta = m_2/m_1 \quad (\text{A.1-37b})$$

For the case $k_1 = 0$ ($\alpha = 0$):

$$\omega^2 = \frac{k_2}{m_2} [0, 1+\beta] \quad (\text{A.1-41})$$

The $\omega^2 = 0$ solution corresponds to the *rigid body mode* in which m_1 and m_2 move at constant velocity.

The normalized characteristic roots may be plotted as functions of the mass and stiffness ratios α & β as shown in Figure A-2. The actual modal frequencies are readily found by multiplying the values read from the figure by the ratio $[k_2/m_2]^{1/2}$.

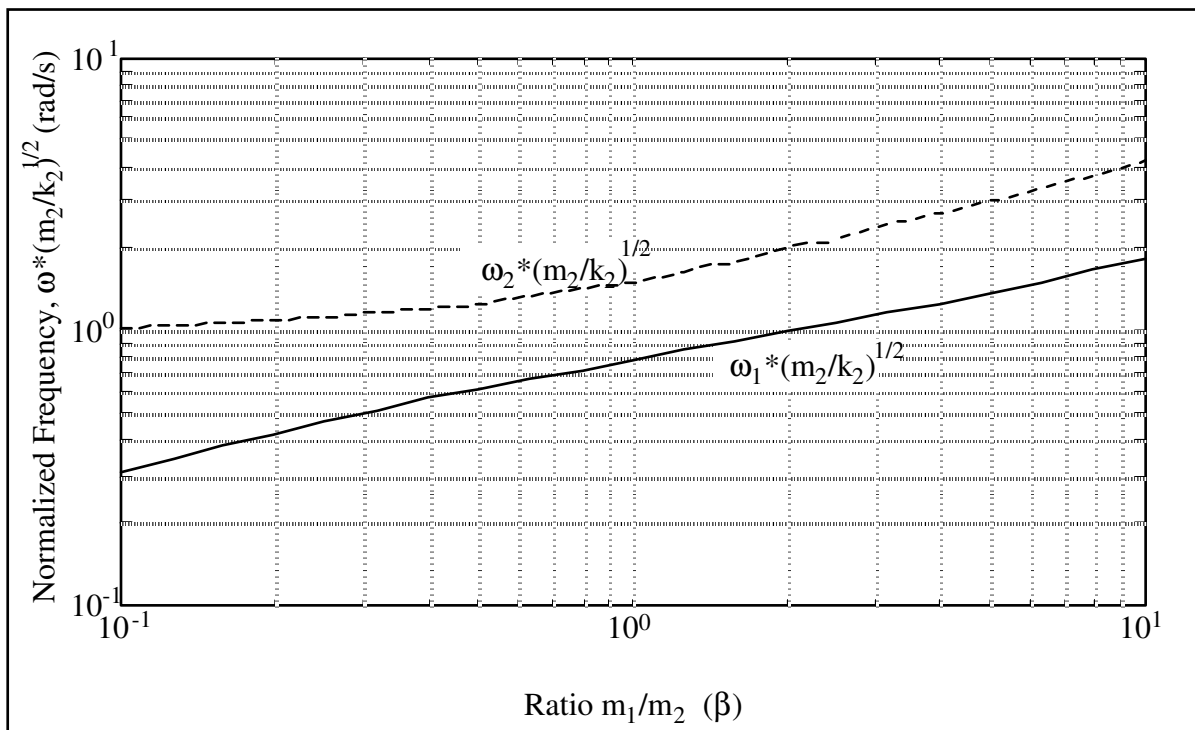


Figure A-2 Dependence of ω_1 and ω_2 on β . $\alpha = 1$

The *mode shapes* (or the relative amplitudes and directions of x_1 & x_2 when the plant is oscillating at the modal frequencies) are found by solution of Eq. A.1-31 as:^[66]

$$\frac{\chi_2}{\chi_1} = \frac{k_1 + k_2 - m_1 \omega^2}{k_2} \quad (\text{A.1-42})$$

where the constant C of Eq A.1-30 scales both amplitudes and therefore the amplitude χ_1 is arbitrary.

The special cases considered above yield:

$$\text{For } k_1=k_2=k_3=k: \quad \frac{\chi_2}{\chi_1} = 2 - \frac{(m_1+m_2) \pm \sqrt{m_1^2 - m_1 m_2 + m_2^2}}{m_2} \quad (\text{A.1-43})$$

For $m_1=m_3=m$, $k_1=k_3=k$, k_2 arbitrary:

$$\frac{\chi_2}{\chi_1} = 1 \left(\omega = \sqrt{\frac{k}{m}} \right), \quad -1 \left(\omega = \sqrt{\frac{k+2k_2}{m}} \right) \quad (\text{A.1-44})$$

which shows that in the first mode, m_1 and m_2 move in unison, while in the second their displacements are in pure opposition.^[67] Note that these mode shapes of course also apply when $k_2=k$.

For the several cases when $k_3=0$:

$$\text{For } m_1=m_2, k_1=k_2: \quad \frac{\chi_2}{\chi_1} = \frac{-1 \pm \sqrt{5}}{2} \quad (\text{A.1-45})$$

$$\text{For } m_2=2m_1, k_2=2k_1: \quad \frac{\chi_2}{\chi_1} = -1 \pm \sqrt{3} \quad (\text{A.1-46})$$

$$\text{For } m_2=m_1/\beta, k_1=0: \quad \frac{\chi_2}{\chi_1} = 1, -\beta \quad (\text{A.1-47})$$

Eq. A.1-47 shows that with k_1 and k_3 absent, there is one mode called the *rigid body mode* in which x_1 & x_2 move in unison and an oscillatory mode in which the coordinate amplitude ratio is proportional to the respective masses.

A.1.4 Modal Coordinate State Space Representation

It may be shown via orthogonality properties that the system motion may at any time be expressed as a linear combination of the mode shapes, i.e.: (Ref Mierovitch, Greenwood)

$$x_i(t) = \sum_{j=1}^2 \chi_{ij} q_j(t) \quad i = 1, 2 \quad (\text{A.1-50})$$

where χ_{ij} is the amplitude $(C^{\chi})_{ij}$ (see Eq. (A.1-30)) associated with the j th mode which has been normalized by:

$$\boldsymbol{\chi} \mathbf{m} \boldsymbol{\chi} = \mathbf{I} \quad (\text{A.1-51})$$

with

$$\boldsymbol{\chi} = \begin{bmatrix} \chi_{11} & \chi_{12} \\ \chi_{21} & \chi_{22} \end{bmatrix} \quad (\text{A.1-52})$$

Here I is the identity matrix. For our case, Eq.(A.1-51) results in:

$$\chi_{1j} = (C\chi_1)_j = \frac{1}{\sqrt{m_1 + m_2 (\chi_2/\chi_1)_j^2}} \quad (\text{A.1-53})$$

and χ_{2j} is found via multiplying χ_{1j} by the amplitude ratio Eq.(A.1-39) for the j th mode.

The variables $q_j(t)$ in Eq.(A.1-50) are called *modal coordinates*. Eq.(A.1-50) shows that $q_j(t^*)$ may be thought of as the amount that the j^{th} mode shape contributes to the system motion at some time t^* .

The equations of motion may then be expressed as (e.g. via modal decoupling or from orthogonality and Lagrange's equation^[68]):

$$q_j(t) + \omega_j^2 q_j(t) = \chi_{1j} F(t) \quad (j=1, 2) \quad (\text{A.1-54})$$

which may be realized in state space form using Eq.(A.1-50) as

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{A}\mathbf{q} + \mathbf{B}F(t) \\ \mathbf{x} &= \mathbf{C}\mathbf{q} \end{aligned} \quad (\text{A.1-55})$$

where:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ \dot{q}_1 \\ q_2 \\ \dot{q}_2 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_1^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_2^2 & 0 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \chi_{11} \\ 0 \\ \chi_{12} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \chi_{11} & 0 & \chi_{12} & 0 \\ \chi_{21} & 0 & \chi_{22} & 0 \end{bmatrix}$$

and x is as per Eq.(A.1-4) (The outputs \dot{x}_i are obtainable by shifting the first and third columns in C one column to the right.)

A.1.5 Transfer Functions

The plant transfer functions are found via the Laplace transform of Eq's(A.1-1,-2) which yield:

$$\frac{x_1(s)}{F(s)} = \frac{m_2 s^2 + k_1 + k_2}{D(s)} \quad (\text{A.1-60})$$

$$\frac{x_2(s)}{F(s)} = \frac{k_2}{D(s)} \quad (\text{A.1-61})$$

where:

$$D(s) = m_1 m_2 s^4 + (m_1(k_2 + k_3) + m_2(k_1 + k_2))s^2 + k_1 k_2 + k_1 k_3 + k_2 k_3 \quad (\text{A.1-62})$$

Note that x_1/F has two imaginary zeroes at $\omega = \pm j((k_1 + k_2)/m_2)^{1/2}$ resulting in a pole excess of two. x_2/F has no zeroes and hence a four pole excess. The roots to $D(s)$ are imaginary and of magnitude given by Eq.(A.1-32). The system is type zero (no zero valued roots to $D(s)$) except for the case $\{k_1 = k_3 = 0\}$ (implies rigid body mode present) which is type two.

An expanded transfer function form may be obtained through partial fraction methodology applied to Eq's (A.1-61,-62), or equivalently by Laplace transform of Eq.(A.1-54) combined per Eq.(A.1-50):

$$\frac{x_1(s)}{T(s)} = \frac{\chi_{11}^2}{s^2 + \omega_1^2} + \frac{\chi_{12}^2}{s^2 + \omega_2^2} \quad (\text{A.1-63})$$

$$\frac{x_2(s)}{T(s)} = \frac{\chi_{11}\chi_{21}}{s^2 + \omega_1^2} + \frac{\chi_{12}\chi_{22}}{s^2 + \omega_2^2} \quad (\text{A.1-64})$$

A.2 Practical Plant Model

A.2.1 Equations Of Motion

While the expressions obtained in Section A.1 are useful in understanding the predominant dynamic behavior, they neglect certain non-ideal effects which are in some cases significant.^[69] Among these are forces associated with motor cogging^[70] and Coulomb (or static) and viscous frictions from the gears, dashpot, and bearings. From Figure A-3, a more precise approximation of the system motion is:

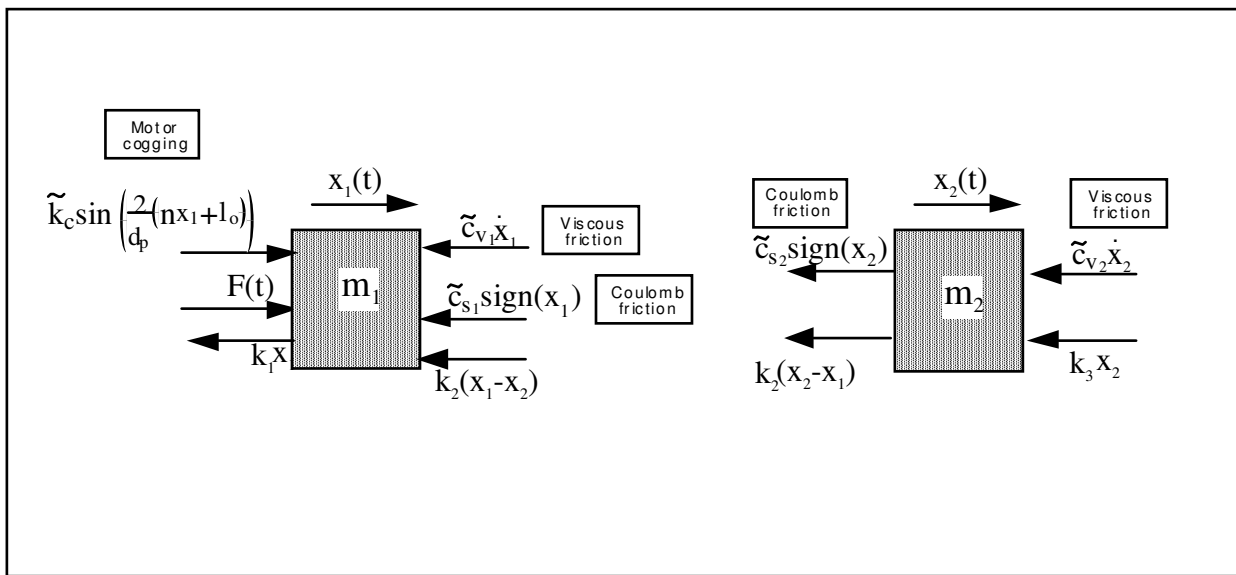


Figure A-3 Free Body Diagram Including Friction and Nonlinearities

$$m_1 \ddot{x}_1 + \tilde{c}_{v1} \dot{x}_1 + \tilde{k}_c \sin\left(\frac{2}{d_p}(nx_1 + l_o)\right) + \tilde{c}_{s1} \text{sign}(\dot{x}_1) + (k_1 + k_2)x_1 - k_2x_2 = F(t) \quad (\text{A.2-1})$$

$$m_2 \ddot{x}_2 + \tilde{c}_{v2} \dot{x}_2 + \tilde{c}_{s2} \text{sign}(\dot{x}_2) + (k_2 + k_3)x_2 - k_2x_1 = 0 \quad (\text{A.2-2})$$

which has nonlinear terms.

Reductions in the complexity of this model may be made. When at rest and with no acting forces, the cogging torque will tend to place the system in a point of stable equilibrium. If we define $x_1=0$ at such a point, then if Coulomb friction is small, $l_o \approx 0$. By linearization (e.g. the first two Taylor's series terms) about $x_1=0$ we have that for small displacements (say $|x_1| \leq 2$ cm) the equivalent cogging force is approximated as:

$$\tilde{k}_c \sin\left(\frac{2}{d_p}(nx_1 + l_o)\right) \cong \tilde{k}_c 2n \frac{x_1}{d_p} \triangleq k_c x_1 \quad (\text{A.2-3})$$

This may then be combined with the spring constant \tilde{k}_1 , to obtain a net spring constant as:

$$k_1 \triangleq \tilde{k}_1 + k_c \quad (\text{A.2-4})$$

An approximation of the Coulomb friction which is satisfactory in many cases is obtained by equating the resulting force to that supplied by viscous friction when operating at some nominal speed \dot{x}_{in} . From a system stability point of view, it is conservative to use a value \dot{x}_{in} at or near the highest expected operating speed (top speed is generally specified as a design requirement) so that the actual associated damping or energy dissipation is generally greater. So doing and combining with other viscous terms gives:

$$\tilde{c}_{si} \text{sign}(\dot{x}_i) \cong \underset{\text{conserv}}{c_{vi}^*} \dot{x}_{i\text{max}} \quad (i=1, 2) \quad (\text{A.2-5})$$

$$c_i \cong \tilde{c}_{vi} + c_{vi}^* \quad (i=1, 2) \quad (\text{A.2-6})$$

The above result in simplified equations of motion:

$$m_1 \ddot{x}_1 + c_1 \dot{x}_1 + (k_1 + k_2)x_1 - k_2x_2 = F(t) \quad (\text{A.2-7})$$

$$m_2 \ddot{x}_2 + c_2 \dot{x}_2 + (k_2 + k_3)x_2 - k_2 x_1 = 0 \quad (\text{A.2-8})$$

which may be expressed in a state space form:

$$\begin{aligned} \dot{x} &= Ax + BF(t) \\ Y &= Cx \end{aligned} \quad (\text{A.2-9})$$

where:

$$\begin{aligned} X &= \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -(k_1 + k_2)/m_1 & -c_1/m_1 & k_2/m_1 & 0 \\ 0 & 0 & 0 & 1 \\ k_2/m_2 & 0 & -(k_2 + k_3)/m_2 & -c_2/m_2 \end{bmatrix}, \\ B &= \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & C_4 \end{bmatrix} \end{aligned}$$

and the output notation is as in Eq A.1-20.

A.2.2 Modal Coordinate Representation

In the modal coordinate representation, plant damping is often approximated by assuming a viscous term for each mode, i.e.:^[71]

$$\ddot{q}_j + 2\zeta_j \omega_j \dot{q}_j + \omega_j^2 q_j = \chi_{1j} F(t) \quad (\text{A.2-11})$$

which motivates a state space realization with

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega_1^2 & -2\zeta_1 \omega_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_2^2 & -2\zeta_2 \omega_2 \end{bmatrix} \quad (\text{A.2-12})$$

and the remaining expressions and terms identical to Eq.(A.1-55). In the case $k_2 = 0$ ($\omega_1 = 0$), the 2,2 element in A becomes $-c^*$ which is equal to $-c/(m_1 + m_2)$ with c being the damping coefficient associated with the rigid body motion. (i.e. the rigid body equation is $(m_1 + m_2)\ddot{x}_i + c_{xi} \dot{x}_i = F$ ($i=1,2$). Again, this approximation is only valid for small c or for c_1 & c_2 proportional to m_1 & m_2 –see note below. When these criteria are met, then $c^* \approx (c_1 + c_2)/(m_1 + m_2)$)

A.2.3 Transfer Functions

From the Laplace transform of Eq's(A.2-8,9) we have:

$$\frac{x_1(s)}{F(s)} = \frac{m_2 s^2 + c_2 s + k_2 + k_3}{D(s)} \quad (\text{A.2-13})$$

$$\frac{x_2(s)}{F(s)} = \frac{k_2}{D(s)} \quad (\text{A.2-14})$$

where:

$$D(s) = m_1 m_2 s^4 + (m_1 c_2 + m_2 c_1) s^3 + (m_1 (k_2 + k_3) + m_2 (k_1 + k_2) + c_1 c_2) s^2 + (c_1 (k_2 + k_3) + c_2 (k_1 + k_2)) s + k_1 k_2 + k_1 k_3 + k_2 k_3 \quad (\text{A.2-15})$$

The expanded transfer function forms resulting from Eq.(A.2-11) are:

$$\frac{x_1(s)}{F(s)} = \frac{\chi_{11}^2}{s^2 + 2\zeta_1 \omega_1 s + \omega_1^2} + \frac{\chi_{12}^2}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} \quad (\text{A.2-16})$$

$$\frac{x_2(s)}{F(s)} = \frac{\chi_{11} \chi_{21}}{s^2 + 2\zeta_1 \omega_1 s + \omega_1^2} + \frac{\chi_{12} \chi_{22}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} \quad (\text{A.12-17})$$

where the χ_{ij} 's may be obtained from Eq.(A.1-53) for small ζ_j .

The above models are useful when the plant parameters m_i , k_i , & c_i are known or are measurable explicitly. In many cases, the plant is identified via its natural frequencies, damping ratios, and gains. Either of the above transfer function types may be expressed in the form:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (\text{A.2-18})$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)} \quad (\text{A.2-19})$$

where the gains K_1 & K_2 , are nominally equal to $1/m_1$ and $k_2/m_1 m_2$ respectively (but may be measured more directly). Equations (A.2-18,-19) model the plant in terms of parameters which are either readily measured or derived and therefore are useful for practical implementation.

Again, for the case $k_1 = k_3 = 0$ ($\omega_1 = 0$), the coefficient $2\zeta_1 \omega_1$ (or $2\zeta_{p1} \omega_{p1}$) becomes c^* which is nominally equal to $c/(m_1 + m_2)$. Thus the first denominator term in the right hand side of each of Eq's(A.2-16 through -19) becomes nominally equivalent to $1/[(m_1 + m_2)s^2 + cs]$.

Measurement of parameters to establish the explicit plant transfer functions for Model 210 is covered in Section 6.1 of this manual.

A.3 One Degree of Freedom Plants

The dynamic models for the one degree of freedom (DOF) "free-clamped" plant (see Fig. A-1d) follow from Eq's(A.1-1 & A.2-8) by setting $x_2 = 0$. The results are stated below where the "rigid body" 1 DOF plant may be found by taking $k_1 = 0$ in the appropriate equations. The undamped expressions are obtained by setting ζ and c_1 to zero.

The equation of motion is

$$m_1 \ddot{x}_1 + c_1 \dot{x}_1 + k_1 x_1 = F(t) \quad (\text{A.3-1})$$

which has the transfer function:

$$\frac{x_1(s)}{F(s)} = \frac{1}{m_1 s^2 + c_1 s + k_1} \quad (\text{A.3-2})$$

or

$$\frac{x_1(s)}{F(s)} = \frac{K_1}{s^2 + 2\zeta\omega s + \omega^2} \quad (\text{A.3-3})$$

The state space realization analogous to Eq (A.2-10) is comprised of the first 2x2, 2x1, 2x1, and 2x2 submatrices of the respective A, B, X, and C matrices in (A.2-10).

The modal coordinate state space realization has system matrix, A, that is the first 2x2 submatrix of Eq.(A.2-12) and with consistent B, C, and \mathbf{q} matrices:

The characteristic root is $\omega^2 = k_1/m_1$, and the single modal weighting is $\chi_{11} = 1/\sqrt{m_1}$.

A.4 Three Degree of Freedom Plant (ECP Model 210a)

The dynamic models for the three degree of freedom (DOF) spring/mass apparatus are given below without derivation. They are however, a straightforward extension of the material in Sections A.1 and A.2. The analogous two DOF equations are given in brackets.

The damped plant model is shown graphically in Figure A-4. Note that the plant may also be configured into any of the two or one DOF types shown in Figure A-1 .

As before for the undamped plant:

$$\mathbf{m}\mathbf{x} + \mathbf{k}\mathbf{x} = \mathbf{F}$$

with [A.1-4]

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \mathbf{m} = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_2 \end{bmatrix}, \quad \mathbf{k} = \begin{bmatrix} k_1+k_2 & -k_2 & 0 \\ -k_2 & k_2+k_3 & -k_3 \\ 0 & -k_3 & k_3+k_4 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} F \\ 0 \\ 0 \end{bmatrix} \quad (\text{A.4-1})$$

which may be expressed in the state space form of Eq (A.2-10) with:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -(k_1+k_2)/m_1 & -c_1/m_1 & k_2/m_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ k_2/m_2 & 0 & -(k_2+k_3)/m_2 & -c_2/m_2 & k_3/m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & k_3/m_3 & 0 & -(k_3+k_4)/m_3 & -c_3/m_3 \end{bmatrix}, \quad (\text{A.4-1})$$

$$\mathbf{X} = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \\ x_3 \\ \dot{x}_3 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1/m_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} C_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & C_6 \end{bmatrix}$$

with the output notation as in Eq (A.1-20).

The Eigenvalue expression [A.1-31] becomes:

$$\begin{bmatrix} k_1+k_2-m_1\omega^2 & -k_2 & 0 \\ -k_2 & k_2+k_3-m_2\omega^2 & -k_3 \\ 0 & -k_3 & k_3+k_4-m_3\omega^2 \end{bmatrix} \begin{Bmatrix} \chi_1 \\ \chi_2 \\ \chi_3 \end{Bmatrix} = 0 \quad (\text{A.4-2})$$

with characteristic equation:

$$\begin{aligned} & m_1 m_2 m_3 \omega^6 + [m_1 m_2 (k_3 + k_4) + m_1 m_3 (k_2 + k_3) + m_2 m_3 (k_1 + k_2)] \omega^4 \\ & + [m_1 (k_2 (k_3 + k_4) + k_3 k_4) + m_2 ((k_1 + k_2) (k_3 + k_4)) + m_3 (k_1 (k_2 + k_3) + k_2 k_3)] \omega^2 \\ & + k_1 k_2 k_3 + k_1 k_2 k_4 + k_1 k_3 k_4 + k_2 k_3 k_4 = 0 \end{aligned} \quad (\text{A.4-3})$$

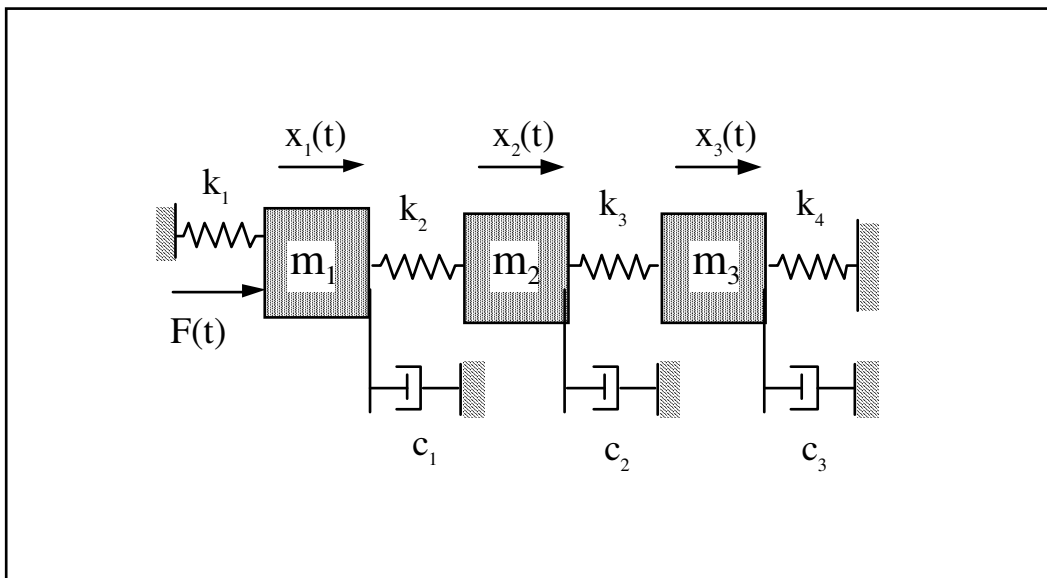


Figure A-4. 3 DOF Plant (damped model)

When $k_1 = k_4 = 0$, Eq. A.4-3 contains ω^2 as a common factor and hence a rigid body mode ($\omega = 0$) is present. Equations (A.4-2 & -3) are used to solve for the characteristic frequencies ω and the amplitude ratios for each of the three modes. The ratio χ_2/χ_1 is given by Eq(A.1-42) as before, and:

$$\frac{\chi_3}{\chi_1} = \frac{k_3(k_1 + k_2 - m_1\omega^2)}{k_2(k_3 + k_4 - m_3\omega^2)} \quad (\text{A.4-4})$$

(Note that amplitude ratios are both equal to one for the rigid body mode.) The modal amplitudes are

normalized by the 3x3 analog of Eq's(A.1-51,52).

Including the effects of damping (the undamped expressions are obtained via setting the ζ 's and c^* to zero in the following equations) we have for the modal coordinate state space form [A.1-55, A.2-11]:

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{A}\mathbf{q} + \mathbf{B}\mathbf{F}(t) \\ \mathbf{x} &= \mathbf{C}\mathbf{q}\end{aligned}\tag{A.4-5}$$

where

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_1 \\ \dot{\mathbf{q}}_1 \\ \mathbf{q}_2 \\ \dot{\mathbf{q}}_2 \\ \mathbf{q}_3 \\ \dot{\mathbf{q}}_3 \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\omega_1^2 & -2\zeta_1\omega_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\omega_2^2 & -2\zeta_2\omega_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\omega_3^2 & -2\zeta_3\omega_3 \end{bmatrix},$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \chi_{11} \\ 0 \\ \chi_{12} \\ 0 \\ \chi_{13} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \chi_{11} & 0 & \chi_{12} & 0 & \chi_{13} & 0 \\ \chi_{21} & 0 & \chi_{22} & 0 & \chi_{23} & 0 \\ \chi_{31} & 0 & \chi_{32} & 0 & \chi_{33} & 0 \end{bmatrix}$$

when $k_1 = k_4 = 0$, the system matrix becomes

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -c^* & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\omega_2^2 & -2\zeta_2\omega_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\omega_3^2 & -2\zeta_3\omega_3 \end{bmatrix}$$

The plant transfer functions are:

$$\frac{x_1(s)}{F(s)} = \frac{N_1(s)}{D(s)} \quad (\text{A.4-6})$$

$$\frac{x_2(s)}{F(s)} = \frac{N_2(s)}{D(s)} \quad (\text{A.4-7})$$

$$\frac{x_3(s)}{F(s)} = \frac{N_3(s)}{D(s)} \quad (\text{A.4-8})$$

where:

$$N_1(s) = m_2 m_3 s^4 + (m_2 c_3 + m_3 c_2) s^3 + (m_2(k_3 + k_4) + m_3(k_2 + k_3) + c_2 c_3) s^2 + (c_2(k_3 + k_4) + c_3(k_2 + k_3)) s + k_2 k_3 + k_2 k_4 + k_3 k_4 \quad (\text{A.4-9})$$

$$N_2(s) = k_2 (m_3 s^2 + c_3 s + k_3 + k_4) \quad (\text{A.4-10})$$

$$N_3(s) = k_2 k_3 \quad (\text{A.4-11})$$

$$\begin{aligned}
D(s) = & m_1 m_2 m_3 s^6 + [m_1(m_2 c_3 + m_3 c_2) + m_2 m_3 c_1] s^5 \\
& + [m_1(m_2(k_3 + k_4) + m_3(k_2 + k_3) + c_2 c_3) + m_2(m_3(k_1 + k_2) + c_1 c_3) + m_3 c_1 c_2] s^4 \\
& + \left[\frac{m_1(c_2(k_3 + k_4) + c_3(k_2 + k_3)) + m_2(c_1(k_3 + k_4) + c_3(k_1 + k_2))}{m_3(c_1(k_2 + k_3) + c_2(k_1 + k_2)) + c_1 c_2 c_3} \right] s^3 \\
& + \left[\frac{m_1(k_2(k_3 + k_4) + k_3 k_4) + m_2((k_1 + k_2)(k_3 + k_4)) + m_3(k_1(k_2 + k_3) + k_2 k_3)}{c_1 c_2(k_3 + k_4) + c_1 c_3(k_2 + k_3) + c_2 c_3(k_1 + k_2)} \right] s^2 \\
& + [c_1(k_2(k_3 + k_4) + k_3 k_4) + c_2((k_1 + k_2)(k_3 + k_4)) + c_3(k_1(k_2 + k_3) + k_2 k_3)] s \\
& + k_1 k_2 k_3 + k_1 k_2 k_4 + k_1 k_3 k_4 + k_2 k_3 k_4
\end{aligned} \tag{A.4-12}$$

Note that this complicated expression for $D(s)$ requires the measurement or estimation of ten dynamic parameters for its coefficients.

The expanded transfer function forms which derive from Eq.(A.4-5) are [A.2-16,-17]:

$$\frac{x_3(s)}{F(s)} = \frac{\chi_{11} \chi_{31}}{s^2 + 2\zeta_1 \omega_1 s + \omega_1^2} + \frac{\chi_{12} \chi_{32}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{33}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-13}$$

$$\frac{x_2(s)}{F(s)} = \frac{\chi_{11} \chi_{21}}{s^2 + 2\zeta_1 \omega_1 s + \omega_1^2} + \frac{\chi_{12} \chi_{22}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{23}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-14}$$

$$\frac{x_3(s)}{F(s)} = \frac{\chi_{11} \chi_{31}}{s^2 + 2\zeta_1 \omega_1 s + \omega_1^2} + \frac{\chi_{12} \chi_{32}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{33}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-15}$$

which have the following forms when $k_1 = k_4 = 0$ (rigid body mode present):

$$\frac{x_3(s)}{F(s)} = \frac{\chi_{11} \chi_{31}}{s^2 + c^* s} + \frac{\chi_{12} \chi_{32}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{33}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-13a}$$

$$\frac{x_2(s)}{F(s)} = \frac{\chi_{11} \chi_{21}}{s^2 + c^* s} + \frac{\chi_{12} \chi_{22}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{23}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-14a}$$

$$\frac{x_3(s)}{F(s)} = \frac{\chi_{11} \chi_{31}}{s^2 + c^* s} + \frac{\chi_{12} \chi_{32}}{s^2 + 2\zeta_2 \omega_2 s + \omega_2^2} + \frac{\chi_{13} \chi_{33}}{s^2 + 2\zeta_3 \omega_3 s + \omega_3^2} \tag{A.4-15a}$$

The more practical forms expressed in terms of measurable frequencies, damping, and gains are [A.2-18,-19]:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_{z1} \omega_{z1} s + \omega_{z1}^2)(s^2 + 2\zeta_{z2} \omega_{z2} s + \omega_{z2}^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)(s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \tag{A.4-16}$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2 (s^2 + 2\zeta_z \omega_z s + \omega_z^2)}{(s^2 + 2\zeta_{p1} \omega_{p1} s + \omega_{p1}^2)(s^2 + 2\zeta_{p2} \omega_{p2} s + \omega_{p2}^2)(s^2 + 2\zeta_{p3} \omega_{p3} s + \omega_{p3}^2)} \tag{A.4-17}$$

$$\frac{x_3(s)}{F(s)} = \frac{K_3}{(s^2 + 2\zeta_{p1}\omega_{p1}s + \omega_{p1}^2)(s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2)(s^2 + 2\zeta_{p3}\omega_{p3}s + \omega_{p3}^2)} \quad (\text{A.4-18})$$

which have rigid body forms:

$$\frac{x_1(s)}{F(s)} = \frac{K_1 (s^2 + 2\zeta_{z1}\omega_{z1}s + \omega_{z1}^2)(s^2 + 2\zeta_{z2}\omega_{z2}s + \omega_{z2}^2)}{s(s + c^*)(s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2)(s^2 + 2\zeta_{p3}\omega_{p3}s + \omega_{p3}^2)} \quad (\text{A.4-16a})$$

$$\frac{x_2(s)}{F(s)} = \frac{K_2 (s^2 + 2\zeta_z\omega_zs + \omega_z^2)}{s(s + c^*)(s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2)(s^2 + 2\zeta_{p3}\omega_{p3}s + \omega_{p3}^2)} \quad (\text{A.4-17a})$$

$$\frac{x_3(s)}{F(s)} = \frac{K_3}{s(s + c^*)(s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2)(s^2 + 2\zeta_{p3}\omega_{p3}s + \omega_{p3}^2)} \quad (\text{A.4-18a})$$

Parameter identification to establish the explicit plant transfer functions or state space realizations for Models 210 & 210a is covered in Section 6.1 of this manual.

-
- [1] For Model 210a the model order may be as high as six with either four, two, or no zeros.
- [2] The system is also available in a PC bus installation form in which the DSP based real-time controller resides in the PC and all other control unit hardware remains in a separate box. This form has faster PC/controller communication rates. (Controller speed is unaffected.)
- [3] A faster computer, such as a ≥ 66 Mhz 486 with the real-time controller on the PC bus provides a much more expeditious working environment than a 386 or 286 and/or RS232 controller/PC communication. Real-time control speed, however, is unaffected.
- [4] Its fastest to simply double-click on the desired file.
- [5] For many designs, the value $T_s = .00442$ sec is a good midpoint between smaller sample periods that may result in signal noise induced by spatial quantization and larger periods that may cause excessive phase lag.
- [6] An often used rule of thumb is that the continuous time approximation of sampled data systems is acceptably accurate if the sampling frequency is at least 10 times the system bandwidth. (This rule is not always conservative however, see Section 6.7 & 6.7i). Since the attainable closed loop bandwidths for the system are generally less than 20 Hz, sampling rates above 200 Hz usually provide results that are indistinguishable between equivalent continuous and discrete controller designs. I.e. for sampling rates above 200 Hz, the user may generally design and specify the controller in continuous time with no measurable difference in system behavior than if the controller were designed in discrete time.
- [7] Note that in previewing the discrete generalized form of a continuous controller you should select Discrete Time , General Form, then Setup Algorithm. If instead the sequence Discrete Time, Preview In General Form, is used, then the selected discrete time algorithm (the one with the red dot next to it and which will not generally contain parameters that correspond to the continuous time design) will be previewed. Subsequent "Implementing" will then download the wrong design.
- [8] You may notice the term r_1 in the Continuous Time General Form has a default value 0.0000002 whenever PD, PID, PID+ Notch, or State Feedback are selected. (in this case you would enter the General Form via the Preview In General Form button). This adds a pole at very high frequency and is of no practical consequence to system stability or performance. It is necessary to make the S(s)/R(s) and T(s)/R(s) blocks proper when implementing the differentiator terms in the above mentioned control forms.
- [9] This format may be produced in Matlab® using the `fprintf` function.
- [10] The system contains safeguards to prevent unsafe operations in most cases. If a hardware or software limit is exceeded, the Controller Status display on the Background Screen will indicate Limit Exceeded. In this event, the user should Reset Controller (Utility menu), and re-Implement (Command menu) using an appropriate (safe) set of control coefficients.
- [11] The software is set to saturate control effort at ± 5 V. If this amplitude is exceeded, the input shape amplitudes or accelerations as appropriate should be reduced.
- [12] If the specified "impulse" duration becomes long enough, the resulting force becomes more step-like than impulsive. Thus the Setup Impulse dialog box may also be used for Step input shapes where the dwell (zero excitation) period may be specified independently of the step duration.
- [13] Sweeping through resonances is very useful in visualizing the frequency response dynamics, but must be done at a sufficiently low input amplitude. When viewing open loop sine sweep results, it is often best to view velocity rather than position data to reduce position drift effects. By selecting a relatively long sweep period, the transient effects of frequency change are minimized and the true frequency response is best approximated.
- [14] Thus, if 3 force cycles per cm were desired, the user would specify $3 \times 7.06 = 21.2$ cycles per encoder revolution.
- [15] The bracketed rows end in semicolons so that the entire file may be read as an array in Matlab by running it as a script once the header is stripped i.e. the script should be: `<array name>= [exported data file]`. Variable values over time are the columns of this array; the rows are the variable value set at successive sample numbers.
- [16] In some cases, you will need to "drag" the Executing Input Shape box out of the way to see the plot during the maneuver. This is practical for longer duration maneuvers.
- [17] See Chapter 4 for a description of component functionality (e.g. actuators and sensors) and I/O processing (e.g. motor commutation and encoder pulse counting)
- [18] This microswitch causes the controller to be disabled during control implementation.
- [19] The majority of this section (2.3.1) pertains to the PC bus installation of the real-time controller. For the controller box/RS-232 version, the control box should generally be powered on before entering the executive software.
- [20] This applies to the PC-bus installation only. For the controller in the Control Box, the RS-232 cable must be connected between the Control Box and the PC.
- [21] The Logarithmic frequency / Db data may be also plotted from the linear sweep which results in fewer cycles at low frequency and more at high frequency.
- [22] Note that if power is turned off to the control box, the travel limit switches will appear to the real time controller as though they are depressed even when they are not.

- [23] Since no trajectory is being input at this point, the system is regulating about a reference input value of zero.
- [24] Each pulse corresponds to a small displacement (1/16,000 revolution in this case) at the plant output.
- [25] The Executive program's Setup Control Algorithm dialog box allows the user to alter the sampling period.
- [26] We qualify the validity of this assumption later when we discuss the PI current loops.
- [27] That is the mass position encoder resolution effectively becomes 16,000 counts per revolution. The effective resolution is then $16,000 \text{ counts}/(2\pi r_p)$ where r_p is the encoder pinion pitch radius – i.e. resolution = 1604.1 counts/cm.
- [28] You may “zoom” the plot via Axis Scaling for more precise measurement in various areas. For an even greater precision, the data may be examined in tabular numerical form – see Export Raw Data, Section 2.1.7.3.
- [29] Note that the calculated masses m_{c1} and m_{c2} will include the reflected inertias of all connected elements – e.g. motor pinion and armature.
- [30] Step 12 may be done later, away from the laboratory, if necessary.
- [31] The resulting value for k should be very close to that measured when considering the second mass case. You may use the average of the two for your identified k value.
- [32] In Model 210 configurations where the third mass is used, the inertia and damping parameters of the third carriage may generally be assumed to be the same as for the second carriage. For more precise results, steps 5 through 9 and 12 may be repeated for the third mass.
- [33] Some small deceleration will exist due to friction.
- [34] For more precise measurement you may “zoom in” on this region of the plot using Axis Scaling in the Plotting menu.
- [35] It is possible to read the accelerations directly by plotting Encoder #1 acceleration. This data, obtained by double numerical differentiation, is typically somewhat noisy however. The student may want to verify this by observing the acceleration plot
- [36] It contains software gain also. This software gain, k_s is used to give higher controller-internal numerical resolution and improves encoder pulse period measurement for very low rate estimates.
- [37] These are the counts that are actually operated on in the control algorithm. i.e. The system input (trajectory) counts and encoder counts are multiplied by 32 prior to control law execution.
- [38] The student may want to later verify that for the relatively high amount of control damping in the scheme that follows – induced via the parameter k_d – that the plant damping is very small.
- [39] Here due to friction the system, which is ideally quasi-stable (characteristic roots on the $j\omega$ axis), remains stable for small k_p . For larger values, the time delay associated with sampling may cause instability.
- [40] E.g. a large following error at the time of implementation.
- [41] For the discrete implementation you must divide the resulting value by T_s for the controller input value. Here, since the PD controller is improper, the backwards difference transformation: $s = (1-z^{-1})/T_s$ is used.
- [42] Recall that for discrete implementation, you must divide the k_d values by T_s for controller input.
- [43] For discrete implementation you must multiply the resulting value of k_i by T_s before inputting into the controller.
- [44][44] For discrete implementation, do not input $k_i > 0.3 * T_s$.
- [45] For the purposes here, consider *bandwidth* to be the frequency in the sine sweep data at which the system attenuates below 1/2 (-6Db) of its low frequency amplitude
- [46] In practice, friction and its effect on system response are much more complex. This assumption however is valid in discussing the effect of the integrating term.
- [47] I.e. with pairs of poles at 7.07 and 11.3 Hz and damping of 70.7% ($\zeta = \sqrt{2}$). The first pole pair is necessary to make the filter proper and is placed sufficiently beyond the zero pair in $N_n(s)$ so as not to adversely effect its effectiveness in canceling the corresponding poles in $D_n(s)$. The second pole acts as a cascaded low pass filter to attenuate higher frequency noise.
- [48] For discrete time design, convert your notch filter design to the z domain using the Tustin (bilinear) transformation:
- $$s = \frac{2}{T_s} \frac{1-z^{-1}}{1+z^{-1}}$$
- [49] $r(s)$ will be the same as $D_n(s)$ and $s(s)$ & $t(s)$ will be identical and equal to $k_p N_n(s)$. Upon selecting Preview In General Form the algorithm is also converted to discrete time form (by the Tustin transform for N_n and D_n , backwards difference for any derivative terms). In this way the discrete time equivalent controller may be viewed in the discrete time General Form controller box.
- [50] Relatively small step sizes may be necessary here to prevent DAC saturation during the initial step response transient. The student may wish to view the control effort during these trials. Saturation occurs at the maximum DAC output of 5.0 V.
- [51] This small value is needed to provide a proper transfer function for bilinear transformation and subsequent discrete control implementation whenever T/R and S/R are used to implement a differentiator. Its small value results in a pole many decades beyond the system bandwidth and is of no practical implication to system modeling or performance. This coefficient may be set to zero here, but should generally remain.
- [52] The notation here is the obvious one.
- [53] This has poles of magnitude $|s| = 2.5 \text{ Hz}$ that lie at 135, 180, and 225 deg. It is similar to a third order Butterworth polynomial but

somewhat more damped.

^[54]You may also select Control Effort if you wish to later observe this value .

^[55]By considering collocated rate an output, the collocated rate feedback gain K_2 is increased (over the result with $C_{12}=0$) in the LQR solution and hence system stability is enhanced. This at the price of minimizing error on the x_2 output.

^[56]See for example Kwakernaak and Sivan, "Linear Optimal Control Systems", Wiley & Sons, 1972.

^[57] K_1 and K_3 scale control effort proportional to position errors, K_2 and K_4 scale control effort proportional to the respective velocities. Excessive values of K_1 or K_3 can lead to low stability margin and in the presence of time delays, instability. Large K_2 or K_4 cause excessive noise propagation and lead to "twitching" of the system. - see Section 6.8

^[58]If using discrete time implementation, be sure to divide your K_2 and K_4 values by T_s before entering them.

^[59] This yields the margins as if the additional phase lag or gain were to occur in the control input (i.e. k_{hw} block), or uniformly among the outputs. This is not necessarily the case in practice, but does provide a general measure of stability margin.

^[60]In either case, the DAC is generally programmed to saturate below or at its full scale output. Without programmed saturation the DAC, which typically accepts two's-complement input numbers, would output full scale voltage of reverse polarity upon exceeding its input range. This would clearly be problematic!

^[61] The natural frequency and damping ratio will vary depending on the particular system's k_{hw} .

^[62]The DSP controller and imbedded control firmware provide very high data and parameter resolution including a 16 bit DAC, and 48 bit multiplication with 96 bit products. These combined with internal data scaling for numerical conditioning make this system immune to many quantization related problems.

^[63] For the units used in this manual, the resolution is the carriage displacement encoder pulse, i.e. approximately 0.0004 cm/count..

^[64]Eq.(6.8-5) should not be viewed as a predictor of noise onset. Many other effects such as system bandwidth, mechanical backlash, and nonlinear behavior all influence noise propagation and generally have a different effect at different sampling frequency. For a given system and controller, however, high frequency torque (or control effort) quantization is often a dominant factor in noise propagation.

^[65]Or more generally with increased high bandwidth gain.

^[66]Eq A.1-31 is over-determined in that two equations exist for one unknown. Another solution is $\chi_2/\chi_1 = k_2/(k_2+k_3 - m_2\omega^2)$ which yield identical χ_2/χ_1 values for solutions ω^2 to Eq A.1-32 (or -37).

^[67]This phenomenon is readily demonstrated on the physical apparatus.

^[68]See for example "Elements Of Vibration Analysis", L. M. Mierovitch, McGraw-Hill Book Co., 1975; or "Principles Of Dynamics", D. T. Greenwood, Prentice-Hall Inc., 1965.

^[69]It is important to keep in mind the difference between the physical system and mathematical approximations of its behavior. Virtually no physical system follows a mathematical model exactly. In some cases even the dominant dynamic behavior evades mathematical description. It is very often possible, however to model the system sufficiently well for control purposes. The system here is relatively amenable to modeling with the ideal linear behavior dominating the plant dynamics. The generation of a plant model that is of minimal complexity but which represents the salient plant dynamics through the ensuing control bandwidth is central to effective control implementation.

^[70]Motor cogging produces the torque "bumps" felt when rotating the motor under a zero torque (force at gear rack) command with the amplifier active and is associated with brushless motor phasing and rotor magnet / stator field interaction. While this is not a true sinusoid, it may be approximated by $k_c \sin(2n(x_1/d_p + l_o))$ where d_p is the drive pinion pitch diameter, l_o is the distance from the stable cogging detent position to $x_1=0$, and n is the number of motor poles, i.e. $n = 2$ for Model 210.

^[71]This damping has a different form than that expressed in Eq's(A.2-6,-7) and in general is not literally representative of physical damping sources in the system. In order for viscous damping to exist and the modes to remain uncoupled, there must be an associated \dot{x} coefficient matrix C (Eq.(A.1-3))which is proportional to either x or k - this is called *proportional damping*.

Thus damping will generally couple the modes to some extent so that an unforced system, initially experiencing single mode motion, will exhibit motion of the other mode(s) in time. With large influence of non proportional damping, the modal motion concept is wholly invalid. For small values of ζ_j however, Eq.(A.2-10) is a useful approximation where the damping may be "visualized" as causing a decay of modal amplitude in the unforced plant.