

AD-A154 076

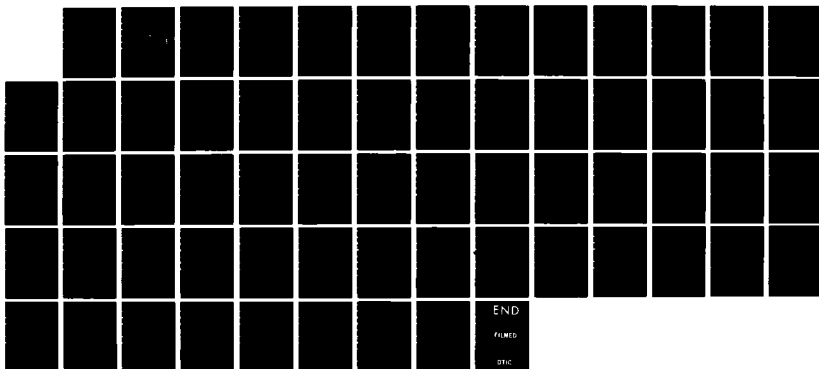
INTERACTIVE COMPUTER PROGRAM FOR THE ANALYSIS AND
DESIGN OF LINEAR TIME INVARIANT SYSTEMS(U) NAVAL
POSTGRADUATE SCHOOL MONTEREY CA H ISMAIL DEC 84

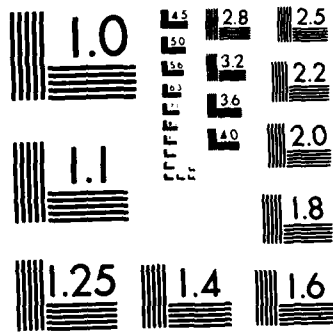
1/1

UNCLASSIFIED

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

G

NAVAL POSTGRADUATE SCHOOL Monterey, California



AD-A154 076

DTIC
ELECTE
MAY 23 1985
S
A

THESIS

INTERACTIVE COMPUTER PROGRAM
FOR THE
ANALYSIS AND DESIGN OF LINEAR
TIME INVARIANT SYSTEMS

by

Habib Ismail
December 1984

Thesis Advisor:

G. J. Thaler

DTIC FILE COPY

Approved for public release; distribution is unlimited.

85

047

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Interactive Computer Program for the Analysis and Design of Linear Time Invariant Systems		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1984
7. AUTHOR(s) Habib Ismail		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE December 1984
		13. NUMBER OF PAGES 62
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution is unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Interactive Program, Bode Plot, Linear Control Systems		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) In this thesis, an interactive computer program for the analysis and design of time invariant unity feedback control systems is presented, using cascade or feedback or both types of compensation. By using this program, the user is freed from the tedious, time consuming and error prone method of hand calculations, letting the computer handle these tasks		

efficiently and speedily. The user can then concentrate fully on the placement of poles and zeroes of the compensator(s) used.

Design of control systems by classical methods being essentially a repetitive, trial and error procedure, this program greatly cuts down the turn around time and leads to faster, more satisfactory results.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
A1	

Approved for public release; distribution is unlimited.

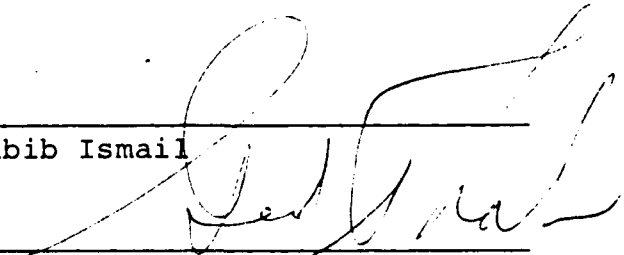
Interactive Computer Program
for the
Analysis and Design of Linear Time Invariant Systems

by

Habib Ismail
Lieutenant Commander, Pakistan Navy
B.E., University of Karachi, Pakistan, 1974

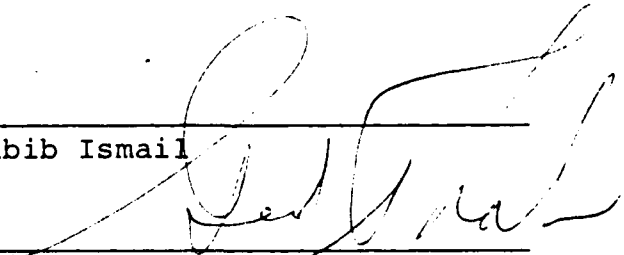
Submitted in partial fulfillment of the
requirements for the degree of
MASTER OF SCIENCE IN ELECTRICAL ENGINEERING
from the
NAVAL POSTGRADUATE SCHOOL
December 1984

Author:

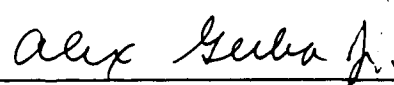


Habib Ismail

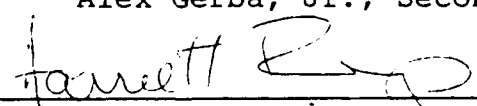
Approved by:



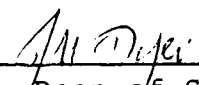
George J. Thaler, Thesis Advisor



Alex Gerba, Jr., Second Reader



Harriett B. Rigas, Chairman, Department of
Electrical and Computer Engineering



John Dyer, Dean of Science and Engineering

ABSTRACT

In this thesis, an interactive computer program for the analysis and design of time invariant unity feedback, ^{linear} control systems is presented, using cascade or feedback or both types of compensation.

By using this program, the user is freed from the tedious, time consuming and error prone method of hand calculations, letting the computer handle these tasks efficiently and speedily. The user can then concentrate fully on the placement of poles and zeroes of the compensator(s) used.

Design of control systems by classical methods being essentially a repetitive, trial and error procedure, this program greatly cuts down the turn around time and leads to faster, more satisfactory results.

*Additional keywords:
Transfer function, state space*

TABLE OF CONTENTS

I.	INTRODUCTION	9
II.	CONTROL ENGINEERING ANALYSIS	11
	A. GENERAL	11
	B. PROBLEM FORMULATION	12
	C. COMPUTER AIDED DESIGN	14
III.	PROGRAMMING CONSIDERATIONS	16
	A. GENERAL	16
	B. MAIN FEATURES OF THE PROGRAM	16
	C. PROGRAM DESCRIPTION	18
IV.	PROGRAM PERFORMANCE INVESTIGATION	20
	A. GENERAL	20
	B. EXAMPLE PROBLEMS	20
	1. Example 4.1 : A Phase Lead Network	20
	2. Example 4.2 : A Lag/Lead Network	26
	3. Example 4.3 : Velocity Feedback	31
V.	CONCLUSION AND RECOMMENDATIONS	37
	A. CONCLUSION	37
	B. RECOMMENDATIONS	38
	1. Curve Fitting	38
	2. Computer Selection of Compensators	38

3.	Root Locus	39
4.	Integrated Control System Design	39
	APPENDIX PROGRAM LISTING	40
	LIST OF REFERENCES	60
	INITIAL DISTRIBUTION LIST	61

LIST OF TABLES

1.	Tabular Output of Example 4.1	24
2.	Tabular Output of Example 4.2	29
3.	Tabular Output of Example 4.3	34

LIST OF FIGURES

1.	Types of Compensators	13
2.	Uncompensated Bode Plot of Example 4.1	22
3.	Lead Compensation Bode Plot of Example 4.1	23
4.	Uncompensated Bode Plot for Example 4.2	27
5.	Lag/Lead Compensation of Example 4.2	28
6.	Uncompensated Bode Plot of Example 4.3	32
7.	Velocity Feedback Used in Example 4.3	33
8.	Closed Loop Response of Example 4.3	36

I. INTRODUCTION

During the past two decades, the scientific and engineering communities have witnessed an ever increasing role of the digital computer in the fields of research, development and analysis of systems. The computer, today, is being used to solve engineering problems, whose solution, until very recently required long and tedious procedures. Still, it is probably true that this machine has potential not fully recognized yet, which is why the attention of so many computer scientists/engineers is focused on devising more efficient and innovative operating procedures.

Control system design is one area where classical theory has been extensively developed and used. It is fair to say that even today, most analysis and design problems of linear, time invariant control systems can still be approached using the methods developed by Bode, Nyquist and others.

A totally new approach to the design of control systems became available with the development of optimal control theory and the state variable analysis. These methods have been intensively developed in the last 10-15 years, but now their weaknesses have been exposed too. The "states" of the plant may not necessarily represent physically measurable quantities, and consequently it may not be possible to

implement the results at all. Luenberger's observers, designed to overcome this problem, can at best provide estimates of the state trajectory. Furthermore, the optimal control approach to design relies very heavily on mathematical manipulation, providing little insight to the actual working of the plant; the only input of the designer being the form of the cost function.

An intelligent use of the speed and information processing ability of the digital computer, coupled with the reliable features of classical theory appear to be the best solution to the problem at hand. The classical approach to design, being essentially a trial and error method, if the order of the system is fairly high, the number of repetitive calculations and the time required to perform these calculations becomes prohibitively large, the assistance of the computer in such problems becomes indispensable.

The work in this thesis was to develop an interactive, user oriented computer program that would prompt the user to input the transfer function and cascade/feedback compensators. The program would then display on the IBM 3277 - Tektronix 618 dial screen terminals the Bode Plot of both magnitude and phase. The program could be repeatedly used, with the user having the option to change/modify the compensators, each time viewing the effect of his modifications on the screen until he arrives at a satisfactory solution.

II. CONTROL ENGINEERING ANALYSIS

A. GENERAL

A continuous time control system may be represented in one of the following forms:

- a. Transfer functions
- b. State equations
- c. System block diagrams or signal flow graphs

Algorithms exist in almost every undergraduate control engineering text to convert the system representation from one given form to another. Gianniotis (Ref. 1) describes a simple method of converting from transfer function to state variable form in Chapter II.

The transfer function representation in its most general form is:

$$\frac{A_m S^m + A_{m-1} S^{m-1} + A_{m-2} S^{m-2} + \dots + A_0 S^0}{B_n S^n + B_{n-1} S^{n-1} + B_{n-2} S^{n-2} + \dots + B_0 S^0}$$

Usually the mathematical description of the system is found in the transfer function form in the literature. Analysis and design of control systems by classical methods also requires the representation of the system in this form.

Table 1. (Contd.)

0.464157E+01	-0.480013E+01	-0.100608E+03
0.514174E+01	-0.636283E+01	-0.103974E+03
0.569579E+01	-0.788934E+01	-0.107576E+03
0.630955E+01	-0.944906E+01	-0.111432E+03
0.693945E+01	-0.110539E+02	-0.115540E+03
0.774261E+01	-0.127110E+02	-0.119885E+03
0.857692E+01	-0.144313E+02	-0.124433E+03
0.950116E+01	-0.162196E+02	-0.129133E+03
0.105250E+02	-0.180814E+02	-0.133930E+03
0.115591E+02	-0.200195E+02	-0.138782E+03
0.129155E+02	-0.220340E+02	-0.143812E+03
0.145072E+02	-0.241230E+02	-0.149009E+03
0.158489E+02	-0.262850E+02	-0.154300E+03
0.175567E+02	-0.285134E+02	-0.159746E+03
0.194485E+02	-0.308033E+02	-0.165335E+03
0.215443E+02	-0.331480E+02	-0.171076E+03
0.238658E+02	-0.355431E+02	-0.176974E+03
0.264375E+02	-0.379807E+02	-0.183021E+03
0.292863E+02	-0.404556E+02	-0.189216E+03
0.324421E+02	-0.429623E+02	-0.195562E+03
0.359380E+02	-0.454962E+02	-0.202067E+03
0.398106E+02	-0.480529E+02	-0.208734E+03
0.441004E+02	-0.506288E+02	-0.215565E+03
0.488526E+02	-0.532208E+02	-0.222564E+03
0.541167E+02	-0.558260E+02	-0.229732E+03
0.599483E+02	-0.584423E+02	-0.237073E+03
0.664080E+02	-0.610677E+02	-0.244588E+03
0.735639E+02	-0.637005E+02	-0.252278E+03
0.814910E+02	-0.663395E+02	-0.259988E+03
0.902722E+02	-0.689835E+02	-0.267703E+03
0.999998E+02	-0.716317E+02	-0.275433E+03

Table 1. Tabular Output of Example 4.1

FREQ	MAGNITUDE	PHASE
0.100000E-01	0.539790E+02	-0.901134E+02
0.110770E-01	0.530907E+02	-0.901258E+02
0.122712E-01	0.522019E+02	-0.901395E+02
0.135936E-01	0.513130E+02	-0.901546E+02
0.150584E-01	0.504242E+02	-0.901714E+02
0.166810E-01	0.495354E+02	-0.901900E+02
0.184785E-01	0.486466E+02	-0.902106E+02
0.204697E-01	0.477579E+02	-0.902333E+02
0.226754E-01	0.468691E+02	-0.902588E+02
0.251188E-01	0.459804E+02	-0.902868E+02
0.276250E-01	0.450918E+02	-0.903179E+02
0.302840E-01	0.442032E+02	-0.903524E+02
0.341455E-01	0.433146E+02	-0.903905E+02
0.378249E-01	0.424262E+02	-0.904329E+02
0.419008E-01	0.415378E+02	-0.904798E+02
0.464138E-01	0.406490E+02	-0.905318E+02
0.514175E-01	0.397615E+02	-0.905890E+02
0.569580E-01	0.388736E+02	-0.906536E+02
0.630950E-01	0.379859E+02	-0.907247E+02
0.698946E-01	0.370985E+02	-0.908036E+02
0.774263E-01	0.362114E+02	-0.908913E+02
0.857695E-01	0.353247E+02	-0.909887E+02
0.950117E-01	0.344385E+02	-0.910972E+02
0.105250E+00	0.335529E+02	-0.912180E+02
0.116391E+00	0.326680E+02	-0.913520E+02
0.129155E+00	0.317841E+02	-0.915030E+02
0.143072E+00	0.309012E+02	-0.916711E+02
0.158489E+00	0.300196E+02	-0.918595E+02
0.175567E+00	0.291397E+02	-0.920711E+02
0.194486E+00	0.282617E+02	-0.923094E+02
0.215443E+00	0.273860E+02	-0.925787E+02
0.238658E+00	0.265130E+02	-0.928840E+02
0.264370E+00	0.256433E+02	-0.932318E+02
0.292864E+00	0.247775E+02	-0.936298E+02
0.324421E+00	0.239162E+02	-0.940870E+02
0.359381E+00	0.230600E+02	-0.946171E+02
0.398106E+00	0.222090E+02	-0.952334E+02
0.441004E+00	0.213655E+02	-0.959547E+02
0.488526E+00	0.205278E+02	-0.968035E+02
0.541168E+00	0.196960E+02	-0.978075E+02
0.599483E+00	0.188709E+02	-0.989995E+02
0.664081E+00	0.180487E+02	-0.100417E+03
0.735840E+00	0.172265E+02	-0.102103E+03
0.814911E+00	0.163984E+02	-0.104101E+03
0.902723E+00	0.155561E+02	-0.106450E+03
0.999999E+00	0.146882E+02	-0.109181E+03
0.110775E+01	0.137807E+02	-0.112300E+03
0.122712E+01	0.128181E+02	-0.115784E+03
0.135935E+01	0.117855E+02	-0.119567E+03
0.150583E+01	0.106723E+02	-0.123548E+03
0.166809E+01	0.947399E+01	-0.127800E+03
0.184785E+01	0.819452E+01	-0.131594E+03
0.204696E+01	0.684498E+01	-0.135427E+03
0.226754E+01	0.544112E+01	-0.139035E+03
0.251188E+01	0.399998E+01	-0.142404E+03
0.276255E+01	0.253084E+01	-0.145561E+03
0.302839E+01	0.106322E+01	-0.148567E+03
0.341454E+01	-0.413782E+00	-0.151495E+03
0.378247E+01	-0.189173E+01	-0.154429E+03
0.419007E+01	-0.337225E+01	-0.157444E+03

EXAMPLE 4.1
LEAD COMPENSATION

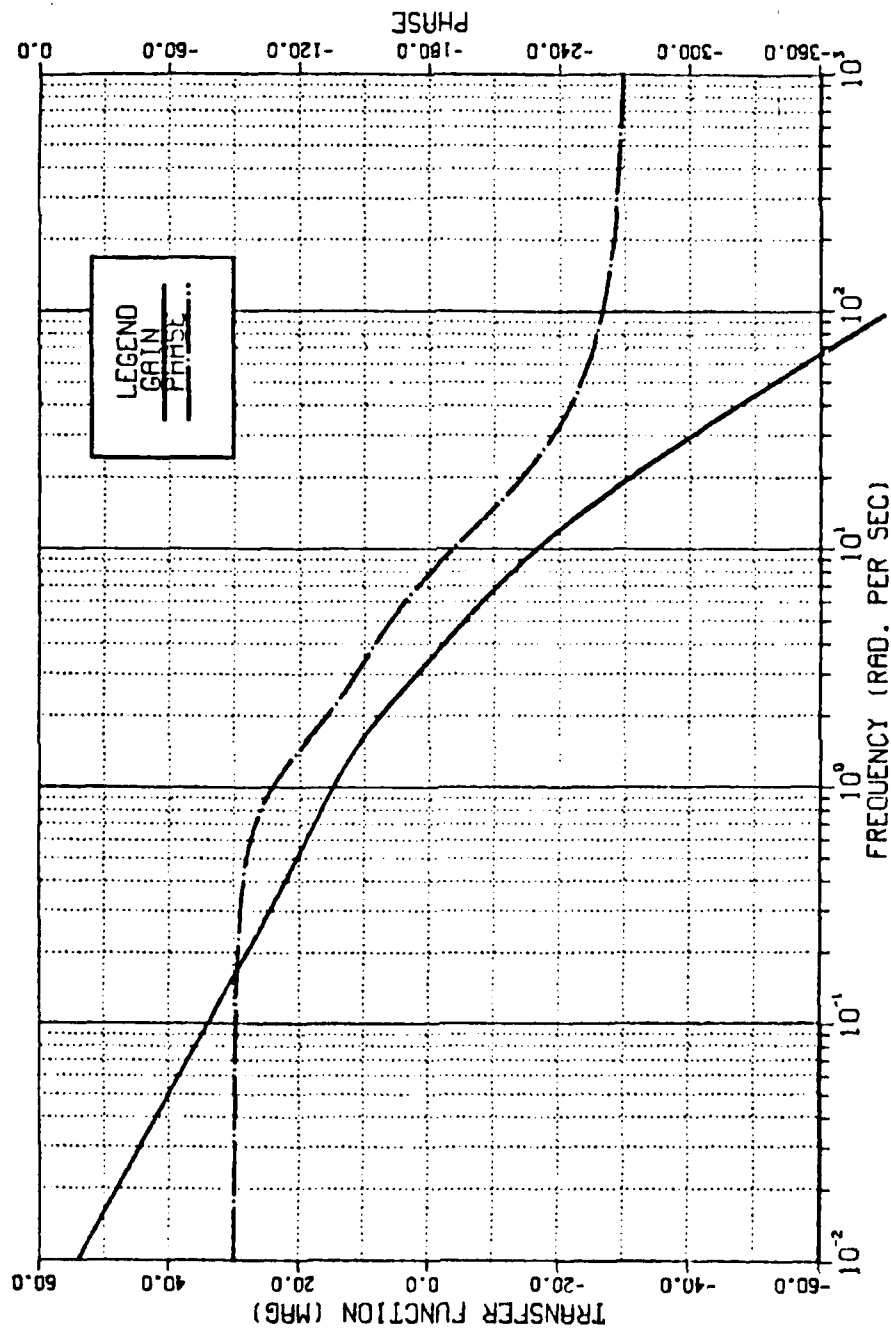


Figure 3. Lead Compensation Bode Plot of Example 4.1

EXAMPLE 4.1
UNCOMPENSATED SYSTEM

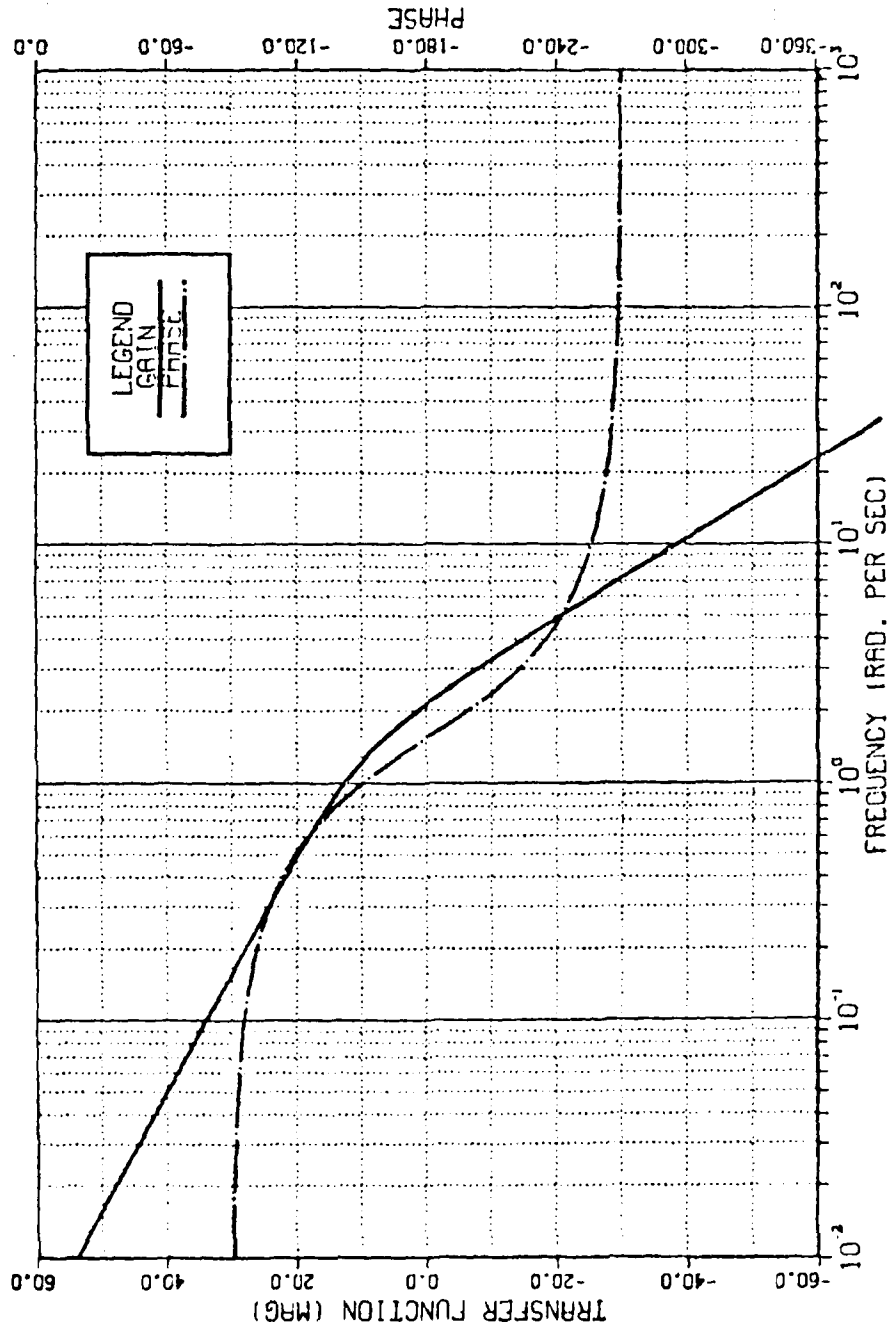


Figure 2. Uncompensated Bode Plot of Example 4.1

The velocity constant is not to be decreased. Design a phase lead compensator which will provide M_{pw} less than 1.5.

b. Solution

The first step is to get the transfer function in the required form:

$$G(s) = \frac{5.0}{0.21s^3 + 1.0s^2 + 1.0s} \quad (4.2)$$

The Bode Plot of the uncompensated system drawn using the program is shown on figure 2 and the tabular output on table 1. The uncompensated system has a phase margin of -20 degrees. To achieve a M_{pw} less than 1.5, a phase margin of 44 degrees or more is required. Approximately 64 degrees of positive phase shift are therefore needed. Two sections of lead filter are introduced as given below:

$$G_c = \frac{(s/3.0 + 1)(s/1.5 + 1)}{(s/10.0 + 1)^2}$$

The compensated Bode Plot is given on figure 3 showing a phase margin of 35 degrees.

It may be pointed out that the final values for the lead filter poles and zeroes were arrived at after 3 iterations and this design problem was solved in less than 15 minutes.

IV. PROGRAM PERFORMANCE INVESTIGATION

A. GENERAL

The program was tested by solving several linear control system design problems. Very satisfactory results were obtained in all cases with remarkable efficiency and speed. The only necessary condition is proper problem formulation. This is true for any interactive computer program. Once the program is used a couple of times, the user gets the necessary familiarity with its working.

The example problems presented below are used to demonstrate the performance and capabilities of the program. The examples can also help the user in formulating his own particular problem. The examples were selected from Thaler's 'Design of Feedback Systems' (Ref. 2)

B. EXAMPLE PROBLEMS

1. Example 4.1 : A Phase Lead Network

a. Problem Statement

A positioning system is single loop with unity feedback and forward transfer function

$$G(s) = \frac{5.0}{s(0.7s + 1)(0.3s + 1)}$$

TITLES inputs the two lines of text as headings for the Bode Plot.

In addition, the main program handles the tasks of displaying the tabular output and the Bode Plot of the open and closed loop response of the system as required.

C. PROGRAM DESCRIPTION

The main program is the coordination center which controls the calling of the supporting subroutines, in order to input the transfer function, cascade/feedback compensators and other necessary information. The tabular results and the Bode Plot are then displayed on the two screens respectively.

The main program as well as the accompanying subroutines can be found in Appendix A. They contain a sufficient number of comment cards to be self explanatory.

The following is a brief description of the performance and purpose of the various subroutines.

NUMER inputs the numerator of the plant transfer function.

DENOM inputs the denominator of the plant transfer function.

CASCAD inputs up to 6 first order lead/lag filters.

SECAS inputs the numerator of the second order band pass/band stop filter.

SECASD inputs the denominator of the second order filter.

FETCH determines the value of the radial frequency, w , at the origin of the x-axis.

DECADE determines the number of decades of frequency to be spanned.

FEEDBK inputs the various parameters of feedback compensators.

In brief, the whole BODPLT package works as follows:
A user, logged into the VM/CMS environment of the system from the dual screen terminals, issues the command DISSPLA BODPLT. The package then assumes control. The program begins its execution by interrogating the user and calling the appropriate subroutine accordingly. All programmer-composed subprograms are included in the main program titled BODPLT FORTRAN.

The BODPLT program has the following important features:

- runs of the VM/CMS time sharing system.
- interrogates the user in entering all problem specifications from the terminal.
- can handle up to a ninth order plant transfer function, six first order and one second order cascade filters.
- prompts the user to input the parameters of velocity feedback, acceleration feedback or approximate acceleration feedback as required.
- provides the solution in tabular form on the IBM 3277 screen and the Bode Plot on the TEK 618 terminal.
- can provide hard copy version of the problem specifications and tabular output by using the RECORD ON/RECORD OFF execs, and of the BODE PLOT on the Tektronix printer where installed.
- allows problem specifications to be changed between runs.

III. PROGRAMMING CONSIDERATIONS

A. GENERAL

Any interactive software package, such as the one developed in this thesis must provide a simple yet unambiguous means of data input and output. The data must be easy to interpret and apply to the problem at hand. Programs producing highly satisfactory results but requiring long studying time and/or special programming skills are of limited use only.

The intent of this thesis is to present such a program, Computer Aided Design of Linear Systems. Special care has been taken to develop the program so that the user has to invest very little time learning to use it.

B. MAIN FEATURES OF THE PROGRAM

The development of a user oriented interactive computer program in solving engineering problems requires a considerable amount of programming work, contributing significantly to its complexity and size.

The computer program, hereafter referred to as BODPLT consists of a main program, a number of programmer-composed subroutines, a few library functions/subroutines, and various subroutines of the DISSPLA graphics package. The entire program is written in the FORTRAN IV language.

It was found on the average that turn around time for a typical third order system with two cascade compensators is less than five minutes.

Cascade compensation may further be classified into two further types; a high pass filter usually called a phase lead compensator, and a low pass or phase lag compensator.

The selection of the type of compensator(s) to be used depends on a number of factors, the important ones being experience, personal preferences, availability, system constraints, etc. Unfortunately, there are no mathematical techniques to help in this selection process. Generally, one has to complete several designs and then choose the most appropriate one.

C. COMPUTER AIDED DESIGN

Of all the classical design methods available, the Bode diagram technique is generally considered to be the simplest, at the same time providing the most insight into system performance and behavior. The Bode design method may be used equally effectively both with cascade and feedback compensation schemes.

The computer program developed in this thesis displays on the terminal, initially, the Bode diagram (magnitude and phase) of the uncompensated open loop system. The user may then select the type of compensator to be used, and feed this information to the computer. The display changes, now showing the compensated system Bode Plot. This procedure may be repeated iteratively, with the computer updating the Bode Plot with each change made.

Compensators used in the design of feedback control systems are generally classified as shown in the block diagrams of figure 1.

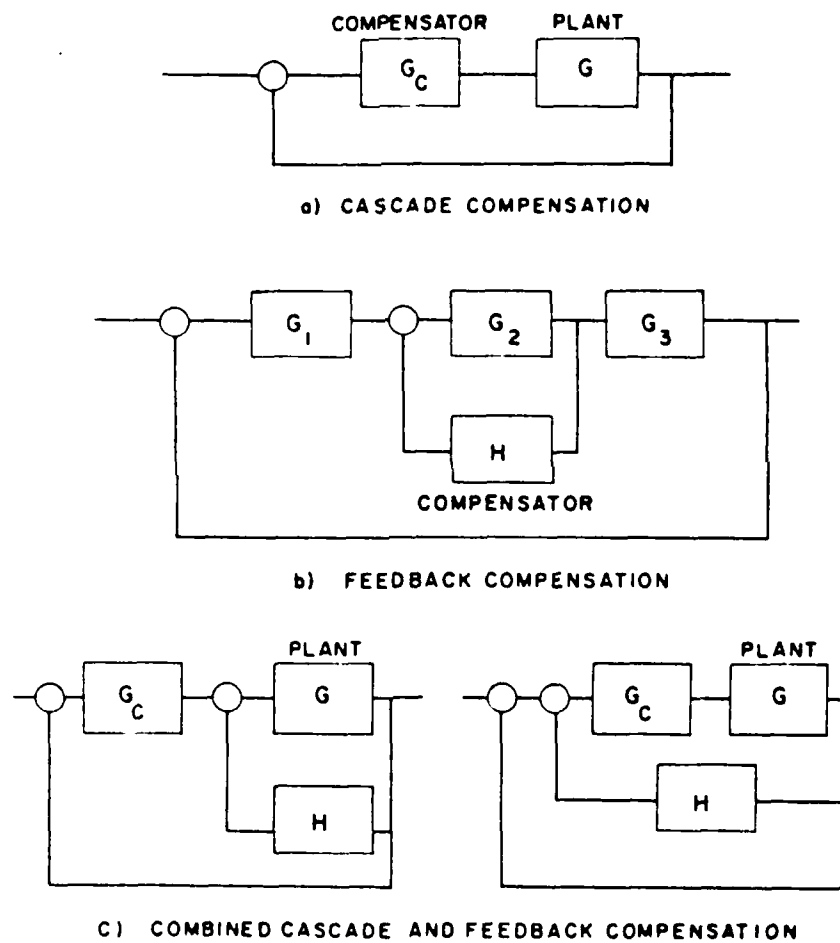


Figure 1. Classification of Compensation Structures

This thesis does not address the problem of converting from one form of representation to another. It is assumed that any conversions necessary have already been performed and that the system is represented by its open loop transfer function.

B. PROBLEM FORMULATION

Each system design has its own unique characteristics, but in general the system has to meet some kind of performance standards. These performance standards are generally provided as numerical specifications. The first step in the design of a control system is to analyze the system by itself in the usual feedback loop configuration. This is usually referred to as the uncompensated system.

Analysis of the uncompensated system almost always shows that the system cannot meet some or all of the given performance standards. Usually, additional components have to be inserted in the system for the purpose of altering the performance of the system. These components are called compensators. Compensation is a two step procedure, in which additional components (compensators) are inserted to change the structure of the system, and these components are then adjusted until the performance characteristics are satisfied.

The theory of cascade and feedback compensation is discussed in detail by Thaler (Ref. 2) in Chapters 5 and 6. Only a brief discussion of the types of compensators is presented here.

EXAMPLE 4.2
UNCOMPENSATED SYSTEM

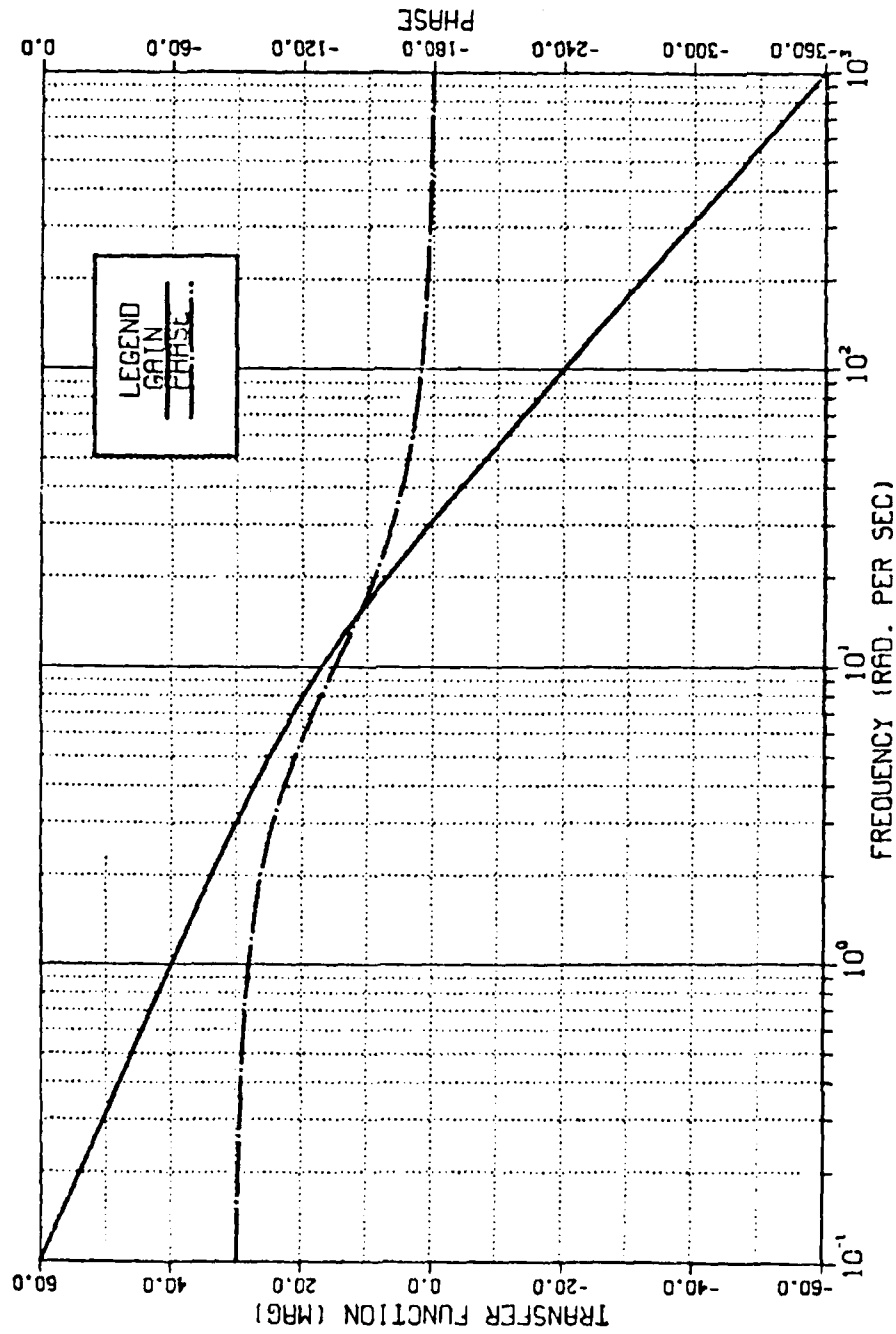


Figure 4. Uncompensated Bode Plot of Example 4.2

EXAMPLE 4.2
LAG/LEAD COMP.

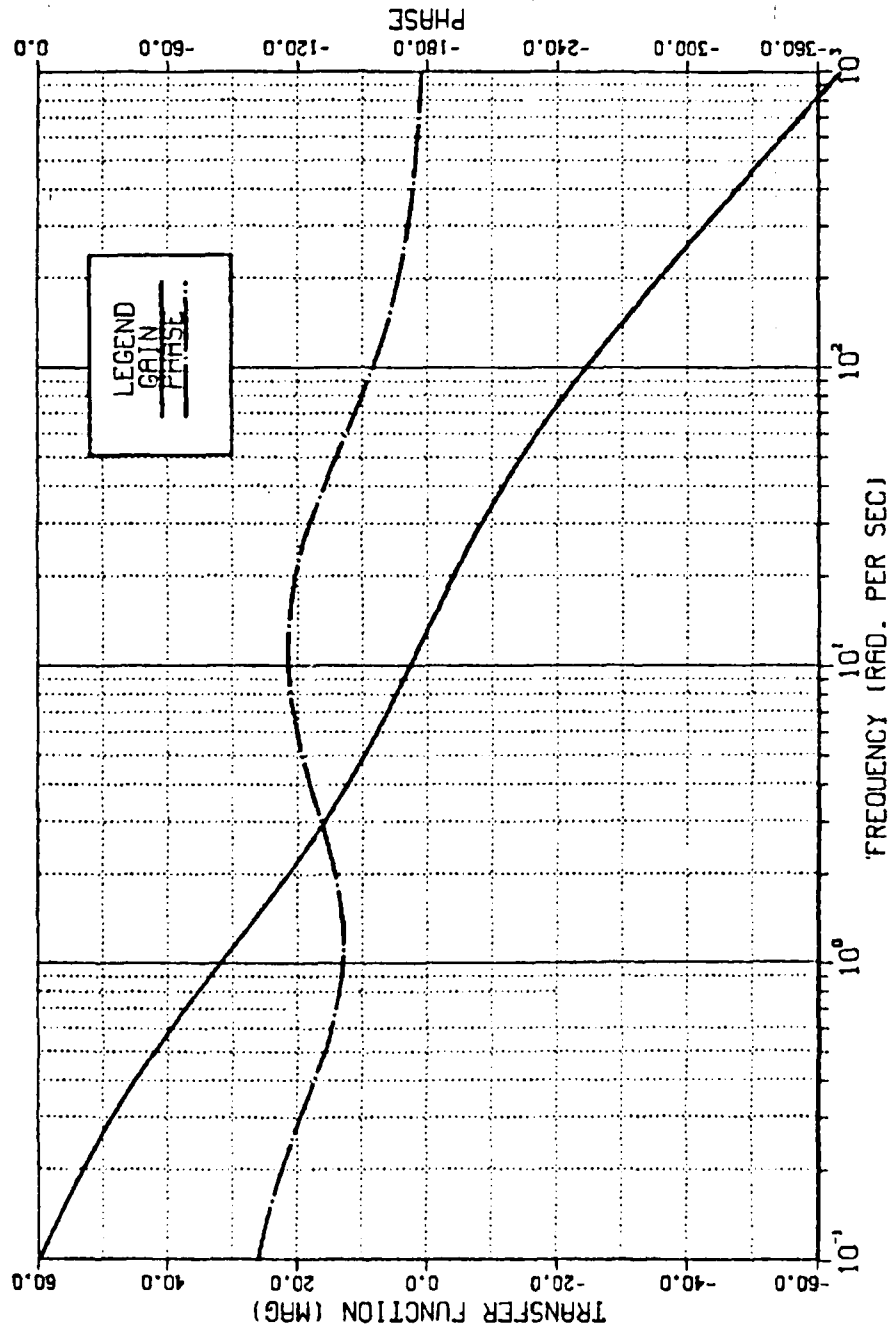


Figure 5. Lag/Lead Compensation in Example 4.2

Table 2. Tabular Output of Example 4.2

FREQ	MAGNITUDE	PHASE
0.10000E+00	0.598345E+02	-0.995141E+02
0.110776E+00	0.589089E+02	-0.100503E+03
0.122712E+00	0.579754E+02	-0.101580E+03
0.135935E+00	0.570325E+02	-0.102769E+03
0.150584E+00	0.560783E+02	-0.104058E+03
0.166810E+00	0.551106E+02	-0.105457E+03
0.184785E+00	0.541270E+02	-0.106968E+03
0.204697E+00	0.531249E+02	-0.108594E+03
0.226754E+00	0.521012E+02	-0.110331E+03
0.251188E+00	0.510529E+02	-0.112174E+03
0.278256E+00	0.499765E+02	-0.114115E+03
0.308240E+00	0.488887E+02	-0.116138E+03
0.341454E+00	0.477266E+02	-0.118226E+03
0.378248E+00	0.465474E+02	-0.120355E+03
0.419008E+00	0.453289E+02	-0.122491E+03
0.464158E+00	0.440698E+02	-0.124600E+03
0.514175E+00	0.427697E+02	-0.126683E+03
0.569580E+00	0.414292E+02	-0.128623E+03
0.630956E+00	0.400502E+02	-0.130449E+03
0.698946E+00	0.386356E+02	-0.132106E+03
0.774263E+00	0.371891E+02	-0.133500E+03
0.857695E+00	0.357155E+02	-0.134785E+03
0.950117E+00	0.342201E+02	-0.135758E+03
0.105250E+01	0.327090E+02	-0.136401E+03
0.116591E+01	0.311881E+02	-0.136884E+03
0.129155E+01	0.296640E+02	-0.137022E+03
0.143072E+01	0.281432E+02	-0.136877E+03
0.158469E+01	0.266320E+02	-0.136457E+03
0.175567E+01	0.251367E+02	-0.135775E+03
0.194485E+01	0.236631E+02	-0.134853E+03
0.215443E+01	0.222164E+02	-0.133716E+03
0.238658E+01	0.208014E+02	-0.132395E+03
0.264375E+01	0.194217E+02	-0.130927E+03
0.292864E+01	0.180801E+02	-0.129351E+03
0.324421E+01	0.167783E+02	-0.127709E+03
0.359381E+01	0.155166E+02	-0.126043E+03
0.398106E+01	0.142947E+02	-0.124393E+03
0.441004E+01	0.131107E+02	-0.122797E+03
0.488526E+01	0.119624E+02	-0.121289E+03
0.541168E+01	0.108466E+02	-0.119899E+03
0.599483E+01	0.975994E+01	-0.118652E+03
0.664081E+01	0.869860E+01	-0.117569E+03
0.735640E+01	0.765877E+01	-0.116666E+03
0.814911E+01	0.663658E+01	-0.115955E+03
0.902723E+01	0.562823E+01	-0.115445E+03
0.999999E+01	0.463006E+01	-0.115145E+03
0.110775E+02	0.363853E+01	-0.115060E+03
0.122712E+02	0.265011E+01	-0.115192E+03
0.135935E+02	0.166130E+01	-0.115547E+03
0.150583E+02	0.663658E+00	-0.116125E+03
0.166809E+02	-0.331396E+00	-0.116927E+03
0.184784E+02	-0.134252E+01	-0.117953E+03
0.204696E+02	-0.236847E+01	-0.119201E+03
0.226754E+02	-0.341315E+01	-0.120600E+03
0.251188E+02	-0.448058E+01	-0.122345E+03
0.278255E+02	-0.557482E+01	-0.124219E+03
0.308239E+02	-0.669992E+01	-0.126281E+03
0.341454E+02	-0.785960E+01	-0.128510E+03
0.378247E+02	-0.905738E+01	-0.130883E+03
0.419007E+02	-0.102961E+02	-0.133373E+03

Table 2. (Contd.)

0.404157E+C2	-0.115780E+02	-0.135950E+03
0.514174E+C2	-0.129043E+02	-0.138581E+03
0.569579E+02	-0.142753E+02	-0.141232E+03
0.630955E+C2	-0.156905E+02	-0.143871E+03
0.698945E+02	-0.171484E+02	-0.146467E+03
0.774261E+02	-0.186466E+02	-0.148993E+03
0.857692E+02	-0.201822E+02	-0.151427E+03
0.950116E+02	-0.217522E+02	-0.153749E+03
0.105250E+C3	-0.233526E+02	-0.155948E+03
0.116591E+03	-0.249806E+02	-0.158014E+03
0.129155E+03	-0.266322E+02	-0.159944E+03
0.143072E+C3	-0.283043E+02	-0.161736E+03
0.158489E+C3	-0.299941E+02	-0.163392E+03
0.175567E+C3	-0.316988E+02	-0.164917E+03
0.194485E+C3	-0.334161E+02	-0.166316E+03
0.215443E+C3	-0.351439E+02	-0.167597E+03
0.238658E+C3	-0.366805E+02	-0.168765E+03
0.264375E+C3	-0.386245E+02	-0.169830E+03
0.292863E+03	-0.403745E+02	-0.170798E+03
0.324421E+C3	-0.421296E+02	-0.171678E+03
0.359380E+C3	-0.438888E+02	-0.172476E+03
0.398105E+C3	-0.456513E+02	-0.173199E+03
0.441004E+C3	-0.474166E+02	-0.173854E+03
0.488525E+03	-0.491842E+02	-0.174447E+03
0.541167E+03	-0.509536E+02	-0.174984E+03
0.599482E+C3	-0.527247E+02	-0.175469E+03
0.664080E+03	-0.544969E+02	-0.175907E+03
0.735639E+C3	-0.562702E+02	-0.176304E+03
0.814910E+03	-0.580443E+02	-0.176662E+03
0.902721E+03	-0.598190E+02	-0.176986E+03
0.999958E+C3	-0.615944E+02	-0.177276E+03

3. Example 4.3 : Velocity Feedback

a. Problem Statement

A simple second-order servo is to be compensated with tachometer feedback. The forward transfer function is

$$G(S) = \frac{100.0}{S(S + 1)}$$

and the tachometer transfer function is $K_t S$. The tachometer is fed back around all of the forward gain. Using Bode diagram methods, set K_t to provide $M_{pw} = 1.3$.

b. Solution

Bode Plot for this system is shown on figure 6. The system has a phase margin of about 6 degrees. For $M_{pw} = 1.3$, a phase margin of 45 degrees is required. A rough graphical design on the uncompensated Bode Plot gives

$$1/H = 12.0/S \quad \text{or} \quad H = 0.08S$$

The Bode Plot for the compensated system is on figure 7, showing a phase margin of 50 degrees.

The close loop frequency response of this example, drawn using BODPLT is shown on figure 8.

EXAMPLE 4.3
UNCOMPENSATED SYSTEM

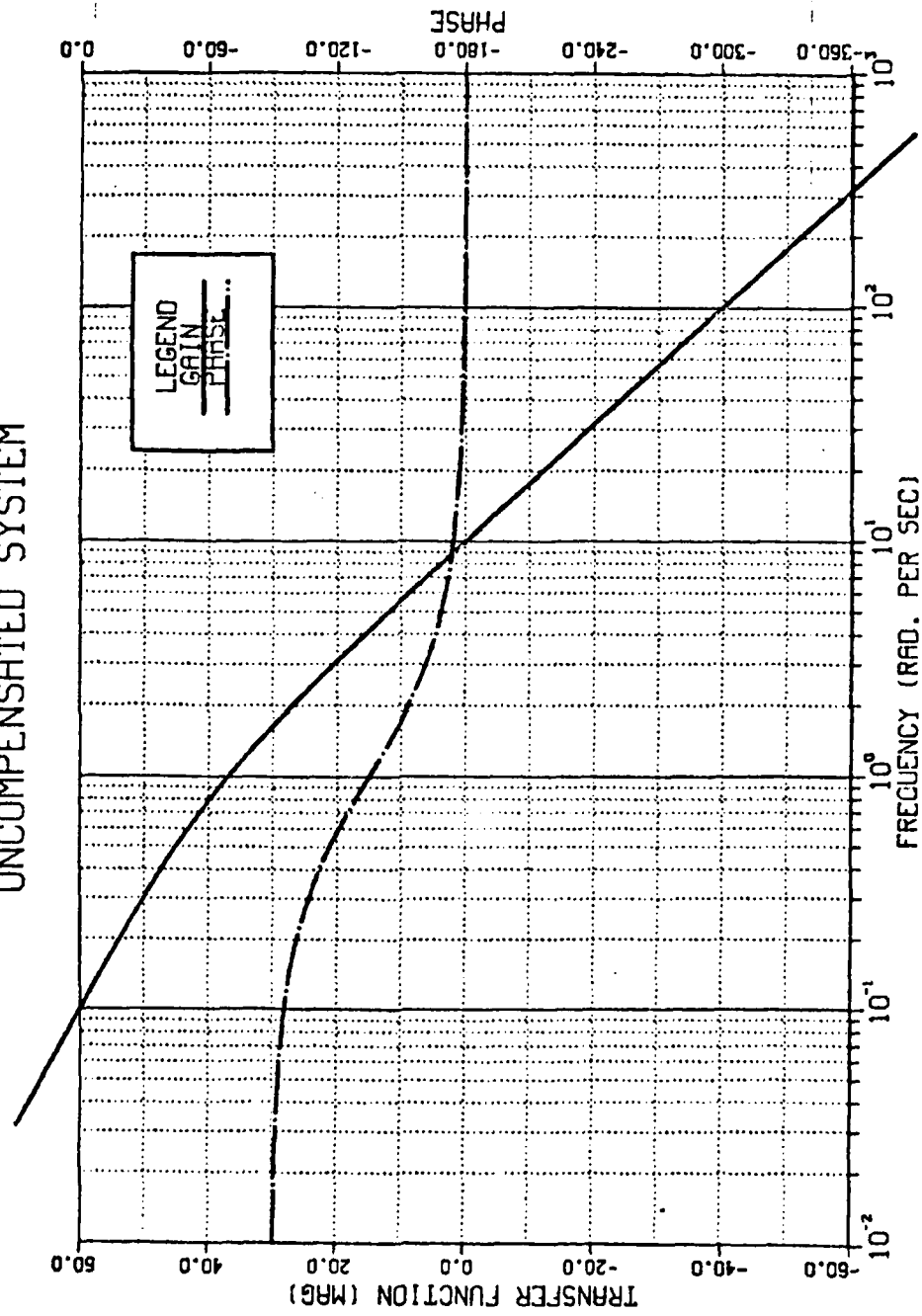


Figure 6. Uncompensated Bode Plot of Example 4.3

EXAMPLE 4.3

VELOCITY FEEDBACK

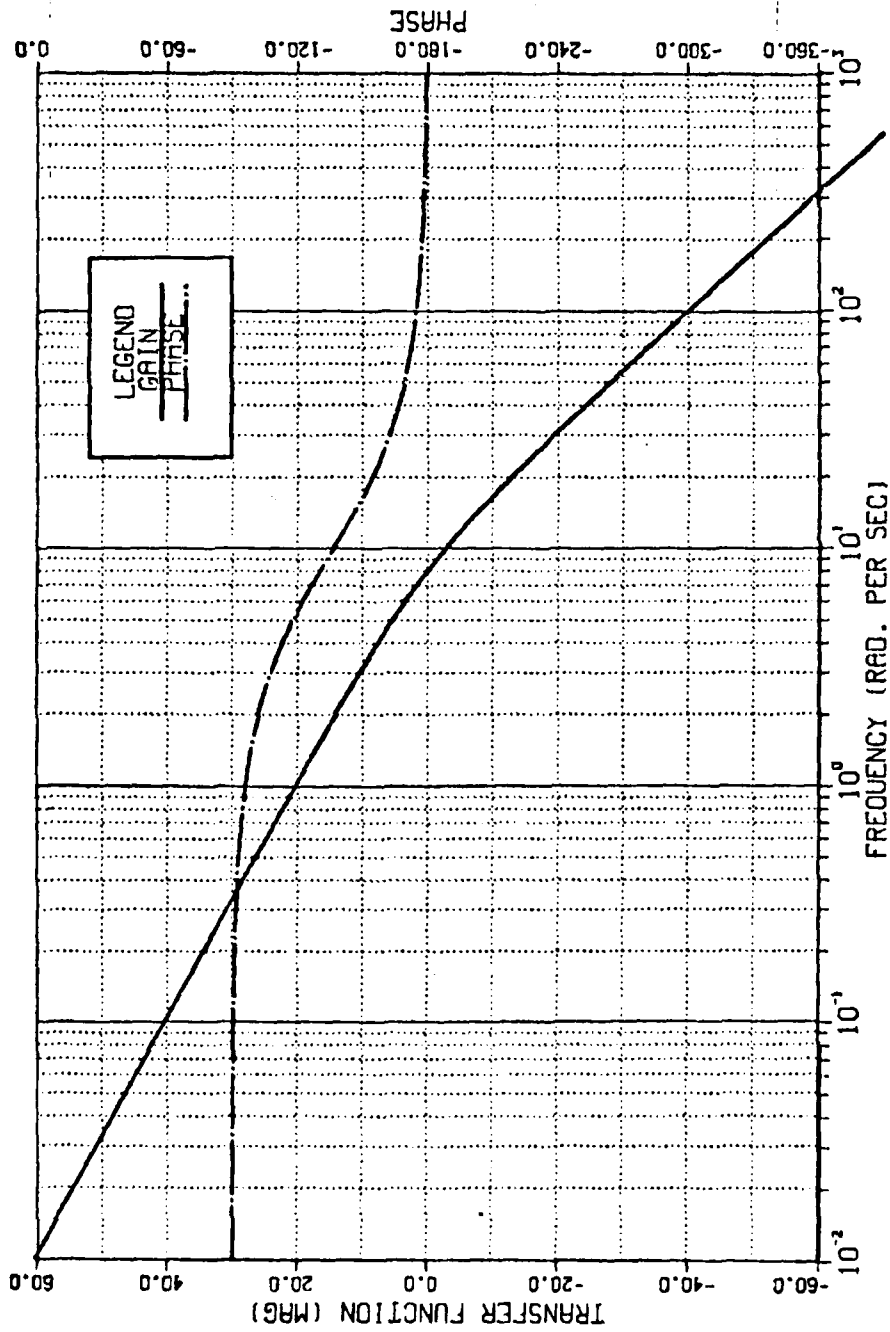


Figure 7. Velocity Feedback Used in Example 4.3

Table 3. Tabular Output of Example 4.3

FREQ	MAGNITUDE	PHASE
0.100000E-C1	0.606303E+02	-C.900604E+02
0.113646E-C1	0.595192E+02	-C.900608E+02
0.129155E-C1	0.584081E+02	-C.900784E+02
0.146780E-C1	0.572970E+02	-U.900692E+02
0.166810E-C1	0.561859E+02	-C.901016E+02
0.189573E-C1	0.550747E+02	-0.901156E+02
0.215443E-C1	0.539636E+02	-C.901310E+02
0.244843E-C1	0.528525E+02	-C.901497E+02
0.278256E-C1	0.517414E+02	-C.901703E+02
0.316227E-C1	0.506303E+02	-C.901936E+02
0.359381E-C1	0.495192E+02	-C.902202E+02
0.408423E-C1	0.484080E+02	-0.902504E+02
0.464159E-C1	0.472969E+02	-C.902848E+02
0.527499E-C1	0.461858E+02	-0.903238E+02
0.599483E-C1	0.450746E+02	-C.903682E+02
0.681291E-C1	0.439635E+02	-C.904185E+02
0.774263E-C1	0.428523E+02	-C.904758E+02
0.879921E-C1	0.417411E+02	-C.905409E+02
0.999998E-C1	0.406298E+02	-C.906149E+02
0.113646E+00	0.395186E+02	-0.906989E+02
0.129155E+00	0.384073E+02	-C.907945E+02
0.146780E+00	0.372959E+02	-0.909030E+02
0.166810E+00	0.361845E+02	-C.910204E+02
0.189573E+00	0.350730E+02	-C.911666E+02
0.215443E+00	0.339613E+02	-C.913259E+02
0.244843E+00	0.328495E+02	-C.915069E+02
0.278255E+00	0.317376E+02	-0.917120E+02
0.316227E+00	0.306253E+02	-0.919463E+02
0.359381E+00	0.295127E+02	-C.922118E+02
0.408423E+00	0.283997E+02	-0.925134E+02
0.464157E+00	0.272862E+02	-C.928500E+02
0.527498E+00	0.261720E+02	-0.932451E+02
0.599483E+00	0.250568E+02	-C.936870E+02
0.681291E+00	0.239404E+02	-C.941886E+02
0.774262E+00	0.228226E+02	-0.947579E+02
0.879919E+00	0.217028E+02	-C.954037E+02
0.999996E+00	0.205804E+02	-C.961360E+02
0.113646E+01	0.194549E+02	-0.969657E+02
0.129155E+01	0.183252E+02	-C.979052E+02
0.146779E+01	0.171902E+02	-0.989075E+02
0.166809E+01	0.160484E+02	-C.100167E+03
0.189573E+01	0.148980E+02	-C.101520E+03
0.215443E+01	0.137367E+02	-0.103042E+03
0.244843E+01	0.125615E+02	-C.104748E+03
0.278255E+01	0.113691E+02	-C.106650E+03
0.316227E+01	0.101552E+02	-C.108778E+03
0.359381E+01	0.891480E+01	-C.111127E+03
0.408423E+01	0.764222E+01	-0.113708E+03
0.464157E+01	0.633108E+01	-0.116522E+03
0.527498E+01	0.497453E+01	-C.119560E+03
0.599482E+01	0.358567E+01	-0.122804E+03
0.681290E+01	0.209793E+01	-C.126224E+03
0.774261E+01	0.565706E+00	-C.129777E+03
0.879919E+01	-0.103506E+01	-0.133413E+03
0.999996E+01	-C.270649E+01	-C.137075E+03
0.113646E+02	-0.444846E+01	-0.140704E+03
0.129155E+02	-C.625853E+01	-0.144242E+03
0.146779E+02	-0.813240E+01	-C.147639E+03
0.166809E+02	-U.100643E+02	-0.150857E+03
0.189573E+02	-C.120476E+02	-C.153860E+03

Table 3. (Contd.)

0.215443E+02	-0.140753E+02	-C.156650E+03
0.244843E+C2	-0.161408E+02	-0.159199E+03
0.278255E+C2	-0.182376E+02	-C.161517E+03
0.316226E+02	-0.203602E+C2	-C.163609E+03
0.359380E+C2	-0.225037E+02	-C.165489E+03
0.408423E+02	-0.246639E+02	-C.167170E+03
0.464157E+C2	-0.268375E+02	-0.168668E+03
0.527457E+C2	-C.290217E+02	-0.169999E+03
0.599482E+02	-0.312143E+02	-C.171179E+03
0.681290E+C2	-0.334134E+02	-0.172224E+03
0.774261E+02	-C.356177E+02	-C.173148E+03
0.879919E+02	-0.378259E+C2	-C.173964E+03
0.999555E+C2	-C.400373E+02	-C.174684E+03
0.113646E+03	-C.422511E+02	-C.175319E+03
0.129154E+C3	-0.444668E+02	-0.175679E+03
0.146779E+C3	-0.466840E+02	-0.176372E+03
0.166809E+03	-0.489023E+02	-C.176806E+03
0.189573E+C3	-0.511215E+02	-0.177189E+03
0.215443E+C3	-0.533413E+02	-C.177526E+03
0.244843E+03	-0.555617E+02	-C.177822E+03
0.278255E+C3	-0.577825E+02	-C.178083E+03
0.316226E+C3	-0.600037E+02	-C.178313E+03
0.359379E+C3	-0.622250E+02	-0.178515E+03
0.408422E+C3	-0.644466E+02	-0.178693E+03
0.464157E+03	-0.666683E+02	-C.178850E+03
0.527457E+C3	-0.688902E+02	-0.178988E+03
0.599481E+C3	-0.711121E+02	-C.179109E+03
0.681290E+03	-0.733341E+02	-C.179215E+03
0.774260E+C3	-0.755561E+02	-C.179309E+03
0.879918E+C3	-C.777782E+02	-C.179392E+03
0.999957E+C3	-0.800003E+02	-0.179465E+03

EXAMPLE 4.3
CLOSED LOOP RESPONSE

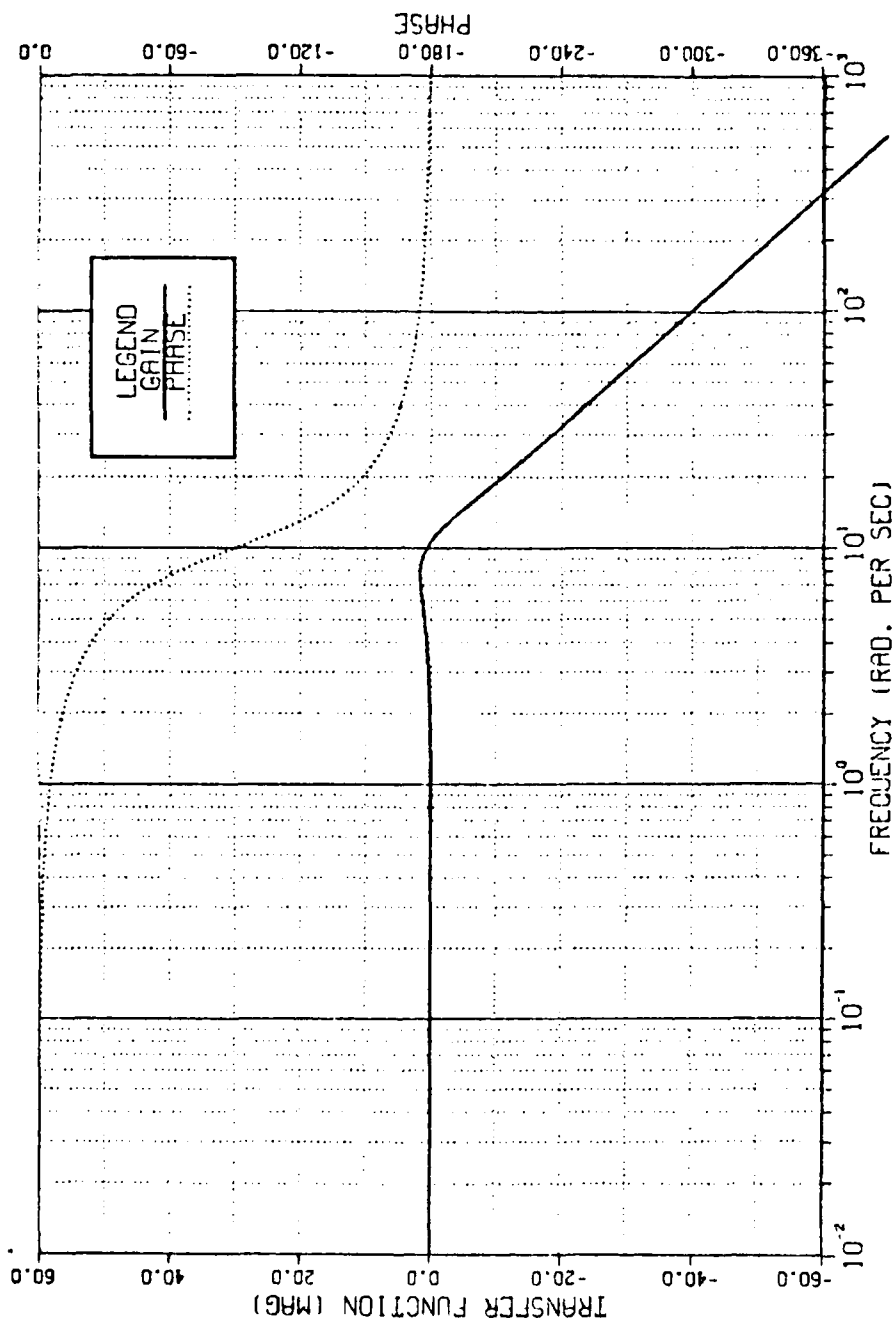


Figure 8. Closed Loop Response of Example 4.3

V. CONCLUSION AND RECOMMENDATIONS

A. CONCLUSION

The objective of this thesis was to develop an interactive user oriented computer program which would aid in solving control engineering problems using the Bode method of design. The presented program proves that frequency domain design of control systems using the digital computer as an aid is not only feasible but highly desirable. The results obtained are readily interpretable and provide good and meaningful insight into the problem.

The results obtained during the investigation of the program performance show that a complicated but well formulated problem can be solved with ease, and the solution is obtained with speed, accuracy and precision.

The entire program is less than 1000 lines, with a total of 9 subroutines. Every effort has been made to keep the program simple yet unambiguous, so that the user has to invest very little time learning how to use it. Effort has also been made to minimize the use of the computer CPU time. However, expenditure of CPU time is to a large measure dependent upon:

- (1) the order of the system.
- (2) the number of iterations used in reaching a solution.

(3) the type and order of the compensator(s) used.

B. RECOMMENDATIONS

The program as presented in this thesis seems to be able to adequately satisfy the usual needs in a control system design problem. A number of useful extensions to the work developed in this thesis can be carried out. These are briefly discussed.

1. Curve Fitting

Although not specifically worked on in this thesis, the program can be used quite effectively for curve fitting purposes. This was demonstrated in the initial stages of the development of this program. The procedure is, by its very nature, iterative and therefore time consuming and cumbersome. However, the entire algorithm can be automated using a minimization subroutine, with the program outputting a polynomial to fit a given curve over a specified range of the independent variable.

2. Computer Selection of Compensators

A suitable minimization routine such as Box PLX can be incorporated into the program which could select the best possible location of poles and zeroes to meet given performance specifications. This would automate the entire Bode design procedure, the user then having the option of only specifying the type of compensation, i.e., cascade or feedback. It may however be pointed out that minimization

routines by the very nature of their operation are very time consuming and wasteful of CPU resources.

3. Root Locus

Most of the subroutines developed in this thesis are very general in nature and can be adapted quite easily to develop a similar interactive program for Root Locus plots.

4. Integrated Control System Design

No meaningful design of control systems is complete without finally analyzing its time domain performance. It is therefore considered highly desirable to incorporate into this program, an interactive Root Locus design procedure and then a time domain analysis of the compensated system. The entire package would then be an excellent teaching aid for control system design.

APPENDIX
PROGRAM LISTING

NUM00010
NUM00020
NUM00030
NUM00040
NUM00050
NUM00060
NUM00070
NUM00080
NUM00090
NUM00100
NUM00110
NUM00120
NUM00130
NUM00140
NUM00150
NUM00160
NUM00170
NUM00180
NUM00190
NUM00200
NUM00210
NUM00220
NUM00230
NUM00240
NUM00250
NUM00260
NUM00270
NUM00280
NUM00290
NUM00300
NUM00310
NUM00320
NUM00330
NUM00340
NUM00350
NUM00360
NUM00370
NUM00380
NUM00390
NUM00400
NUM00410
NUM00420
NUM00430
NUM00440
NUM00450
NUM00460
NUM00470
NUM00480

```

SUBROUTINE NUMER (NA)
** *****
** SUBROUTINE TO INPUT THE NUMERATOR *****
** OF THE PLANT TRANSFER FUNCTION *****
** *****
** VARIABLE DECLARATIONS ***
INTEGER ANSWER,YES
REAL NA
DIMENSION NA(10)
DATA YES/.Y./
DO 301 I=1,10
NA(I) = 0.0
CONTINUE
CALL FRTCMS('CLRSCRN')
WRITE(6,351)
READ(5,375)N
WRITE(6,352)N
READ(5,377)ANSWER
IF (ANSWER.EQ. YES)GO TO 302
GO TO 301
CONTINUE
N=N+1
CONTINUE
CALL FRTCMS('CLRSCRN')
WRITE(6,360)
DO 306 I=1,N
J=I-1
WRITE(6,353) J
READ(5,378,END = 305) NA(I)
GO TO 306
REWINO 5
CALL FRTCMS ('CLRSCRN')
WRITE(5,356)
GO TO 304
CONTINUE
CALL FRTCMS('CLRSCRN')
DO 307 I = 1,N
J = N-I
K = J+1
WRITE(6,354) J,NA(K)
CONTINUE
WRITE(6,355)
READ(5,375)ANSWER
IF (ANSWER.EQ.YES) GO TO 308.

```



```

DEC00010
DEC00020
DEC00030
DEC00040
DEC00050
DEC00060
DEC00070
DEC00080
DEC00090
DEC00100
DEC00110
DEC00120
DEC00130
DEC00140
DEC00150
DEC00160
DEC00170
DEC00180
DEC00190
DEC00200
DEC00210
DEC00220
DEC00230
DEC00240
DEC00250
DEC00260
DEC00270
DEC00280
DEC00290
DEC00300
DEC00310
DEC00320
DEC00330
DEC00340
DEC00350
DEC00360
DEC00370
DEC00380
DEC00390
DEC00400
DEC00410
DEC00420
DEC00430
DEC00440
DEC00450
DEC00460

```

```

SUBROUTINE DECADE(DEC)
** ***** THE NUMBER OF ***** **
* SUBROUTINE TO INPUT THE NUMBER OF *
* DECADES OF FREQUENCY TO BE SPANNED. *
** ***** **
** VARIABLE DECLARATIONS ***
INTEGER DEC, ANSWER, YES
DATA YES / 'Y' /
CONTINUE (6, 151)
WRITE (6, 176) END = 102) DEC
CALL FRICMS (0, CLRS CRN)
GO TO 163
REWINJ 5
CALL FRICMS (0, CLRS CRN)
WRITE (6, 152)
GO TO 161
CONTINUE
WRITE (6, 153) DEC
READ (5, 177) ANSWER
IF (ANSWER.EQ. YES) GO TO 104
GO TO 161
CONTINUE
RETURN
C ***** WRITE FORMAT STATEMENTS ***** **
151 * FORMAT (' //, 7X, ENTER AS A SINGLE DIGIT INTEGER, THE NUMBE',
* 'R OF DECADES OF //, 7X, FREQUENCY TO BE SPANNED.',
152 * ' //, 7X, MAXIMUM //, 7X, MINIMUM : 1 //)
* ' //, 7X, YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY ',
153 * 'NUMBER. //, 7X, TRY AGAIN.')
* ' //, 7X, DECADES OF FREQUENCY WILL BE SPANNED',
* ' 5X, CORRECT? (Y/N).)
C ***** READ FORMAT STATEMENTS ***** **
176 * FORMAT(I1)
177 * FORMAT(A1)
C ***** **
C END

```

FET00010
 FET00020
 FET00030
 FET00040
 FET00050
 FET00060
 FET00070
 FET00080
 FET00090
 FET00100
 FET00110
 FET00120
 FET00130
 FET00140
 FET00150
 FET00160
 FET00170
 FET00180
 FET00190
 FET00200
 FET00210
 FET00220
 FET00230
 FET00240
 FET00250
 FET00260
 FET00270
 FET00280
 FET00290
 FET00300
 FET00310
 FET00320
 FET00330
 FET00340
 FET00350
 FET00360
 FET00370
 FET00380
 FET00390
 FET00400
 FET00410
 FET00420
 FET00430
 FET00440

```

SUBROUTINE FETCH ( FENUM)
** *****
* SUBROUTINE TO INPUT THE LOWER LIMIT *****
* CF THE FREQUENCY AXIS. *
** *****
** VARIABLE DECLARATIONS ***
INTEGER FENLM, YES, ANSWER
DATA YES/'Y'/

CONTINUE
WRITE (6, 51)
READ (5, 76) END = 02) FENUM
CALL FRICMS (0 CLRSCRN)
GO TO 03
REMAIN 5
CALL FRICMS (0 CLRSCRN)
WRITE (6, 52)
GO TO 01
CONTINUE
WRITE (6, 53) FENUM
READ (5, 77) ANSWER
IF (ANSWER.EQ.'YES') GO TO 04
CONTINUE
GO TO 01
RETURN
C ***** WRITE FORMAT STATEMENTS *****
C *****
C *****
51 FORMAT (0, //, 4X, ' ENTER THE LOWER LIMIT OF THE FREQUENCY AXIS',
* AS FOLLOWS: //, 26X, ' FOR EXAMPLE //, 8X,
* IF LOWER LIMIT IS 10**(-02), ENTER -02, //, 8X,
* IF LOWER LIMIT IS 10**(01), ENTER +11, //, 8X,
52 FORMAT (//, 7X, ' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY ',
* NUMBER: //, 7X, ' TRY AGAIN.')
53 FORMAT (//, 4X, ' THE LOWER LIMIT IS 10**', I3, ' CORRECT? (Y/N)')
C ***** READ FORMAT STATEMENTS *****
C *****
76 FORMAT (I2)
77 FORMAT (A1)
ENC

```


SEC00490
SEC00500
SEC00510
SEC00520
SEC00530
SEC00540
SEC00550
SEC00560
SEC00570
SEC00580

FCRMAI(//,7X,0 ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N).)
FORMAT(//,7X,0 YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY ,
* NUMBER.,/,7X,0 TRY AGAIN.0)

***** STATEMENTS *****

***** READ FORMAT STATEMENTS *****

FCRMAI(F14.0)
FCRMAI(A1)

END

455
456
C
C
478
479
C


```

353          FCRMAT(//,7X, ' ENTER COEFFICIENT OF S**(1,12,1,1)')
354          FCRMAT(//, ' COEFFICIENT OF S**12 : E14.6)
355          FCRMAT(//,7X, ' ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')
356          FCRMAT(//,7X, ' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
*           ' NUMBER.//,7X, ' TRY AGAIN.//)
C ***** READ FCRMAT STATEMENTS *****
C *****
378          FCRMAT(F14.6)
379          FCRMAT(A1)
C          ENC

```

```

SEC00490
SEC00500
SEC00510
SEC00520
SEC00530
SEC00540
SEC00550
SEC00560
SEC00570
SEC00580
SEC00590
SEC00600

```

SSEC00070
 SSEC00071
 SSEC00072
 SSEC00073
 SSEC00074
 SSEC00075
 SSEC00076
 SSEC00077
 SSEC00078
 SSEC00079
 SSEC00080
 SSEC00081
 SSEC00082
 SSEC00083
 SSEC00084
 SSEC00085
 SSEC00086
 SSEC00087
 SSEC00088
 SSEC00089
 SSEC00090
 SSEC00091
 SSEC00092
 SSEC00093
 SSEC00094
 SSEC00095
 SSEC00096
 SSEC00097
 SSEC00098
 SSEC00099
 SSEC00100
 SSEC00101
 SSEC00102
 SSEC00103
 SSEC00104
 SSEC00105
 SSEC00106
 SSEC00107
 SSEC00108
 SSEC00109
 SSEC00110
 SSEC00111
 SSEC00112
 SSEC00113
 SSEC00114
 SSEC00115
 SSEC00116
 SSEC00117
 SSEC00118
 SSEC00119
 SSEC00120
 SSEC00121
 SSEC00122
 SSEC00123
 SSEC00124
 SSEC00125
 SSEC00126
 SSEC00127
 SSEC00128
 SSEC00129
 SSEC00130
 SSEC00131
 SSEC00132
 SSEC00133
 SSEC00134
 SSEC00135
 SSEC00136
 SSEC00137
 SSEC00138
 SSEC00139
 SSEC00140
 SSEC00141
 SSEC00142
 SSEC00143
 SSEC00144
 SSEC00145
 SSEC00146
 SSEC00147
 SSEC00148

```

SUBROUTINE SECAS(CN)
** ***** INPUT THE NUMERATOR *****
* SUBROUTINE TO INPUT THE NUMERATOR *
* OF THE SECOND ORDER COMPENSATOR *
** ***** ORDER COMPENSATOR *****

** VARIABLE DECLARATIONS ***
INTEGER ANSWER, YES
REAL CN
DIMENSION CN(3)
DATA YES / .Y. /

CALL FRICMS ('CLRSCRN')
CONTINUE
WRITE(6,351)
DO 306 I=1,3
J=I-1
CONTINUE
WRITE(6,353) J
READ(5,378) END = 305) CN(I)
GO TO 306
REWIND 5
CALL FRICMS ('CLRSCRN')
WRITE(6,350)
GO TO 304
CONTINUE
CALL FRICMS ('CLRSCRN')
DO 307 I = 1,3
J = 3-I
K = J+1
WRITE(6,354) J,CN(K)
CONTINUE
WRITE(6,355)
READ(5,379) ANSWER
IF (ANSWER.EQ.YES) GO TO 308
GO TO 301
CONTINUE
CALL FRICMS ('CLRSCRN')
RETURN
C ***** WRITE FORMAT STATEMENTS *****
C
351 * *
* FORMAT(/,7X,'COEFFICIENTS OF NUMERATOR OF SECOND',
* ORDER CASCADE COMPENSATOR',/,'X','ENTER ALL',
* COEFFICIENTS IN DECIMAL FORMAT')

```

```

515 READ (5,578,END 515) CPS(I)
GO TO 516
REWIN 5
CALL FRTCMS ('CLRSCRN')
WRITE(5,556)
GO TO 514
CONTINUE
CALL FRTCMS ('CLRSCRN')
DU 517 I = 1,N
WRITE(6,564) I,CPS(I)
CONTINUE
WRITE(6,555)
READ(5,577) ANSWER
IF (ANSWER.EQ.'YES) GO TO 518
CONTINUE
CALL FRTCMS ('CLRSCRN')
RETURN
C ***** WRITE FORMAT STATEMENTS *****
551 FORMAT (//,7X, 'ENTER NUMBER OF CASCADE FILTERS AS A 1',
, 'DIGIT INTEGER',)
552 * FORMAT (//,7X,11, ' CASCADE FILTERS ARE BEING USED',4X,
, 'CORRECT? (Y/N)',)
553 FORMAT (//,7X, 'ENTER ZERO OF FILTER NUMBER',I2,
, 'E14.6',)
554 * FORMAT (//,7X, 'ZERO NUMBER',I2, ' VALUES CORRECT? (Y/N)',)
555 * FORMAT (//,7X, ' ARE THE ABOVE ENTERED "ENTER" WITHOUT ENTERING ANY',
, 'NUMBER',//,7X, ' TRY AGAIN.',)
556 * FORMAT (//,7X, ' VALUES OF POLES AND ZEROS OF THE CASCADE',
, ' FILTERS MUST BE ENTERED',//,7X, ' IN DECIMAL FORMAT',)
560 * FORMAT (//,7X, ' ENTER PCLE OF FILTER NUMBER',I2,
, ' POLE NUMBER',I2, 'E14.0',)
563 *
564 *
C ***** READ FORMAT STATEMENTS *****
C
570 FORMAT (I1)
577 *
578 *
C
END

```

```

CAS00490
CAS00500
CAS00510
CAS00520
CAS00530
CAS00540
CAS00550
CAS00560
CAS00570
CAS00580
CAS00590
CAS00600
CAS00610
CAS00620
CAS00630
CAS00640
CAS00650
CAS00660
CAS00670
CAS00680
CAS00690
CAS00700
CAS00710
CAS00720
CAS00730
CAS00740
CAS00750
CAS00760
CAS00770
CAS00780
CAS00790
CAS00800
CAS00810
CAS00820
CAS00830
CAS00840
CAS00850
CAS00860
CAS00870
CAS00880
CAS00890
CAS00900
CAS00910

```

CAS00010
CAS00020
CAS00030
CAS00040
CAS00050
CAS00060
CAS00070
CAS00080
CAS00090
CAS00100
CAS00110
CAS00120
CAS00130
CAS00140
CAS00150
CAS00160
CAS00170
CAS00180
CAS00190
CAS00200
CAS00210
CAS00220
CAS00230
CAS00240
CAS00250
CAS00260
CAS00270
CAS00280
CAS00290
CAS00300
CAS00310
CAS00320
CAS00330
CAS00340
CAS00350
CAS00360
CAS00370
CAS00380
CAS00390
CAS00400
CAS00410
CAS00420
CAS00430
CAS00440
CAS00450
CAS00460
CAS00470
CAS00480

```

SUBROUTINE CASCAD(CZS,CPS)
** *****
** SUBROUTINE TO INPUT THE ZEROS AND
** POLES OF THE CASCADE COMPENSATORS
** *****
** VARIABLE DECLARATIONS ***
INTEGER ANSWER,YES
REAL CZS,CPS
DIMENSION CZS(6),CPS(6)
DATA YES/,'Y,/'
DO 520 I = 1,6
CZS(I)=1.0
CPS(I)=1.0
CONTINUE
CALL FRTCMS('CLRSCRN')
WRITE(6,551)
READ(5,576)N
WRITE(6,552)N
READ(5,577)ANSWER
IF(ANSWER.EQ.'YES)GO TO 505
GO TO 501
CONTINUE

520 CALL FRTCMS('CLRSCRN')
WRITE(6,560)
DO 506 I=1,N
WRITE(6,553) I
READ(5,578)END = 505) CZS(I)
GO TO 506
REWINDE
CALL FRTCMS('CLRSCRN')
WRITE(5,556)
GO TO 504
CONTINUE
CALL FRTCMS('CLRSCRN')
DO 507 I = 1,N
WRITE(6,554) I,CZS(I)
CONTINUE
WRITE(6,555)
READ(5,577)ANSWER
IF (ANSWER.EQ.'YES) GO TO 508
GO TO 503
CONTINUE
DO 510 I=1,N
WRITE(6,563) I

```

DENUC490
 DENUC500
 DENUC510
 DENUC520
 DENUC530
 DENUC540
 DENUC550
 DENUC560
 DENUC570
 DENUC580
 DENUC590
 DENUC600
 DENUC610
 DENUC620
 DENUC630
 DENUC640
 DENUC650
 DENUC660
 DENUC670
 DENUC680
 DENUC690
 DENUC700
 DENUC710
 DENUC720
 DENUC730
 DENUC740
 DENUC750

```

408      GO TO 403
          CONTINUE
          CALL FRICMS('CLRSCRN')
          RETURN
C***** WRITE FORMAT STATEMENTS *****
C 451      FURMAT(//,7X,'ENTER ORDER OF DENOMINATOR POLYNOMIAL AS A 1',
C          *   DIGIT NUMBER',
C          *   ORDER OF DENOMINATOR POLYNOMIAL IS ',I1,4X,
C          *   CORRECT? (Y/N)',)
C 452      FURMAT(//,7X,'ENTER COEFFICIENT OF S*(',I2,')',)
C          *   COEFFICIENT OF S*,I2,
C          *   ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)',)
C 453      FURMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
C          *   NUMBER.',/7X,' TRY AGAIN.',)
C 454      FURMAT(//,7X,' ALL DENOMINATOR COEFFICIENTS MUST BE ENTERED ',
C          *   IN DECIMAL FORMAT.',)
C***** READ FORMAT STATEMENTS *****
C 476      FURMAT(I1)
C 477      FURMAT(A1)
C 478      FURMAT(F14.4)
C 479      FURMAT(A1)
          END

```



```

308      GO TO 303
          CONTINUE
          CALL FRTCMS('CLRSCRN')
          RETURN
C***** WRITE FORMAT STATEMENTS *****
351      * FORMAT(//,7X,'ENTER ORDER OF NUMERATOR POLYNOMIAL AS A 1',
          *   DIGIT INTEGER')
352      * FORMAT(//,7X,'ORDER OF NUMERATOR POLYNOMIAL IS ',I1,4X,
          *   'CORRECT? (Y/N)')
353      * FORMAT(//,7X,'ENTER COEFFICIENT OF S*(',I2,')')
354      * FORMAT(//,7X,'COEFFICIENT OF S*(',I2,')',E14.6)
355      * FORMAT(//,7X,'ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')
356      * FORMAT(//,7X,'YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
          *   'NUMBER.',7X,'TRY AGAIN.')
360      * FORMAT(//,7X,'ALL NUMERATOR COEFFICIENTS MUST BE ENTERED ',
          *   'IN DECIMAL FORMAT')
C***** READ FORMAT STATEMENTS *****
376      FORMAT(I1)
377      FORMAT(A1)
378      FORMAT(F14.4)
379      FORMAT(A1)
          END
C*****
NUM00490
NUM00500
NUM00510
NUM00520
NUM00530
NUM00540
NUM00550
NUM00560
NUM00570
NUM00580
NUM00590
NUM00600
NUM00610
NUM00620
NUM00630
NUM00640
NUM00650
NUM00660
NUM00670
NUM00680
NUM00690
NUM00700
NUM00710
NUM00720
NUM00730
NUM00740
NUM00750

```


PAI00C490
 PAI00C500
 PAI00C510
 PAI00C520
 PAI00C530
 PAI00C540
 PAI00C550
 PAI00C560
 PAI00C570
 PAI00C580
 PAI00C590
 PAI00C600
 PAI00C610
 PAI00C620
 PAI00C630
 PAI00C640
 PAI00C650
 PAI00C660
 PAI00C670
 PAI00C680
 PAI00C690
 PAI00C700
 PAI00C710
 PAI00C720
 PAI00C730
 PAI00C740
 PAI00C750
 PAI00C760
 PAI00C770
 PAI00C780
 PAI00C790
 PAI00C800
 PAI00C810
 PAI00C820
 PAI00C830
 PAI00C840
 PAI00C850
 PAI00C860
 PAI00C870
 PAI00C880
 PAI00C890
 PAI00C900
 PAI00C910
 PAI00C920
 PAI00C930
 PAI00C940
 PAI00C950
 PAI00C960

```

1002 CALL SECASC(CN1)
      CALL SECASC(CD1)
      CONTINUE
      WRITE(6,1007)ANSWER
      READ(5,1008)ANSWER,EG,YES,GGU,TG,1019
      IF(.CN1)ANSWER,EG,YES,GGU,TG,1019
      CALL FEELEK(FK,FA,CA,FF)
      CONTINUE
1015 CALL FETCH(NUM)
      CALL LECADE(DECS)
      CALL TITLE(TTL,MSS)
1014 CONTINUE = FLGAT(DECS)
      ADECS = FC76C
      WRITE(6,1017)ANS2
      READ(5,1018)ANS2
      DO 10 I = 1,271
      LUGA(I) = (1-I)/270.*ADECS
      FREC(I) = (10.0**((LCGA(I))))*(10.**((NUM)))
      S = CMFLX(2,FREQ(I))
C***** UNCOMPENSATED SYSTEM : NUMERATOR *****
C*****
      N = A(1)*(S**0) + A(2)*(S**1) + A(3)*(S**2) + A(4)*(S**3)
      * + A(5)*(S**4) + A(6)*(S**5) + A(7)*(S**6) + A(8)*(S**7)
      * + A(9)*(S**8) + A(10)*(S**9)
C***** UNCOMPENSATED SYSTEM : DENOMINATOR *****
C*****
      D = B(1)*(S**0) + B(2)*(S**1) + B(3)*(S**2) + B(4)*(S**3)
      * + B(4)*(S**3) + B(5)*(S**4) + B(6)*(S**5) + B(7)*(S**6)
      * + B(8)*(S**7) + B(9)*(S**8) + B(10)*(S**9)
      HPLANT = N/D
C***** FIRST ORDER CASCADE FILTERS *****
C*****
      NCAS = ((S/CZ(1))+1.)*((S/CZ(2))+1.)*((S/CZ(3))+1.)*
      * ((S/CZ(4))+1.)*((S/CZ(5))+1.)*((S/CZ(6))+1.)
C*****
      DCAS = ((S/CP(1))+1.)*((S/CP(2))+1.)*((S/CP(3))+1.)*
      * ((S/CP(4))+1.)*((S/CP(5))+1.)*((S/CP(6))+1.)
C*****
      HCAS = NCAS/DCAS
C***** SECOND ORDER CASCADE FILTERS *****
C*****
      NCAS2 = (CN1(3))*(S**2)+CN1(2)*(S**1)+CN1(1)
      * (CN2(3))*(S**2)+CN2(2)*(S**1)+CN2(1)
      * (CN3(3))*(S**2)+CN3(2)*(S**1)+CN3(1)
      * (CN4(3))*(S**2)+CN4(2)*(S**1)+CN4(1)
      * (CN5(3))*(S**2)+CN5(2)*(S**1)+CN5(1)
  
```

MAI00970
 MAI00980
 MAI00990
 MAI01000
 MAI01010
 MAI01020
 MAI01030
 MAI01040
 MAI01050
 MAI01060
 MAI01070
 MAI01080
 MAI01090
 MAI01100
 MAI01110
 MAI01120
 MAI01130
 MAI01140
 MAI01150
 MAI01160
 MAI01170
 MAI01180
 MAI01190
 MAI01200
 MAI01210
 MAI01220
 MAI01230
 MAI01240
 MAI01250
 MAI01260
 MAI01270
 MAI01280
 MAI01290
 MAI01300
 MAI01310
 MAI01320
 MAI01330
 MAI01340
 MAI01350
 MAI01360
 MAI01370
 MAI01380
 MAI01390
 MAI01400
 MAI01410
 MAI01420
 MAI01430
 MAI01440

SECFIL=NCAS2/DCAS2

***** FEEDBACK COMPENSATION *****

NFBK = FK*(S**FN)
 DFbk = (S**DN)+(FP)
 HFbk = NFBK/DFBK
 H = HPLANT*HCAS*SECFIL
 H = H/(1.0 + (HFbk**H))
 IF (ANS2.EQ. UPEN) GO TO 1128
 H = H/(1.0 + H)
 CONTINUE
 RE = REAL(H)
 AI = AIMAG(H)
 PHD = 57.295 * ATAN2(AI,RE)
 IF (PHD.GT.0.) PHD = PHD - 360.
 PHASE(1) = PHD
 HMAG(1) = 20.*ALOG10(CABS(H))
 IF (HMAG(1).GT.120.)HMAG(1) = 120.
 IF (HMAG(1).LT.-120.)HMAG(1) = -120.
 CONTINUE
 WRITE(6,1067)
 READ(5,1076)ANSWER
 IF (.NOT. ANSWER.EQ. YES) GO TO 1999
 CALL FRICMS(:CLRSRN,)
 WRITE(6,1005)
 DO 1999 I=1,271,3
 WRITE(6,1060)FREQ(I),HMAG(I),PHASE(I)
 CONTINUE
 CALL TEKOLD
 CALL BLOWUP(1.2)
 CALL PAGE(1,8-5)
 CALL NCCHEK
 CALL AREA2D(9.,6.)
 CALL XNAME(:FREQUENCY (RAD. PER SEC),\$,100)
 CALL YNAME(:TRANSFER FUNCTION (MAG),\$,100)
 CALL HEADIN(TTL,20,1.5,2)
 CALL HEADIN(MSS,20,1.5,2)
 CALL XLUG(FREQ(1),9./ADELS,-60.00,20.)
 CALL LINES(:PHASE\$,IPAK,2)
 CALL LINES(:GAIN\$,IPAK,1)
 XM=XLUGND(IPAK,2)
 YW=YLEUGND(IPAK,2)
 XL=8. - .5-XW-.1
 YL=6. - .5-YW-.1
 CALL PLNK1(XL-.3,0.-.4,YL-.3,6.-.4,2)
 CALL LINESP(3.5)
 CALL DOT

C
 C

1128

C
 C
 10

1999
 C

MAI01450
 MAI01460
 MAI01470
 MAI01480
 MAI01490
 MAI01500
 MAI01510
 MAI01520
 MAI01530
 MAI01540
 MAI01550
 MAI01560
 MAI01570
 MAI01580
 MAI01590
 MAI01600
 MAI01610
 MAI01620
 MAI01630
 MAI01640
 MAI01650
 MAI01660
 MAI01670
 MAI01680
 MAI01690
 MAI01700
 MAI01710
 MAI01720
 MAI01730
 MAI01740
 MAI01750
 MAI01760
 MAI01770
 MAI01780
 MAI01790
 MAI01800
 MAI01810
 MAI01820
 MAI01830
 MAI01840
 MAI01850
 MAI01860
 MAI01870
 MAI01880
 MAI01890
 MAI01900
 MAI01910
 MAI01920

CALL GRID(1,2)
 CALL RESET(,03T,)
 CALL THKCRV(,0315)
 CALL LEGLIN
 CALL CURVE(FREQ,HMA6,271,0)
 CALL YGRAXS(-360,160,0,0,5,0,PHASE\$,,-100,9,0,0,0,)
 CALL DOT
 CALL LEGLIN
 CALL CURVE(FREQ,PHASE,271,0)
 CALL RESET(,BLNK1,)
 CALL RESET(,THKCRV,)
 CALL LEGEND(IPAK,2,XL,YL)
 CALL ENDPL(0)
 WRITE(6,1051)
 READ(5,1076) ANSWER
 IF(,NOT,ANSWER,=EQ, YES) GO TO 1006

 CONTINUE
 WRITE(5,1053)
 READ(5,1076) ANS
 CALL FRICMS(,CLKSCRN,)
 IF(ANS,=EQ,CHP)GO TO 1014
 IF(ANS,=EQ,CHW)GO TO 1010
 IF(ANS,=EQ,CHL)GO TO 1011
 IF(ANS,=EQ,CHC)GO TO 1012
 IF(ANS,=EQ,CHY)GO TO 1017
 IF(ANS,=EQ,CHN)GO TO 1124
 IF(ANS,=EQ,CHS)GO TO 1125
 IF(ANS,=EQ,CHF)GO TO 1126
 CALL DECADE(DEC5)
 GO TO 1015
 CALL FETCH(NUM)
 GO TO 1015
 CALL CASCAD(CZ,CP)
 GO TO 1015
 CALL TTLES(TTL,MSS)
 GO TO 1015
 CALL NUMER(A)
 GO TO 1015
 CALL DENUM(B)
 GO TO 1015
 CALL SECAS(CN1)
 CALL SECAS(CD1)
 GO TO 1015
 CALL FEEDBK(FK,FN,D,FP)
 WRITE(6,1055)
 READ(5,1076) ANS1

1018

 1010
 1011
 1012
 1017
 1124
 1125
 1126
 1127
 1015

LIST OF REFERENCES

1. Gianniotis, Demetrios D., Interactive Control System Optimization and Simulation, Thesis Dissertation, Naval Postgraduate School, October 1982.
2. Thaler, G.J., Design of Feedback Systems, Dowden Hutchinson and Ross, Inc., 1973
3. Ward, J.R., and Strum, R.D., State Variable Analysis, Naval Postgraduate School, Monterey, California, 1975.
4. Luenberger, D.G., Introduction to Linear and Nonlinear Programming, Addison-Wesley Publishing Company, Inc. 1973.
5. Kuo, Benjamin C., Automatic Control Systems, Prentice-Hall, Inc., 1982.
6. Thaler, G.J., Automatic Control Systems, Naval Postgraduate School, 1984.
7. Bazarra, M.S., Nonlinear Programming, Theory and Algorithms, John Wiley and Sons, Inc., 1979
8. Box, M.J., "A New Method of Constraint Optimization and a Comparison with Other Methods," Computer Journal, pp. 45-52, 8 April 1965.
9. Schaum's Outline Series, Programming With Fortran, McGraw-Hill Book Company, 1978.
10. W.R. Church Computer Center, Technical Note, "User's Guide to VM/CMS at NPS," April 1981.
11. W.R. Church Computer Center, Technical Note, "Using Disspla at NPS," June 1984.
12. Integrated Software Systems Corporation, "Disspla Users' Manual," October 1981.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
2. Professor George J. Thaler, Code 62TR Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	5
3. Professor Alex Gerba, Jr., Code 62GZ Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	1
4. Department Chairman, Code 62RR Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	1
5. Naval Secretary Naval Headquarters Islamabad Pakistan	1
6. Directorate of Naval Training Naval Headquarters Islamabad Pakistan	1
7. Directorate of Naval Weapons and Equipment Naval Headquarters Islamabad Pakistan.	1
8. Commander Karachi 9 Liaquat Barracks Karachi Pakistan	1

- | | | |
|-----|---|---|
| 9. | Commanding Officer
Pakistan Naval Engineering College
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan | 1 |
| 10. | LCDR Habib Ismail, P.N.
Pakistan Naval Engineering College
PNS Jauhar
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan | 5 |
| 11. | LCDR M. Tariq, P.N.
Pakistan Naval Engineering College
PNS Jauhar
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan | 1 |
| 12. | Tayfun Tansan
Goztepe Istasyon Cad. Np. 31/13
Istanbul
Turkey | 1 |
| 13. | LT Kyritsis-Spromilios Pericles, H.N.
Sarantaporou 125
Papagou
Athens
Greece | 1 |
| 14. | LT Horianopoulos Emmanuel, H.N.
Travlantoni 39
Patisia
Athens
Greece | 1 |
| 15. | Defense Technical Information Center
Cameron Station
Alexandria, Virginia 22314 | 2 |

END

FILMED

7-85

DTIC