

AD-A081 464

GENERAL ELECTRIC CO UTICA N Y AIRCRAFT EQUIPMENT DIV
NONLINEAR SYSTEM IDENTIFICATION STUDY, PART I, IMPLEMENTATION F--ETC(U)
DEC 79 E J EWEN

F/G 9/2

F30602-79-C-0046

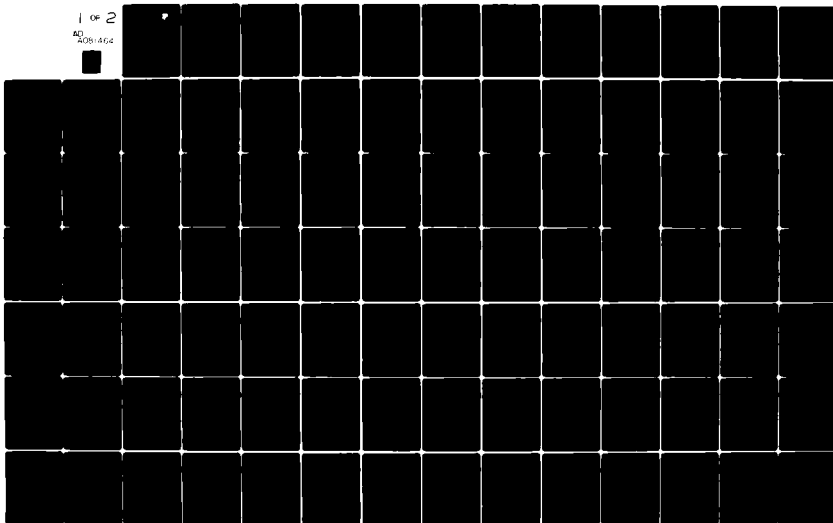
UNCLASSIFIED

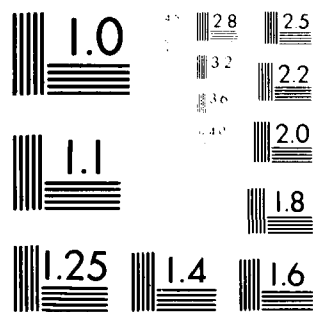
RADC -TR-79-199-PT-1

NL

1 of 2

AD
A081 464





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

RADC-TR-79-199, Part I (of two)
Final Technical Report
December 1979

12 LLVCE

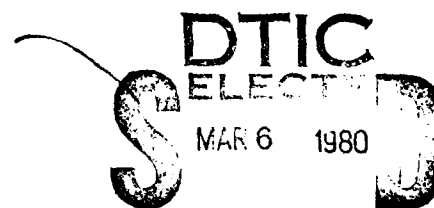


NONLINEAR SYSTEM IDENTIFICATION STUDY Implementation Feasibility Study

General Electric Company

Dr. E. J. Ewen

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED



ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

A

80 3 4 000

AD A 081 464

NONLINEAR SYSTEM IDENTIFICATION STUDY Part 1

DDC FILE COPY

7

This report has been reviewed by the RADC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-79-199 has been reviewed and is approved for publication.

APPROVED: *Daniel J. Kenneally*

DANIEL J. KENNEALLY
Project Engineer

APPROVED: *David C. Luke*

DAVID C. LUKE, Lt Colonel, USAF
Chief, Reliability and Compatibility Division

FOR THE COMMANDER:

John P. Huss
JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (RBCT) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 18 RADC TR-79-199 Part 1 (of two)	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9
4. TITLE (and Subtitle) NONLINEAR SYSTEM IDENTIFICATION STUDY, Part I Implementation Feasibility Study.	5. TYPE OF REPORT & PERIOD COVERED Final Technical Report. Jan - Sep 79	
6. AUTHOR(s) 10 Dr. E. J. Ewen	7. PERFORMING ORG. REPORT NUMBER N/A	
9. PERFORMING ORGANIZATION NAME AND ADDRESS General Electric Company Aircraft Equipment Division Utica NY 13501	8. CONTRACT OR GRANT NUMBER(s) 15 F30602-79-C-0046	
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (RBCT) Griffiss AFB NY 13441	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62702F 23380321	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same	12. REPORT DATE December 1979	
	13. NUMBER OF PAGES N/A	
	15. SECURITY CLASS. (of this report) UNCLASSIFIED	
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE N/A	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Same		
18. SUPPLEMENTARY NOTES RADC Project Engineer: Daniel J. Kenneally (RBCT)		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Nonlinear system identification A/D converter weakly nonlinear system nonlinear impulse response Volterra series pencil of functions		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The implementation feasibility of a nonlinear system identification technique is evaluated in this report. The identification technique uses a "black box" approach requiring measurements only at system input and output terminals and is applicable to weakly nonlinear systems whose behavior is adequately characterized by a finite Volterra series. Three hardware implementations of the identification technique are postulated		

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

and their respective performances are evaluated. The impact of A/D converter quantization error, non-ideal amplifiers, multipliers and integrators on performance of the identification process is assessed. Performance requirements for each component of the three implementations are derived via simulation and analysis. The feasibility of implementing the technique using commercially available state of the art components and measurement equipment in each implementation is assessed.

RADC-TR-79-199, Part II, A computational complexity study of the identification technique processing to determine the class of nonlinear systems to which the technique can be practically applied will be published at a later date.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TECHNICAL REPORT SUMMARY
NONLINEAR SYSTEM IDENTIFICATION FINAL REPORT

PART I. IMPLEMENTATION FEASIBILITY STUDY

A. STUDY OBJECTIVES

The basic objective of this study effort is to evaluate the practical feasibility of a nonlinear system identification technique. The identification procedure studied is a black box technique where only input and output terminal measurements of the nonlinear system are used. The identification technique is applicable to a broad class of weakly nonlinear systems whose response can be characterized by a finite Volterra series. The identification procedure involves processing the input and output responses of a nonlinear system to obtain a set of linearly independent equations which uniquely define the parameters of a functional form of the second-order impulse response. Theoretically, the proposed identification technique represents a significant improvement over existing identification techniques because of its black box formulation. The intent of the study to determine if this identification technique can be practically implemented and maintain an advantage over existing techniques.

The study effort is divided into two parts:

- Part I An implementation feasibility study to determine practical methods of implementing the measurement scheme - both digital and analog - and to evaluate the requirements for the components of the measurement scheme.
- Part II A computational complexity study of the identification technique processing to determine the class of nonlinear systems to which the technique can be practically applied.

This technical report summary covers the results of Part I of the study effort - the implementation feasibility study.

B. SUMMARY OF RESULTS AND CONCLUSIONS

Three basic implementations of the identification technique were evaluated and the requirements for the critical parameters of each element of the measurement scheme were evaluated. The

results of these performance evaluations and significant conclusions are summarized below for the three configurations.

1. Digital Implementation (Final Report Section III.B)

The digital implementation of the identification technique functions as follows. A signal generator excites the nonlinear system with the appropriate signal. This input signal and the resultant nonlinear system output are amplified and converted into digital form via A/D converters. The resultant samples are stored in memory for future nonreal-time identification processing on a general purpose digital computer.

The performance evaluation of the digital implementation of the identification technique showed that the critical components of the digital implementation are the A/D converter and the pre-A/D converter amplifier. The signal generator and data storage requirements are not technology limited in terms of enabling implementation of the identification technique. The important conclusions impacting parameter specification of these devices are summarized below.

a. A/D Converter

- (1) The A/D converter must have 14 to 16 bits of resolution for adequate performance on a two-pole system. This increases to 20 to 24 bits as the number of poles increases to four. Since the highest resolution commercially available A/D converter has 16 bits of resolution at this point in time, any experimental validation of this implementation should be restricted to systems with two poles or less.
- (2) The sampling rate requirements for the A/D converter are driven by the accuracy requirements of the processing technique. For a two pole system, the sampling rate should be 4 to 10 times slower than the highest break frequency of the system under test. Current 16-bit A/D converter technology implies that the system under test be limited to an upper break frequency of approximately 10 to 30 kHz.

b. Amplifier

- (1) The pre-A/D converter amplifier is necessary to adjust the output of the system under test to the full-scale input voltage level of the A/D converter.

- (2) The bandwidth requirements of the amplifier are a function of the processing approach used for identification. In general, one approach requires an amplifier with a bandwidth 1000 times the bandwidth of the system under test while the other approach requires the amplifier bandwidth to be approximately equal to the bandwidth of the system under test.

The above conclusions support the subsequent conclusion that the digital implementation of the identification technique can feasibly be constructed and used in an experimental test setup under the various constraints presented above.

2. Hybrid Implementation (Final Report Section III.C)

A hybrid implementation of the identification technique was evaluated. This implementation differs from the digital implementation in that the input and output of the system under test are integrated a number of times using analog integrators prior to sampling via A/D converters. These samples are stored for future nonreal-time processing on a digital computer.

The analyses of the performance of this implementation led to the following conclusions:

- (1) The resolution requirements for the hybrid implementation are significantly greater than for the digital implementation. The hybrid implementation requires 24 bits which is beyond the current state of the art in A/D converter technology. The conversion speed requirements are essentially the same as those required for the digital implementation.
- (2) For systems with two poles or less, the hybrid implementation offers no advantages over the digital implementation. For systems with more than two poles, the hybrid implementation offers potential performance improvement over the digital approach for an A/D converter with 24 bits. This improvement however increases the measurement implementation complexity and cost.

It was concluded that it is not feasible to consider implementation of this approach for an experimental validation at this time because of the A/D converter requirements. However, future improvements in A/D converter technology may permit implementation of this approach at that time.

3. Analog Implementation (Final Report Section III.D)

An analog implementation of the identification technique was also evaluated.

The analog implementation derives the necessary processing (inner product) quantities using analog components. The inner product device outputs are sampled and stored for further nonreal-time processing on a digital computer.

The performance evaluation has shown that the critical components in the analog implementation are the inner product device and the A/D converter; this has led to the following conclusions.

- (1) A minimum of 18 bits of A/D converter resolution is required to achieve minimum identification performance. Conversion speed is not important since only one sample per inner product device output is required.
- (2) The maximum tolerable error in the inner product output is on the order of 10^{-3} percent to achieve a minimum level of identification performance for A/D converters with 18 or more bits of resolution.
- (3) Performance improvement requires less inner product error (10^{-4} to 10^{-6} percent) and increased A/D converter resolution (20 to 24 bits). However, the performance of the analog implementation is below that demonstrated for the digital implementation.
- (4) Currently available analog multipliers have an output error on the order of 0.05 percent, which is approximately 50 times greater than the maximum tolerable error of 0.001 percent required for minimum performance of the identification technique.

The analog multiplier and the A/D converter requirements for the analog implementation imply that it is not feasible to consider this implementation for an experimental test setup in the present time frame. Significant technological developments for analog multipliers and A/D converters are necessary before this implementation can prove feasible.

TABLES OF CONTENTS

Section	Page
I INTRODUCTION	1
A. Study Objectives	1
B. Summary of Results and Conclusions	1
1. Digital Implementation	2
a. A/D Converter	2
b. Amplifier	2
c. Remaining Components	4
2. Hybrid Implementation	4
3. Analog Implementation	6
II IDENTIFICATION TECHNIQUE	9
A. Identification Technique Development	9
1. Background	9
2. Functional Form for $h_2(t_1, t_2)$	11
3. Identification Technique Description	14
a. Identification of Linear Impulse Response $h_1(t)$	14
b. Identification of Second Order Impulse Response	19
B. Identification Technique Processing	24
C. Performance Indices for the Identification Technique	25
III IMPLEMENTATION FEASIBILITY STUDY	29
A. Implementation Approaches	29
B. Digital Implementation	29
1. A/D Converter	31
a. Requirements	31
b. Simulation Model	31
c. Performance Evaluation	35
(1) Two-Pole System Analysis and Results	35
(2) Four-Pole System Analysis and Results	41

TABLES OF CONTENTS (Continued)

Section	Page
d. A/D Conversion Time Results	49
e. Second Order Impulse Response Identification	56
f. Additional A/D Converter Requirements	62
g. Summary of Currently Available A/D Converters	67
2. Amplifier Requirements	72
3. Signal Generator Requirements	88
4. Data Storage Requirements	94
5. Digital Implementation - Conclusions	95
a. A/D Converter	95
b. Amplifier	96
c. Remaining Components	97
d. Summary	97
C. Hybrid Implementation	97
1. Simulation of Hybrid Implementation	98
2. Two-Pole System Analysis and Results	100
3. Four-Pole System Analysis and Results	103
4. Amplifier, Signal Generator, and Data Storage Requirements	105
5. Conclusions - Hybrid Implementation	105
D. Analog Implementation	105
1. Digital Simulation - Analog Implementation	107
2. A/D Converter Requirements	107
a. Two-Pole System Analysis and Results	108
b. Integrator and Inner Product Device Requirements	108
(1) Integrator Error	116
(2) Inner Product Device	119
3. Amplifier, Signal Generator, and Data Storage Requirements	123
4. Conclusions - Analog Implementation	123
IV REFERENCES	124

TABLES OF CONTENTS (Continued)

APPENDICES		Page
A	COMPUTER PROGRAM LISTING FOR THE DIGITAL IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE	125
B	COMPUTER PROGRAM LISTING FOR THE HYBRID IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE	146
C	COMPUTER PROGRAM LISTING FOR THE ANALOG IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE	155

LIST OF ILLUSTRATIONS

Figure	Title	Page
1	Digital Implementation of Identification Technique	3
2	Hybrid Implementation	5
3	Analog Implementation	7
4	Identification Technique Configuration with N Integrators	16
5	Equivalent Linear System with Transfer Function $Y_2(s)$	21
6	Digital Implementation of Identification Technique	30
7	A/D Converter Simulation Model	31
8	Digital Output Voltages vs. Input Voltages	32
9	Listing of FORTRAN Subroutine for A/D Converter	34
10	Amplifier Circuit	36
11	Normalized Mean Square Error vs. A/D Converter Resolution for Two-Pole System (Integration Time = 9.6 μ s; Sampling Time = 4 ns)	40
12	Results of A/D Converter Resolution Simulation Runs for Two-Pole System	45
13	Results of Sampling Interval Rate Simulation for all A/D Converters	55
14	Second-Order System Response, Integration Time = 3.2 μ s, Sampling Interval = 1 ns, MSB Set for $y_1(t) + y_2(t)$	63
15	Second-Order System Response, Integration Time = 3.2 μ s, Sampling Period = 1 ns, MSB Set for $y_2(t)$	66
16	Commercially Available A/D Converter Characteristics	69
17	Current A/D Converter Conversion Speed Characteristics	70
18	Current A/D Converter Frequency Characteristics	71
19	Identification Technique Model to Account for Amplifier	73
20	Frequency Extent Comparison for Test System and Amplifier	74
21	Bode Diagram of Amplifier with Transfer Function $H_{A1}(s)$	75
22	Normalized Mean Squared Error as a Function of ω_1	76
23	Bode Diagram of Amplifier with Transfer Function $H_{A2}(s)$	78
24	Identification Technique Performance as a Function of Amplifier Characteristics	79
25	Linear System Equivalence Concept	81

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
26	Equivalent Linear System Representation for Amplifier Analysis	82
27	Identification Technique Performance as a Function of Amplifier Characteristics	85
28	Identification Performance as a Function of Amplifier Characteristics	86
29	RC Network	89
30	Operational Amplifier Network for Signal Generator	90
31	Short Pulse Response of Operational Amplifier Network of Figure 30	92
32	Signal Generator Configuration for $N = 2$	93
33	Hybrid Implementation	99
34	Hybrid Implementation Identification Performance as a Function of A/D Converter Resolution	102
35	Analog Implementation	106
36	Inner Product Device Model	106
37	Identification Technique Performance as a Function of Inner Product Error - Analog Implementation	115
38	Practical Integrator	116
39	Inner Product Device	119
40	Analog Multiplier Model	120

LIST OF TABLES

Table	Title	Page
1	Current-Voltage Relationships of Possible Nonlinearities	12
2	Identification Results for Perfect A/D Converter (Machine Accuracy), Digital Implementation (9.6 μ s Integration Time)	38
3	Results for Levels of A/D Converter Resolution from 8 to 24 Bits	39
4	Results for Levels of A/D Converter Resolution, Integration Time = 7.2 μ s, Sampling Interval = 3 ns	42
5	Results for Levels of A/D Converter Resolution, Integration Time = 4.8 μ s, Sampling Interval = 2 ns	43
6	Results for Levels of A/D Converter Resolution, Integration Time = 2.4 μ s, Sampling Interval = 1 ns	44
7	Four-Pole System, Perfect A/D Conversion	46
8	Simulation Results for Levels of A/D Converter Resolution, Four-Pole System, Integration Time = 4.8 μ s, Sampling Interval = 1.5 ns	48
9	Sampling Interval Results for Perfect A/D Conversion, Integration Time = 9.6 μ s	50
10	Sampling Interval Results for 16-Bit A/D Converter, Integration Time = 9.6 μ s	51
11	Sampling Interval Results for 14-Bit A/D Converter, Integration Time = 9.6 μ s	52
12	Sampling Interval Results for 12-Bit A/D Converter, Integration Time = 9.6 μ s	53
13	Sampling Interval Results for 10-Bit A/D Converter, Integration Time = 9.6 μ s	54
14	Second-Order Response Results for A/D Converters, Integration Time = 3.2 μ s, Sampling Interval = 1 ns, MSB Set for $y_1(t)$	60
15	Second-Order Response Results for A/D Converters, Integration Time = 3.2 μ s, Sampling Interval = 1 ns, MSB Set for $y_2(t)$	64
16	Commercially Available A/D Converters	72
17	Identification Technique Performance for Different Amplifier Configurations, Perfect A/D Conversion, Integration Time = 9.6 μ s, Sampling Interval = 4 ns	77
18	Identification Technique Performance for Different Amplifier Configurations, Perfect A/D Conversion, Integration Time = 9.6 μ s, Sampling Interval = 4 ns	80

LIST OF TABLES (Continued)

Table	Title	Page
19	Identification Technique Performance for Different Amplifier Configurations, Perfect A/D Conversion, Integration Time = 9.6 μ s, Sampling Interval = 4 ns	83
20	Identification Technique Performance for Different Amplifier Configurations, Perfect A/D Conversion, Integration Time = 9.6 μ s, Sampling Interval = 4 ns	84
21	Simulation Results for Hybrid Implementation for Different Levels of A/D Converter Resolution, Four-Pole System, Integration Time = 4.8 μ s, Sampling Interval = 1.5 ns	101
22	Simulation Results for Hybrid Implementation for Different Levels of A/D Converter Resolution, Four-Pole System, Integration Time = 4.8 μ s, Sampling Interval = 1.5 ns	104
23	Comparison of Hybrid and Digital Implementation Performance for Four-pole System, Integration Time = 4.8 μ s, Sampling Interval = 1.5 ns	104
24	Simulation Results for Analog Implementation for Different Levels of A/D Converter Resolution, Integration Time = 9.6 μ s, Analytical Integration	109
25	Simulation Results for Analog Implementation for Different Levels of Inner Product Error, Integration Time = 9.6 μ s, Perfect A/D Converter	110
26	Simulation Results for Analog Implementation for Different Levels of Inner Product Error, Integration Time 9.6 μ s, 24-Bit A/D Converter	111
27	Simulation Results for Analog Implementation for Different Levels of Inner Product Error, Integration Time = 9.6 μ s, 20-Bit A/D Converter	112
28	Simulation Results for Analog Implementation for Different Levels of Inner Product Error, Integration Time = 9.6 μ s, 18-Bit A/D Converter	113
29	Simulation Results for Analog Implementation for Different Levels of Inner Product Error, Integration Time = 9.6 μ s, 16-Bit A/D Converter	114
30	Practical Integrator Error	118
31	Integrated Output Peak Values - Two-Pole System	118

EVALUATION

The objective of this effort was to develop, quantify, and evaluate the practical constraints for the implementation of a time domain methodology for weakly nonlinear system identification. This time domain methodology permits the functional characterization (as opposed to numerical) of the second and third order Volterra kernels from input/output measurements (and subsequent analysis) on an otherwise, nonlinear black box with memory.

The process of system identification consists of postulating a valid analytical model for the system under consideration and performing tests on the system to completely specify or "identify" the parameters which describe the system analytical model. For example, a linear system is completely characterized by its impulse response, $h(t)$. The system identification process for this linear system analytical model consists of any procedure that completely determines $h(t)$. The present consideration in the area of nonlinear system identification is the derivation of a valid analytical model for the nonlinear system under consideration.

The identification procedure successfully studied is a black box technique where only input and output terminal measurements of the nonlinear system are used. The identification technique is applicable to a broad class of weakly nonlinear systems whose response can be characterized by a finite Volterra series. The identification procedure involves processing the input and output responses of a nonlinear system to obtain a set of linearly independent equations which uniquely define the parameters of a functional form of the second-order impulse response. Theoretically, the proposed identification technique represents a significant improvement over existing identification techniques because of its black box formulation. The intent of the study was to determine where this identification technique

can be practically implemented and maintain an advantage over existing techniques. To these ends, the practical implementation constraints have been developed, quantified and assessed for these candidate measurement configurations. The robustness of the technique to nonlinear circuits with many and/or repeated poles is the subject of Part II of this final report.

Daniel J. Kenneally

DANIEL J. KENNEALLY
Project Engineer

SECTION I

INTRODUCTION

A. STUDY OBJECTIVES

The basic objective of this study effort is to evaluate the practical feasibility of a nonlinear system identification technique. The identification procedure studied is a black box technique where only input and output terminal measurements of the nonlinear system are used. The identification technique is applicable to a broad class of weakly nonlinear systems whose response can be characterized by a finite Volterra series. The identification procedure involves processing the input and output responses of a nonlinear system to obtain a set of linearly independent equations which uniquely define the parameters of a functional form of the second-order impulse response. Theoretically, the proposed identification technique represents a significant improvement over existing identification techniques because of its black box formulation. The intent of the study to determine if this identification technique can be practically implemented and maintain an advantage over existing techniques.

The study effort is divided into two parts:

- Part I An implementation feasibility study to determine practical methods of implementing the measurement scheme - both digital and analog - and to evaluate the requirements for the components of the measurement scheme.
- Part II A computational complexity study of the identification technique processing to determine the class of nonlinear systems to which the technique can be practically applied.

This final report represents the results of Part I of the study effort - the implementation feasibility study. The computational complexity study results will be presented in Part II of this final report.

B. SUMMARY OF RESULTS AND CONCLUSIONS

Three basic implementations of the identification technique were evaluated and the requirements for the critical parameters of each element of the measurement scheme were evaluated. The

results of these performance evaluations and significant conclusions are summarized below for the three configurations.

1. Digital Implementation

The block diagram of the digital implementation of the identification technique is shown in Figure 1. A signal generator excites the nonlinear system with the appropriate sum of decaying exponential functions. This input and the resultant nonlinear system output are amplified and converted into digital form via A/D converters. The resultant samples are stored in memory for future nonreal-time processing on a general-purpose digital computer.

The performance evaluation of the digital implementation of the identification technique showed that the critical components of the digital implementation are the A/D converter and the pre-A/D converter amplifier. The important conclusions impacting parameter specification of these devices are summarized below.

a. A/D Converter

- (1) The A/D converter must have 14 to 16 bits of resolution for adequate performance with a two-pole system. This increases to 20 to 24 bits as the number of poles increases to four. Since the highest resolution commercially available A/D converter has 16 bits of resolution at this point in time, any experimental validation of this implementation should be restricted to systems with two poles or less.
- (2) The sampling rate requirements for the A/D converter are driven by the accuracy requirements of the processing technique. For a two pole system, the sampling rate should be 4 to 10 times slower than the highest break frequency of the system under test. The fastest 16-bit A/D converter currently available is limited to a sampling rate of 125 kHz. This implies that the system under test be limited to an upper break frequency of approximately 10 to 30 kHz.

b. Amplifier

- (1) The pre-A/D converter amplifier is necessary to adjust the output of the system under test to the full-scale input voltage level of the A/D converter.

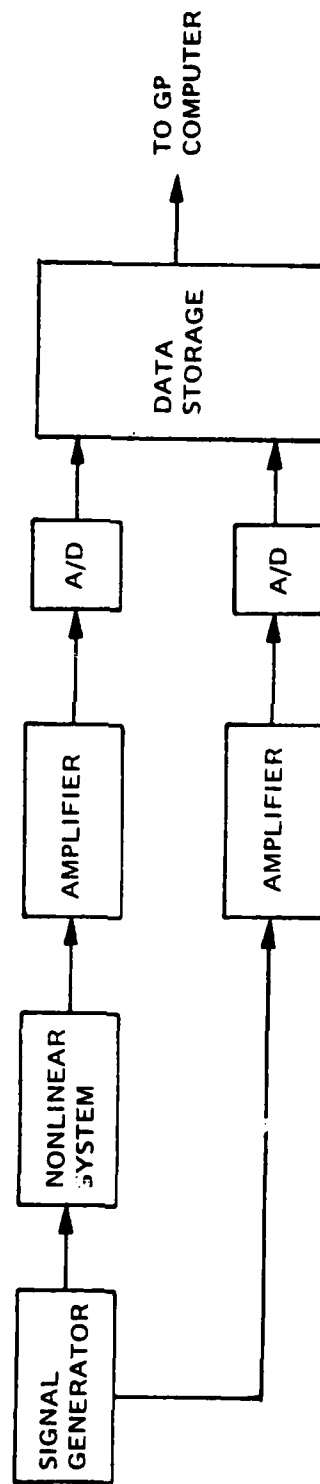


Figure 1. Digital Implementation of Identification Technique

- (2) The bandwidth requirements of the amplifier are a function of the processing approach used for identification. In general, one approach requires an amplifier with a bandwidth 1000 times the bandwidth of the system under test while the other approach requires the amplifier bandwidth to be approximately equal to the bandwidth of the system under test.
- (3) Operational amplifiers with gain-bandwidth products of up to 1000 MHz are presently commercially available. These are compatible with either processing approach described in (2) above.

c. Remaining Components

The remaining components of the digital implementation are not technology limited in terms of enabling implementation of the identification technique. The important conclusions are given below:

- (1) No commercially available waveform generator has an exponential function capability. The appropriate inputs will be generated by using operational amplifiers as low-pass filters and appropriately clamping the short pulse response to obtain the exponential function.
- (2) Data storage will be accomplished using static random access memory (RAM) chips and a programmable interface to transmit the data to the digital computer for nonreal-time processing.

The above conclusions support the subsequent conclusion that the digital implementation of the identification technique can feasibly be constructed and used in an experimental test setup under the various constraints presented above.

2. Hybrid Implementation

A hybrid implementation of the identification technique is shown in Figure 2 for a test system with two poles.

This implementation differs from the digital implementation, in that the input and output of the system under test are integrated twice using analog integrators prior to sampling via A/D converters. These samples are stored for future nonreal-time processing on a digital computer.

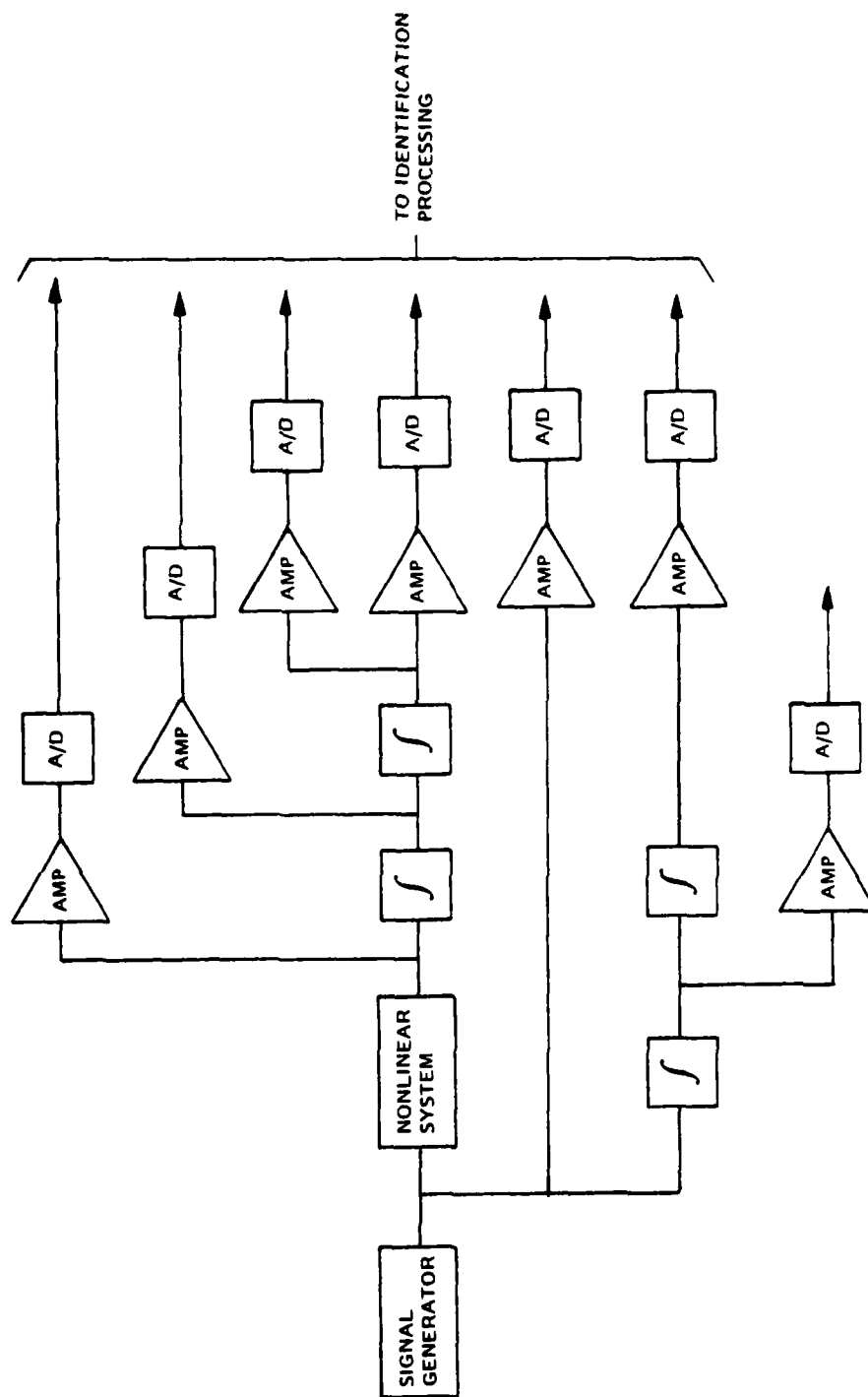


Figure 2. Hybrid Implementation

The analyses of the performance of this implementation led to the following conclusions:

- (1) The resolution requirements for the hybrid implementation are significantly greater than for the digital implementation. The hybrid implementation requires 24 bits which is beyond the current state of the art in A/D converter technology. The conversion speed requirements are essentially the same as those required for the digital implementation.
- (2) For systems with two poles or less, the hybrid implementation offers no advantages over the digital implementation. For systems with more than two poles, the hybrid implementation offers potential performance improvement over the digital approach for an A/D converter with 24 bits. This improvement however increases the measurement implementation complexity and cost.
- (3) The amplifier and signal generator requirements are the same as those derived for the digital implementation.
- (4) The amount of data to be stored is $[N + (1/2)]$ times greater for the hybrid implementation than for the digital implementation (N is the number of poles in the linear portion of the system under test).

It is not feasible to consider implementation of this approach for an experimental validation at this time because of the A/D converter requirements.

However, future improvements in A/D converter technology may permit implementation of this approach at that time.

3. Analog Implementation

An analog implementation of the identification technique is shown in Figure 3.

The analog implementation derives the necessary inner product quantities using analog components. The inner product device outputs are sampled and stored for further nonreal-time processing on a digital computer.

The performance evaluation has shown that the critical components in the analog implementation are the inner product

device and the A/D converter; this has led to the following conclusions.

- (1) A minimum of 18 bits of A/D converter resolution is required to achieve minimum identification performance. Conversion speed is not important since only one sample per inner product device output is required.
- (2) The maximum tolerable error in the inner product output is on the order of 10^{-3} percent to achieve a minimum level of identification performance for A/D converters with 18 or more bits of resolution.
- (3) Performance improvement requires less inner product error (10^{-4} to 10^{-6} percent) and increased A/D converter resolution (20 to 24 bits). However, the performance of the analog implementation is below that demonstrated for the digital implementation.
- (4) Currently available analog multipliers have an output error on the order of 0.05 percent, which is approximately 50 times greater than the maximum tolerable error of 0.001 percent required for minimum performance of the identification technique.
- (5) Amplifier and signal generator requirements are essentially the same as those derived for the digital implementation.
- (6) Data storage requirements are significantly reduced for the analog implementation. Only $(4N + 1)$ data words need to be stored (N is the number of poles in linear portions of systems under test).

The analog multiplier and the A/D converter requirements for the analog implementation imply that it is not feasible to consider this implementation for an experimental test setup in the present time frame. Significant technological developments for analog multipliers and A/D converters are necessary before this implementation can prove feasible.

SECTION II

IDENTIFICATION TECHNIQUE

A. IDENTIFICATION TECHNIQUE DEVELOPMENT

1. Background

The basic objective of this study (Part I) is to investigate the implementation feasibility of an identification technique for nonlinear systems. The identification technique is described in detail in this section and is based on the analysis presented in Reference 1. This technique is a "black box" procedure in that only measurements at the system input and output terminals are required. The feasibility of implementing a test setup to make the required data measurements of the input and output is the primary focus of this study (Part I). The identification technique is applicable to a class of weakly nonlinear systems whose behavior is adequately characterized in terms of a finite Volterra functional series given by

$$y(t) = \sum_{n=1}^{\bar{N}} y_n(t) = \sum_{n=1}^{\bar{N}} \int h_n(\tau_1, \dots, \tau_n) \prod_{p=1}^n x(t - \tau_p) d\tau_p, \quad (1)$$

where

$y_n(t)$ is the n -th order portion of the response

$\int$ denotes an n -fold integration from $-\infty$ to $+\infty$

$\prod_{p=1}^n$ denotes an n -fold product

The n th-order Volterra kernel $h_n(\tau_1, \dots, \tau_n)$ can be referred to as the n th-order nonlinear impulse response (Reference 2). In actuality, the nonlinear impulse responses may not be identically zero above order $\bar{N}$. However, the finite sum of

equation (1) implies that higher-order terms contribute negligibly to the output.

As is the case with linear systems, there is a corresponding representation in the Laplace transform domain. The n^{th} -order nonlinear transfer function, which is defined to be the n -dimensional Laplace transform of $h_n(\tau_1, \dots, \tau_n)$, is given by

$$H_n(s_1, \dots, s_n) = \int h_n(\tau_1, \dots, \tau_n) \prod_{p=1}^n e^{-s_p \tau_p} d\tau_p. \quad (2)$$

Closely related to $y_n(t)$ is the multidimensional time function

$$y_n(t_1, \dots, t_n) = \int h_n(\tau_1, \dots, \tau_n) \prod_{p=1}^n x(t_p - \tau_p) d\tau_p. \quad (3)$$

It is observed that $y_n(t_1, \dots, t_n) = y_n(t)$ when $t_1 = \dots = t_n = t$. If the n -dimensional Laplace transform of $y_n(t_1, \dots, t_n)$ is denoted by $Y_n(s_1, \dots, s_n)$, it follows from equation (3) that

$$Y_n(s_1, \dots, s_n) = H_n(s_1, \dots, s_n) \prod_{p=1}^n X(s_p) \quad (4)$$

where $X(s)$ is the conventional one-dimensional Laplace transform of $x(t)$. $Y_n(s_1, \dots, s_n)$ is reduced to $Y_n(s)$, the Laplace transform of $y_n(t)$, by applying the "association of variables" technique introduced by George (Reference 3). This approach implies that the nonlinear system is completely characterized by

the nonlinear impulse responses or, equivalently, the nonlinear transfer functions. Once either of these is known, the system response can be determined for arbitrary inputs. The problem of identifying a weakly nonlinear system therefore, consists of identifying the nonlinear impulse responses, by $h_n(t_1, t_2, \dots, t_n)$, $n = 1, 2, \dots, \bar{N}$.

The identification technique developed in Reference 1 is designed to identify the parameters of closed-form expressions for the nonlinear impulse responses, $h_n(t_1, t_2, \dots, t_n)$, $n = 1, 2, \dots, \bar{N}$. The analysis presented in Reference 1 demonstrates how the technique identifies the parameters of $h_1(t)$, $h_2(t_1, t_2)$ and $h_3(t_1, t_2, t_3)$. On the basis of this analysis, it is believed that the technique is extendable to identification of higher order nonlinear impulse responses ($\bar{N} > 4$). This study (Part I) is concerned with the feasibility of implementing the identification of only the linear and second-order nonlinear impulse responses, $h_1(t)$ and $h_2(t_1, t_2)$.

A functional form for $h_2(t_1, t_2)$ for a broad class of nonlinear systems is presented below.

2. Functional Form for $h_2(t_1, t_2)$

In Volterra analysis, $h_1(t)$ is the impulse response usually associated with the linear incremental model of a nonlinear system. When this model consists of linear resistors, capacitors, inductors, and controlled sources, the linear impulse response is described by

$$h_1(t) = \begin{cases} \sum_{i=1}^N R_i e^{\lambda_i t} & , t \geq 0 \\ 0 & , t < 0 \end{cases} \quad (5)$$

where $\text{Re} \{ \lambda_i \} \leq 0$ and it is assumed that the λ_i are distinct.

In general, a weakly nonlinear system contains several nonlinearities. If the nonlinearities can be modeled by nonlinear resistors, capacitors, inductors, and controlled sources whose current-voltage relationships can be expanded into power series as shown in Table 1 and if $h_1(t)$ is given by equation (5), the second-order nonlinear impulse response can be expressed in the symmetrical form (Reference 1):

TABLE 1. CURRENT-VOLTAGE RELATIONSHIPS OF
POSSIBLE NONLINEARITIES

- 1) Zero Memory, Independent Nonlinearity

$$i = K(v) = \sum_{j=1}^{\infty} K_j v^j$$

- 2) Zero Memory, Dependent Nonlinearity

$$i = G(u, v) = \sum_{\substack{j=0 \\ j \neq k=0}}^{\infty} \sum_{k=0}^{\infty} g_{jk} u^j v^k$$

- 3) Capacitive, Independent Nonlinearity

$$i = \frac{d}{dt} Q(v) = \frac{d}{dt} \sum_{j=1}^{\infty} \gamma_j v^j$$

- 4) Inductive, Independent Nonlinearity

$$i = \sum_{j=1}^{\infty} \psi_j \left[\int_{-\infty}^t v(z) dz \right]^j$$

where

v = incremental voltage across the element

i = incremental current through the element

u = incremental voltage elsewhere in the circuit.

$$\begin{aligned}
h_2(t_1, t_2) = & \sum_{k_1=1}^M \sum_{k_2=1}^N A_{k_1 k_2} e^{a_{k_1} t_1 + a_{k_2} t_2} U(t_2 - t_1) \\
& + \sum_{k_1=1}^M \sum_{k_2=1}^N A_{k_1 k_2} e^{a_{k_1} t_2 + a_{k_2} t_1} U(t_1 - t_2)
\end{aligned} \quad (6)$$

where

$$M = N^2 + 1. \quad (7)$$

$$U(t) = \begin{cases} 1 & , t > 0 \\ 0 & , t < 0 \end{cases} \quad (8)$$

and where the natural frequencies in equation (6) are related to those in equation (5) according to:

$$\begin{aligned}
a_1 &= \lambda_1, a_2 = \lambda_2, \dots, a_N = \lambda_N, \\
a_{N+1} &= \lambda_1 - \lambda_1 = 0, a_{N+2} = \lambda_1 - \lambda_2, \dots, a_{2N} = \lambda_1 - \lambda_N \\
a_{2N+1} &= \lambda_2 - \lambda_1, a_{2N+2} = \lambda_2 - \lambda_3, \dots, a_{3N-1} = \lambda_2 - \lambda_N \\
a_{N2-N+3} &= \lambda_N - \lambda_1, a_{N2-N+4} = \lambda_N - \lambda_2, \dots, \\
a_{N2+1} &= \lambda_N - \lambda_{N-1}.
\end{aligned} \quad (9)$$

The ordering of the a_{k_1} terms in equation (6) assumes all the factors $\lambda_i - \lambda_j$ to be distinct, such that $\lambda_i - \lambda_j \neq \lambda_k$ for any $i, j, k = 1, \dots, N$. Also, the zero entry that results from $\lambda_i - \lambda_j$ when $i = j$ is included only once as the entry a_{N+1} . In addition, it is readily shown that (Reference 1)

$$A_{k_1 k_2} = A_{k_2 k_1} \quad \text{for } k_1, k_2 \leq N \quad (10)$$

and that the coefficients of terms in equation (8) having the form

$$e^{(\lambda_i - \lambda_j)t_1 + \lambda_i t_2}, \quad i \neq j$$

are identically zero.

The identification technique will identify the parameters of $h_2(t_1, t_2)$ as represented in equation (6).

3. Identification Technique Description

The functional form for $h_2(t_1, t_2)$ established in equation (6) implies that the identification of $h_2(t_1, t_2)$ reduces to identification of the parameters a_{k1} , a_{k2} , A_{k1k2} and N . However, equations (5) and (9) show that a_{k1} , a_{k2} and N can be determined once the linear impulse response is known. Therefore, the task of identifying these parameters reduces to the task of identifying $h_1(t)$. The problem of identifying the coefficients A_{k1k2} still remains.

The identification process separates into two distinct steps: (1) identification of $h_1(t)$; and (2) identification of the A_{k1k2} quantities of $h_2(t_1, t_2)$. These two steps are considered below.

a. Identification of the Linear Impulse Response, $h_1(t)$

The first step in the identification of $h_1(t)$, the linear impulse response of a nonlinear system, is to excite the system with an input amplitude such that the output is linear. The amplitude of this signal can be determined by exciting the system with a sinusoidal signal of amplitude A and performing a spectral analysis of the resultant response. Amplitude A is then adjusted until the amplitude level of the harmonic frequencies of the output becomes sufficiently small compared to the level of the fundamental component. The following analysis assumes that the output of the nonlinear system represents the linear response of the systems described by

$$h_1(t) = \begin{cases} \sum_{i=1}^N R_i e^{\lambda_i t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (11)$$

where $\text{Re } \{\lambda_i\} < 0$ and the λ_i are distinct. The λ_i and R_i will be identified using the pencil-of-functions approach (Reference 4).

The pencil-of-functions approach operates on the system as shown in Figure 4, where the input to the linear system and resulting output are integrated N times over the real-time interval $(0, T)$. The following notation is used in Figure 4.

$$x_{i+1}(t) = \begin{cases} \int_0^t x_i(\tau) d\tau & 0 \leq t \leq T \\ 0 & \text{elsewhere} \end{cases} \quad (12)$$

$i = 1, \dots, N$

$$\bar{y}_{i+1}(t) = \begin{cases} \int_0^t \bar{y}_i(\tau) d\tau & 0 \leq t \leq T \\ 0 & \text{elsewhere} \end{cases} \quad (13)$$

$i = 1, \dots, N$

$$X_1(s) = \mathcal{L} \{x_1(t)\} = \int_0^T x(t) e^{-st} dt \quad (14)$$

$$X_{i+1}(s) = \frac{X_i(s)}{s} \quad i=1, \dots, N \quad (15)$$

$$\bar{Y}_1(s) = \mathcal{L} \{\bar{y}_1(t)\} = \int_0^T y(t) e^{-st} dt \quad (16)$$

$$\bar{Y}_{i+1}(s) = \frac{\bar{Y}_i(s)}{s} \quad i=1, \dots, N \quad (17)$$

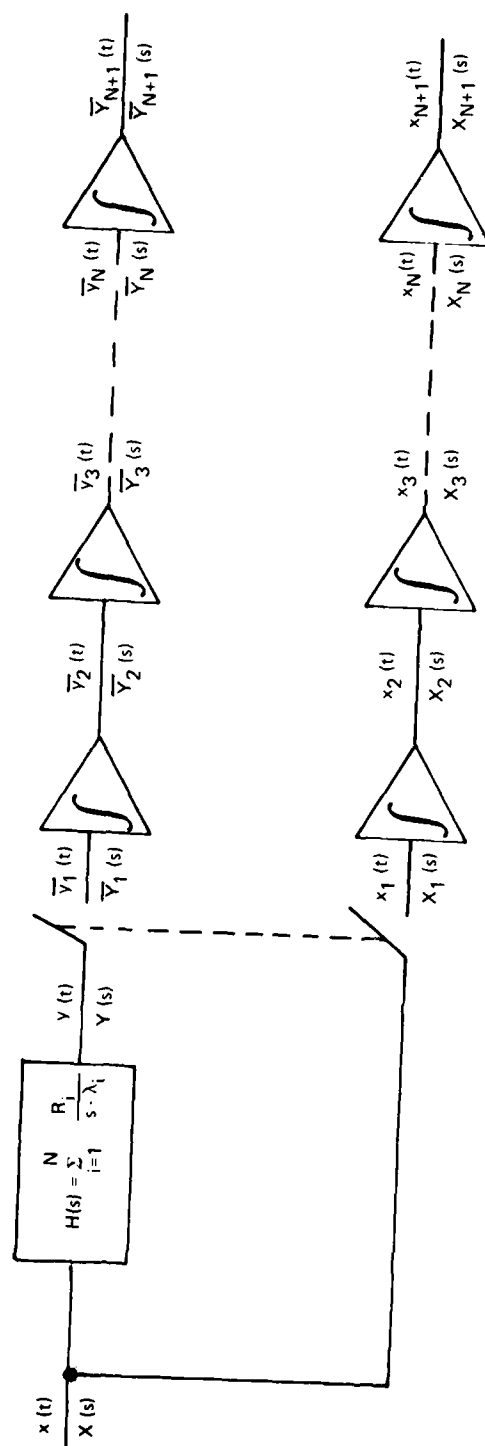


Figure 4. Identification Technique Configuration with N Integrators

It has been shown (Reference 4) that poles of the linear system satisfy the polynomial equation

$$\sum_{i=0}^N \lambda^{N-i} \left(\left| G_{2N+1} \right|_{i+1, i+1} \right)^{1/2} = 0 \quad (18)$$

where G_{2N+1} is the Gram determinant shown in equation (19) below:

$$G_{2N+1} = \begin{vmatrix} \langle \bar{y}_1, \bar{y}_1 \rangle & \langle \bar{y}_1, \bar{y}_2 \rangle & \dots & \langle \bar{y}_1, \bar{y}_{N+1} \rangle & \langle \bar{y}_1, x_2 \rangle & \dots & \langle \bar{y}_1, x_{N+1} \rangle \\ \langle \bar{y}_2, \bar{y}_1 \rangle & \langle \bar{y}_2, \bar{y}_2 \rangle & \dots & \langle \bar{y}_2, \bar{y}_{N+1} \rangle & \langle \bar{y}_2, x_2 \rangle & \dots & \langle \bar{y}_2, x_{N+1} \rangle \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \langle \bar{y}_N, \bar{y}_1 \rangle & \langle \bar{y}_N, \bar{y}_2 \rangle & \dots & \langle \bar{y}_N, \bar{y}_{N+1} \rangle & \langle \bar{y}_N, x_2 \rangle & \dots & \langle \bar{y}_N, x_{N+1} \rangle \\ \langle x_2, \bar{y}_1 \rangle & \langle x_2, \bar{y}_2 \rangle & \dots & \langle x_2, \bar{y}_{N+1} \rangle & \langle x_2, x_2 \rangle & \dots & \langle x_2, x_{N+1} \rangle \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \langle x_{N+1}, \bar{y}_1 \rangle & \langle x_{N+1}, \bar{y}_2 \rangle & \dots & \langle x_{N+1}, \bar{y}_{N+1} \rangle & \langle x_{N+1}, x_2 \rangle & \dots & \langle x_{N+1}, x_{N+1} \rangle \end{vmatrix} \quad (19)$$

Further, the residues R_i of the poles λ_i satisfy the equation

$$R = C^{-1} Y \quad (20)$$

where

$$R = \text{residue matrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_N \end{bmatrix} \quad (21)$$

$$Y = \text{output matrix} = \begin{bmatrix} \bar{y}_2(T) \\ \bar{y}_3(T) \\ \bar{y}_4(T) \\ \vdots \\ \bar{y}_{N+1}(T) \end{bmatrix} \quad (22)$$

$C = N \times N$ matrix whose i, j^{th} element is defined by

$$C_{ij} = \frac{P_j(T)}{\lambda_j^i} - \sum_{m=1}^i \frac{x_{m+1}(T)}{(\lambda_j)^{i+1-m}} \quad (23)$$

where

$$P_j(T) = \int_0^T e^{\lambda_j(T-\tau)} x(\tau) d\tau \quad (24)$$

Equations (18), (20), (23), and (24) show how R_i and λ_i are obtained once N is known. The pencil of functions technique also permits determination of N , the number of poles of the linear system. This can be accomplished as follows:

Using the system shown in Figure 4:

- (1) Assume $N = 1$
- (2) Measure $\bar{y}_i(t)$, $i=1, \dots, N+1$, and $x_i(t)$, $i=2, \dots, N+1$
- (3) Form the Gram determinant G_{2N+1} as given by equation (19)
- (4) Check singularity of G_{2N+1}

- (5) If G_{2N+1} is nonsingular, increase N by 1 and repeat from step (2)
- (6) If G_{2N+1} is singular, then the number of poles is $N - 1$

Therefore, the pencil-of-functions system identification technique completely specifies $h_1(t)$ as given in equation (11).

b. Identification of the Second Order Impulse Response, $h_2(t_1, t_2)$

The second step of the identification procedure is to identify the unknown parameters of $h_2(t_1, t_2)$. With $h_2(t_1, t_2)$ given by:

$$h_2(t_1, t_2) = \sum_{k_1=1}^M \sum_{k_2=1}^N A_{k_1 k_2} e^{a_{k_1} t_1 + a_{k_2} t_2} U(t_2 - t_1) + \sum_{k_1=1}^M \sum_{k_2=1}^N A_{k_1 k_2} e^{a_{k_1} t_2 + a_{k_2} t_1} U(t_1 - t_2) \quad (25)$$

the only unknown parameters are the $A_{k_1 k_2}$ quantities since M , N , A_{k_1} and A_{k_2} are known from identification of $h_1(t)$. A procedure for determining the $A_{k_1 k_2}$ using the pencil-of-functions method is described in this section.

The identification procedure utilizes the response of the weakly nonlinear system to a sum of L decaying exponentials as described by:

$$x(t) = \begin{cases} \sum_{i=1}^L e^{-\alpha_i t} & , \quad t \geq 0 \\ 0 & , \quad t < 0 \end{cases} \quad (26)$$

where $\text{Re } \{\alpha_i\} > 0$. The second-order portion of the response to $x(t)$ is given by the two-dimensional transform

$$Y_2(s_1, s_2) = H_2(s_1, s_2) X(s_1) X(s_2) \quad (27)$$

Taking the one-dimensional Laplace transform of $x(t)$ and the two-dimensional Laplace transform of $h_2(t_1, t_2)$ and substituting into equation (27) results in

$$Y_2(s_1, s_2) = \sum_{k_1=1}^M \sum_{k_2=1}^N \sum_{i=1}^L \sum_{j=1}^L A_{k_1 k_2} \cdot \left(\frac{s_1 + s_2 - a_{k_2}}{(s_1 + s_2 - a_{k_1} - a_{k_2})(s_1 - a_{k_2})(s_2 - a_{k_2})} \right) \left(\frac{1}{s_1 + \alpha_i} \right) \left(\frac{1}{s_2 + \alpha_j} \right) \cdot \quad (28)$$

Applying George's "association of variables" technique (Reference 3), $Y_2(s)$ becomes

$$Y_2(s) = \sum_{k_1=1}^M \sum_{k_2=1}^N \sum_{i=1}^L \sum_{j=1}^L A_{k_1 k_2} \cdot \left[\frac{\alpha_i + \alpha_j + 2a_{k_1}}{(\alpha_j + a_{k_1})(\alpha_i + a_{k_1})(\alpha_i + \alpha_j + a_{k_1} + a_{k_2})} \frac{1}{s - (a_{k_1} + a_{k_2})} \right. \\ - \frac{1}{(\alpha_j + a_{k_1})(\alpha_i + a_{k_2})} \frac{1}{s + (\alpha_j - a_{k_2})} \\ - \frac{1}{(\alpha_i + a_{k_1})(\alpha_j + a_{k_2})} \frac{1}{s + (\alpha_i - a_{k_2})} \\ \left. + \frac{\alpha_i + \alpha_j + 2a_{k_2}}{(\alpha_j + a_{k_2})(\alpha_i + a_{k_2})(\alpha_i + \alpha_j + a_{k_1} + a_{k_2})} \frac{1}{s + \alpha_i + \alpha_j} \right] \quad (29)$$

where

$$\begin{aligned}
 \alpha_j &\neq a_{k_1} && \text{for } j = 1, \dots, L; k_1 = 1, \dots, M \\
 \alpha_i + \alpha_j + a_{k_1} + a_{k_2} &\neq 0 && \text{for } i, j = 1, \dots, L; k_1 = 1, \dots, M; \\
 &&& k_2 = 1, \dots, N \\
 a_{k_1} &\neq a_{k_2} && \text{for } k_2 = 1, \dots, N; k_1 = N + 1, \dots, M.
 \end{aligned}
 \tag{30}$$

The expression in equation (29) is the Laplace transform of a sum of exponential time functions. This sum can be interpreted as the impulse response of an equivalent linear system as indicated in Figure 5. In other words, the second-order response $y_2(t)$ can be visualized as though it were generated by an equivalent linear system. However, the equivalence is valid only if the equivalent linear system is considered to be excited by an impulse. It follows that the problem of identifying $h_2(t_1, t_2)$ has been reduced to the simpler problem of identifying a linear system and the pencil-of-functions technique can be used again.

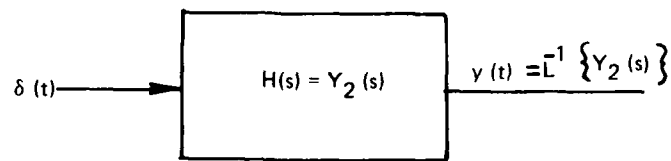


Figure 5. Equivalent Linear System with Transfer Function $Y_2(s)$

The system is excited by an input amplitude such that the output is described by linear and second-order terms, $y_1(t)$ and $y_2(t)$. The identification process will operate the signal $y_2(t)$. For this purpose, the second-order portion of the response, $y_2(t)$ is isolated from the total response. $y_2(t)$ is obtained by subtracting from the total response the corresponding linear response $y_1(t)$, which is known because $h_1(t)$ has been identified. It is shown in Reference 1 that the second-order response $y_2(t)$ need not be isolated from the total response for the identification procedure to work. (This will be investigated in detail in Part II of the study.) However, isolation of $y_2(t)$ from the total response eases the mathematical presentation and will be assumed necessary at this point, to identify implementation constraints.

Once $y_2(t)$ is isolated from the total response, the coefficients $A_{k_1 k_2}$ are then evaluated by applying the pencil-of-functions method to $y_2(t)$, treating it as though it were the impulse response of a linear system. This latter step is now discussed in detail.

From equation (29), the poles of $Y_2(s)$ are given by

$$\begin{aligned} s &= a_{k_1} + a_{k_2} & , k_1 &= 1, \dots, M; k_2 = 1, \dots, N \\ s &= -\alpha_i + a_{k_2} & , i &= 1, \dots, L; k_2 = 1, \dots, N \\ s &= -\alpha_i - \alpha_j & , i, j &= 1, \dots, L. \end{aligned} \quad (31)$$

First, consider poles of the form $s = a_{k_1} + a_{k_2} = 2\lambda_\ell$; $\ell = 1, \dots, N$. The terms in $Y_2(s)$ corresponding to the pole at $2\lambda_\ell$ are given by

$$Y_{2\ell\ell}(s) = \sum_{i=1}^L \sum_{j=1}^L A_{\ell\ell} \frac{1}{(\alpha_j + \lambda_\ell)(\alpha_i + \lambda_\ell)} \frac{1}{s - 2\lambda_\ell}. \quad (32)$$

If the residue of the pole at $2\lambda_\ell$, as evaluated using the pencil-of-functions method, is $\beta_{\ell\ell}$, it follows that

$$A_{\ell\ell} = \beta_{\ell\ell} \sum_{i=1}^L \sum_{j=1}^L \frac{1}{(\alpha_j + \lambda_\ell)(\alpha_i + \lambda_\ell)} \quad \ell = 1, \dots, N. \quad (33)$$

This procedure results in identification of N of the coefficients.

Consider next poles of the form $s = a_{k1} + a_{k2} = \lambda_\ell + \lambda_m$ where $\ell \neq m$ and $\ell, m = 1, \dots, N$. Since $A_{\ell m} = A_{m\ell}$ for $\ell, m \leq N$, the terms in $Y_2(s)$ corresponding to the pole at $\lambda_\ell + \lambda_m$ are given by

$$Y_{2\ell m}(s) = \sum_{i=1}^L \sum_{j=1}^L A_{\ell m} \left[\frac{\alpha_i + \alpha_j + 2\lambda_\ell}{(\alpha_j + \lambda_\ell)(\alpha_i + \lambda_\ell)(\alpha_i + \alpha_j + \lambda_\ell + \lambda_m)} + \frac{\alpha_i + \alpha_j + 2\lambda_m}{(\alpha_j + \lambda_m)(\alpha_i + \lambda_m)(\alpha_i + \alpha_j + \lambda_\ell + \lambda_m)} \frac{1}{s - \lambda_\ell - \lambda_m} \right]. \quad (34)$$

If the residue of the pole at $\lambda_\ell + \lambda_m$, as evaluated using the pencil-of-functions method, is $\beta_{\ell m}$, it follows that

$$A_{\ell m} = \beta_{\ell m} \left\{ \sum_{i=1}^L \sum_{j=1}^L \left[\frac{\alpha_i + \alpha_j + 2\lambda_\ell}{(\alpha_j + \lambda_\ell)(\alpha_i + \lambda_\ell)(\alpha_i + \alpha_j + \lambda_\ell + \lambda_m)} + \frac{\alpha_i + \alpha_j + 2\lambda_m}{(\alpha_j + \lambda_m)(\alpha_i + \lambda_m)(\alpha_i + \alpha_j + \lambda_\ell + \lambda_m)} \right] \right\}^{-1} \quad \begin{matrix} \ell, m = 1, \dots, N \\ \ell \neq m, \ell < m. \end{matrix} \quad (35)$$

This procedure results in identification of $N(N-1)/2$ of the coefficients.

The remaining unknown N^2 coefficients cannot be evaluated directly, as was done in equation (33) and (35), because the residues of the other poles in $Y_0(s)$ involve linear combinations of more than one unknown coefficient. However, if the number of exponential input signals, L , is set equal to N , N^2 linearly independent equations involving the N^2 unknown A_k/k^2 coefficients can be obtained by considering the poles of $Y_0(s)$ of the form $s = -\lambda_1 + j\lambda_2, i = 1, \dots, N, j = 1, \dots, N$, in a manner similar to the above analysis. This fact is proven in Reference 1. Solution of the N^2 equations completes the identification process.

B. IDENTIFICATION TECHNIQUE PROCESSING

The required identification technique processing is done on a nonreal time basis using a general purpose digital computer. The processing uses the input and output measurements of the system under test, which are obtained in real time. The computer program used to perform the identification technique processing is described in this section.

The computer program listing is presented in Appendix A. The inputs to the program include:

- (1) Number of output samples, M
- (2) Sampling interval
- (3) System order, $3N$
- (4) Analytical description of input and output

This study assumes that the order of the system under test is known. In general, this information may or may not be known. A technique was discussed in paragraph A.3.2a that resulted in determination of system order using the pencil of functions approach. This technique may be limited by the numerical accuracy of the digital computer and is an issue that merits further attention. In general, determining system order is a key problem in system identification and an in depth analysis of this problem is beyond the scope of the current study.

An important aspect of the identification technique processing is the determination of the inner product entries of the Gram matrix (equation (19)). These entries are the results of several integrations of the input and output data. When analytical or analog integration is not used, numerical integration techniques are implemented.

The input and output functions are integrated using Simpson's rule of integration (Reference 5), given by

$$\int_a^b y(t) dt = \frac{(b-a)}{6n} \left[y(0) + 4y(\Delta T) + 2y(2\Delta T) + 4y(3\Delta T) + \dots + 2y((2n-2)\Delta T) + 4y((2n-1)\Delta T) + y(2n\Delta T) \right] \quad (36)$$

where

$$\Delta T = \frac{b-a}{2n} = \text{time between samples}$$

$2n$ = number of subintervals between data points.

The effect of numerical integration techniques on the performance of the identification technique will be addressed in Part II of this study.

The remaining processing involves matrix manipulations which are accomplished using standard FORTRAN computer subroutines, as listed in Appendix A.

C. PERFORMANCE INDICES FOR THE IDENTIFICATION TECHNIQUE

A first step in assessing the performance of given implementation of the identification technique is to establish a set of performance indices which adequately reflects that performance. This is not a simple task, as performance indices are very numerous and in many cases require subjective interpretation. The intent in this study has been to establish a set of performance indices that is used consistently for all implementations and that reflects, at least for comparison purposes, the performance of the identification technique.

Two primary performance indices were used during the study:

- (1) Percentage error between the predicted system poles and residues and the actual system poles and residues
- (2) Normalized mean-squared error between predicted system output response and actual system response.

The percentage error index of performance indicates how well the technique identifies each pole and residue of the test system. The percentage error is given by

$$E_{p_c \text{ pole}} = \frac{(\text{Predicted System Pole}) - (\text{Actual System Pole})}{\text{Actual System Pole}} \times 100 \quad (37)$$

$$E_{p_c \text{ residue}} = \frac{(\text{Predicted System Residue}) - (\text{Actual System Residue})}{\text{Actual System Residue}} \times 100 \quad (38)$$

The normalized mean squared error is defined by the equation

$$NMSE = \frac{\frac{1}{T} \int_0^T (y_p - y_a)^2 dt}{\frac{1}{T} \int_0^T y_a^2 dt} \quad (39)$$

where

$y_p(t)$ is the predicted system output to input $x(t)$

$y_a(t)$ is the actual system output to input $x(t)$

T is the period of integration

The normalized mean squared error performance index indicates how well the predicted system response approximates the actual response to the same input. It is possible to construct this performance index in this study because the actual system response is analytically known for the systems considered. This index of performance is better than the percentage error in the sense that it evaluates the influence of errors in the poles and residues on predicting system responses. It is possible that a very large error in a pole or residue (>100 percent) will have only a minor impact on normalized mean squared error since the pole and residue contribute negligibly to the total system output.

This study considers systems whose outputs generally consist of sums of exponential functions. In these cases, the predicted system output and actual system output are represented as:

$$y_p(t) = \sum_{i=1}^N R_{p_i} e^{-\lambda_{p_i} t} \quad t > 0 \quad (40)$$

$$y_a(t) = \sum_{i=1}^N R_{a_i} e^{-\lambda_{a_i} t} \quad t > 0 \quad (41)$$

The normalized mean squared error is given by

$$NMSE = \frac{F}{D} \quad (42)$$

where

$$\begin{aligned} F = & \sum_{i=1}^N \sum_{j=1}^N \left[\frac{R_{p_i} R_{p_j}}{(\lambda_{p_i} + \lambda_{p_j})} \left(e^{(\lambda_{p_i} + \lambda_{p_j}) T} - 1 \right) \right. \\ & - \frac{2 R_{p_i} R_{a_j}}{(\lambda_{p_i} + \lambda_{a_j})} \left(e^{(\lambda_{p_i} + \lambda_{a_j}) T} - 1 \right) \\ & \left. + \frac{R_{a_i} R_{a_j}}{(\lambda_{a_i} + \lambda_{a_j})} \left(e^{(\lambda_{a_i} + \lambda_{a_j}) T} - 1 \right) \right] \quad (43) \end{aligned}$$

$$D = \sum_{i=1}^N \sum_{j=1}^N \frac{R_{a_i} R_{a_j}}{(\lambda_{a_i} + \lambda_{a_j})} \left(e^{(\lambda_{a_i} + \lambda_{a_j}) T} - 1 \right) \quad (44)$$

One remaining issue in the area of performance indices is the interpretation of the results. The question is "How accurate must the identification technique be to achieve a satisfactory level of performance?" The absolute accuracy requirements depend on the application of the technique. If a precise description of the system under test is required, then percentage errors of 0.05 to 0.1 might be necessary. On the other hand, prediction of the poles of a system to within 10 to 20 percent may be more than adequate in some cases.

Similarly, the interpretation of the normalized mean square error (NMSE) also causes this dilemma. For this study, an arbitrary level of minimum performance has been established which corresponds to that level of NMSE induced by a 10-percent error in each pole and residue. This NMSE performance level is a function of the system under test and is established quantitatively for the test systems considered later in the report.

It should be noted that significantly better performance of the identification technique will generally be desired, but this level of minimum performance will permit determination of a set of minimum system requirements for each implementation of the technique.

SECTION III

IMPLEMENTATION FEASIBILITY STUDY

A. IMPLEMENTATION APPROACHES

The basic objective of this study phase is to establish requirements for implementation of the identification technique described in Section II. Three basic implementations were considered:

- (1) Digital - the input and output of the system under test are sampled using analog-to-digital (A/D) converters and all subsequent processing is done on a general-purpose (GP) computer in nonreal time.
- (2) Hybrid - the input and output of the system under test are integrated N times by analog integrators and the outputs of each integrator are sampled for further processing on a GP computer.
- (3) Analog - the pencil-of-functions processing is implemented using analog components and only the inner products for the Gram matrix are sampled using A/D converters. These samples are then used in a GP computer for solution of the appropriate equations.

These implementations are discussed in paragraphs B, C, and D below and the results of the implementation feasibility study are reported.

B. DIGITAL IMPLEMENTATION

The digital implementation of the identification technique is shown in Figure 6. A signal generator excites the nonlinear system with the appropriate sum of decaying exponential functions. This input and the resultant nonlinear system output are amplified and converted into digital form via A/D converters. The resultant samples are stored in memory for future nonreal-time processing on a general-purpose digital computer using the program described in Section II.A. These input and output samples are then numerically integrated to form the appropriate inner product entries of the Gram matrix for the pencil-of-functions method of system identification.

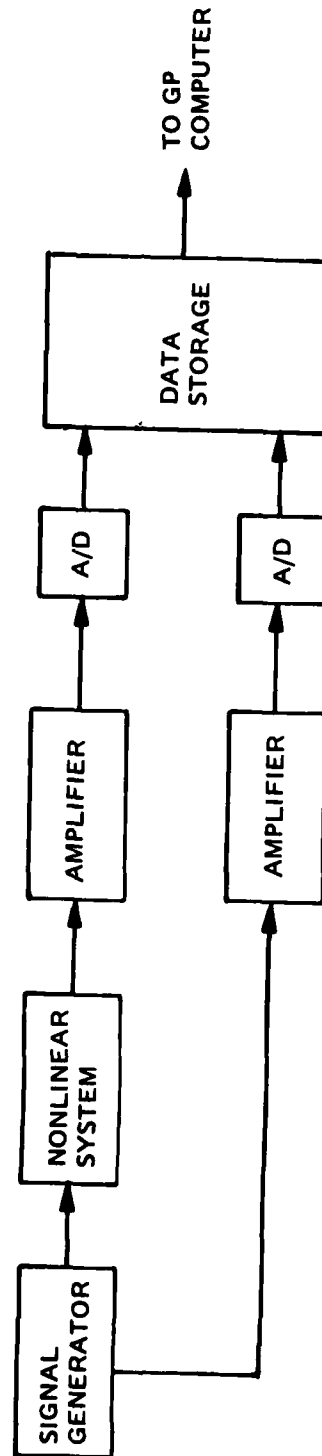


Figure 6. Digital Implementation of Identification Technique

The requirements for each element of the digital implementation of Figure 6 are discussed in detail below.

1. A/D Converter

- a. Requirements

A key element of the digital implementation of Figure 6 is the A/D converter used to obtain the input and output data samples for identification processing. The study determined the required performance specifications for the A/D converter to achieve satisfactory identification technique performance and also assessed the state of the art in commercially available A/D converters to determine if the required specifications can be met. Performance specifications for A/D converters are numerous but the major parameters impacting technique performance are resolution and conversion time. The requirements for these parameters were established during the study.

The identification processing computer simulation was modified to include an A/D converter model which is described in detail below.

- b. Simulation Model

The A/D converter simulation model is illustrated in Figure 7.

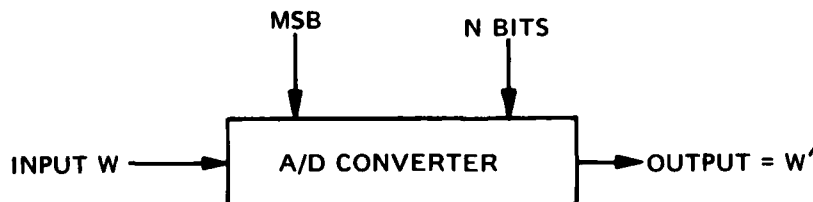


Figure 7. A/D Converter Simulation Model

The simulation inputs for the A/D converter are: (1) the most significant bit magnitude (MSB) and (2) the number of bits of resolution of the A/D converter (NBITS). The simulation converts the input to the A/D converter to a digital representation according to the input-output representation of Figure 8.

The level L in Figure 8 corresponds to the magnitude of the least significant bit. The least significant bit is related to the most significant bit by the relation

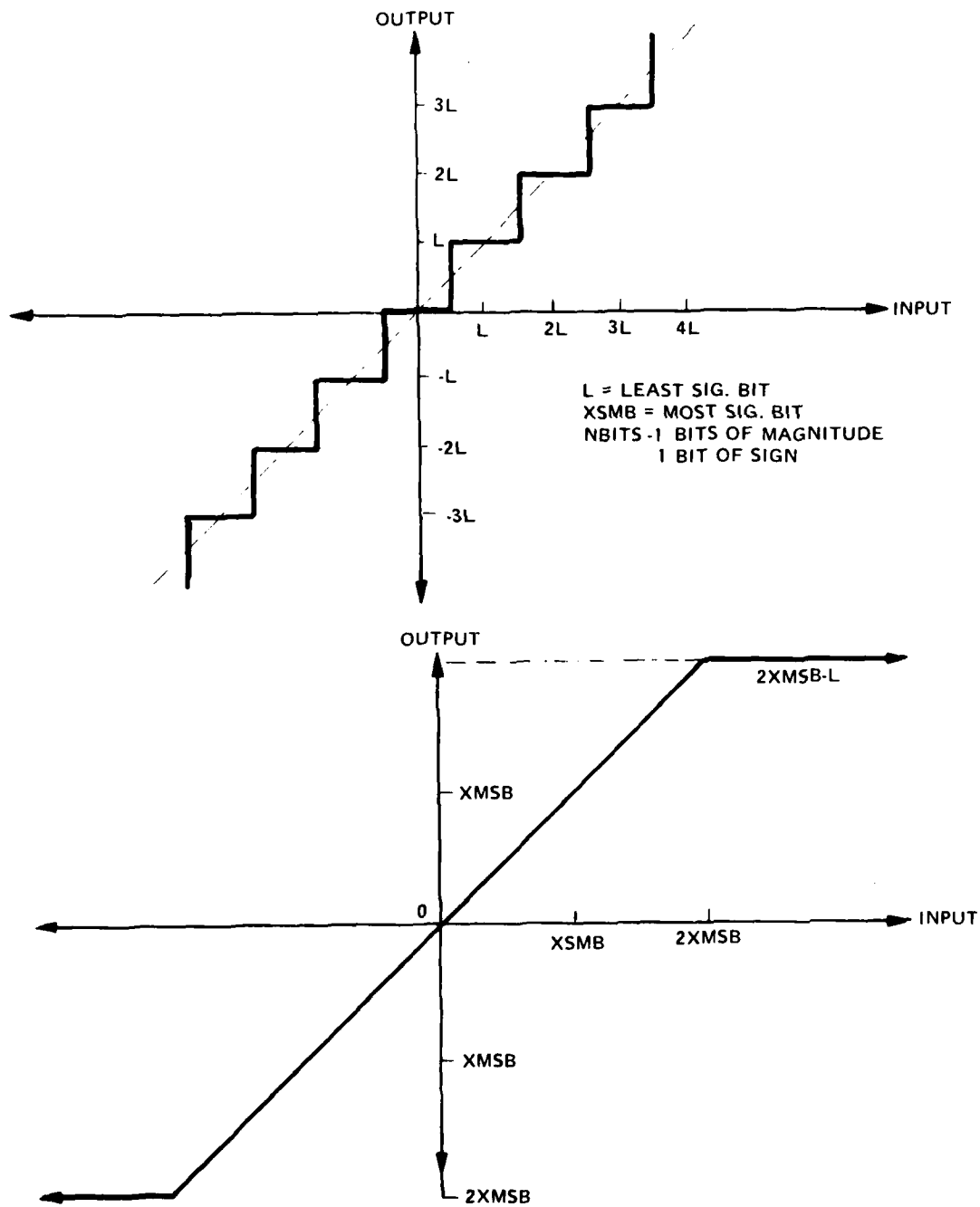


Figure 8. Digital Output Voltages vs. Input Voltage

$$L = \text{MSB}/2^{(\text{NBITS}-2)} \quad (45)$$

This assumes that one bit is used to indicate the sign (positive or negative) of the input signal.

The output of an A/D converter is a string of digital 1's and 0's that represent the input voltage level. The simulation model of the A/D converter produces a decimal number that is the equivalent of these 1's and 0's. The output of the A/D converter is given by

$$W' = \text{SIGN} \cdot \sum_{i=1}^{\text{NBITS}-1} \alpha_i \cdot \frac{(\text{MSB})}{2^{i-1}} \quad (46)$$

for $\alpha_i \in \{0, 1\}$

where $\text{SIGN} = \pm$ depending on polarity of input voltage W .

From Figure 8, if

$$2\text{MSB} - L < W < 2\text{MSB} - L \quad (47)$$

then the input voltage W is not hard limited by the A/D converter. This level determines the value of MSB to be used in the simulation to obtain greatest resolution for a given converter size.

The A/D converter simulation model is given in its FORTRAN representation in Figure 9.

Most commercially available A/D converters have a fixed full-scale input signal level (typically ± 5 or ± 10 volts). This input signal level determines the magnitude of the most significant bit at the A/D converter output. This implies that an input to the A/D converter will need to be amplified to the full scale input voltage level in order to obtain maximum resolution out of the A/D. This presents little problem to the identification technique if the amplifier is ideal (infinite bandwidth, linear) since its effect can be removed in the nonreal-time processing of the GP computer. Consequently, the simulation

```

SUBROUTINE ATOD(W,XMSB,NBITS)
  IF(NBITS.LE.0) GO TO 80
  U2=0.0
  SIGN=-1.
  IF(W.GE.0.0) SIGN=1.
  XW=W
  D=2.0*XMSB
  XW=XW*SIGN
20 DELTA2=XMSB
  DO 30 I=1,NBITS-1
30 DELTA2=DELTA2/2.
  WOUT=2.*(XMSB-DELTA2)
  DO 5 I=1,NBITS
    D=D/2.0
    Y=DABS(XW-U2)
    IF(Y.LE.DELTA2) WOUT=U2
    X=U2-D
    IF(XW.GE.U2) X=U2+D
  5 U2=X
    IF(XW.LT.DELTA2) WOUT=0.0
  IF(XW.GE.(2.*XMSB-D/2.)) PRINT,"A/D HARD LIMITED SAMPLE"
70 W=WOUT*SIGN
80 RETURN
  END

```

Figure 9. Listing of FORTRAN Subroutines for
A/D Converter

assumes, at this point of the study, that an ideal amplifier is used and the most significant bit of the A/D converter is set according to the level of the function input to it. The effects of a non-ideal amplifier are evaluated in paragraph B.9. below.

c. Performance Evaluation

Two representative systems were used to evaluate the performance of the identification technique and assess requirements for the components of the digital implementation. The first is a two-pole system whose linear and second-order impulse responses were determined in Reference 1. The second is a four-pole system introduced to evaluate the effects of system complexity (number of poles, N, in linear model) on technique performance. The details of these systems and the performance results are presented below.

(1) Two-Pole System Analysis and Results

The initial system selected for investigation was a two-pole linear system with an impulse response given by

$$h(t) = 2.8069192 \times 10^5 e^{-0.011550998 (2\pi \times 10^6)t} - 2.7368441 \times 10^8 e^{-10.616986 (2\pi \times 10^6)t} \quad (48)$$

This is a linear representation of the amplifier shown in Figure 10. The poles of the system are at 11.550998 KHz and 10.616986 MHz.

The input to the system was selected to be

$$x(t) = (1 \times 10^{-3}) e^{-10^7 t} u(t) \quad (49)$$

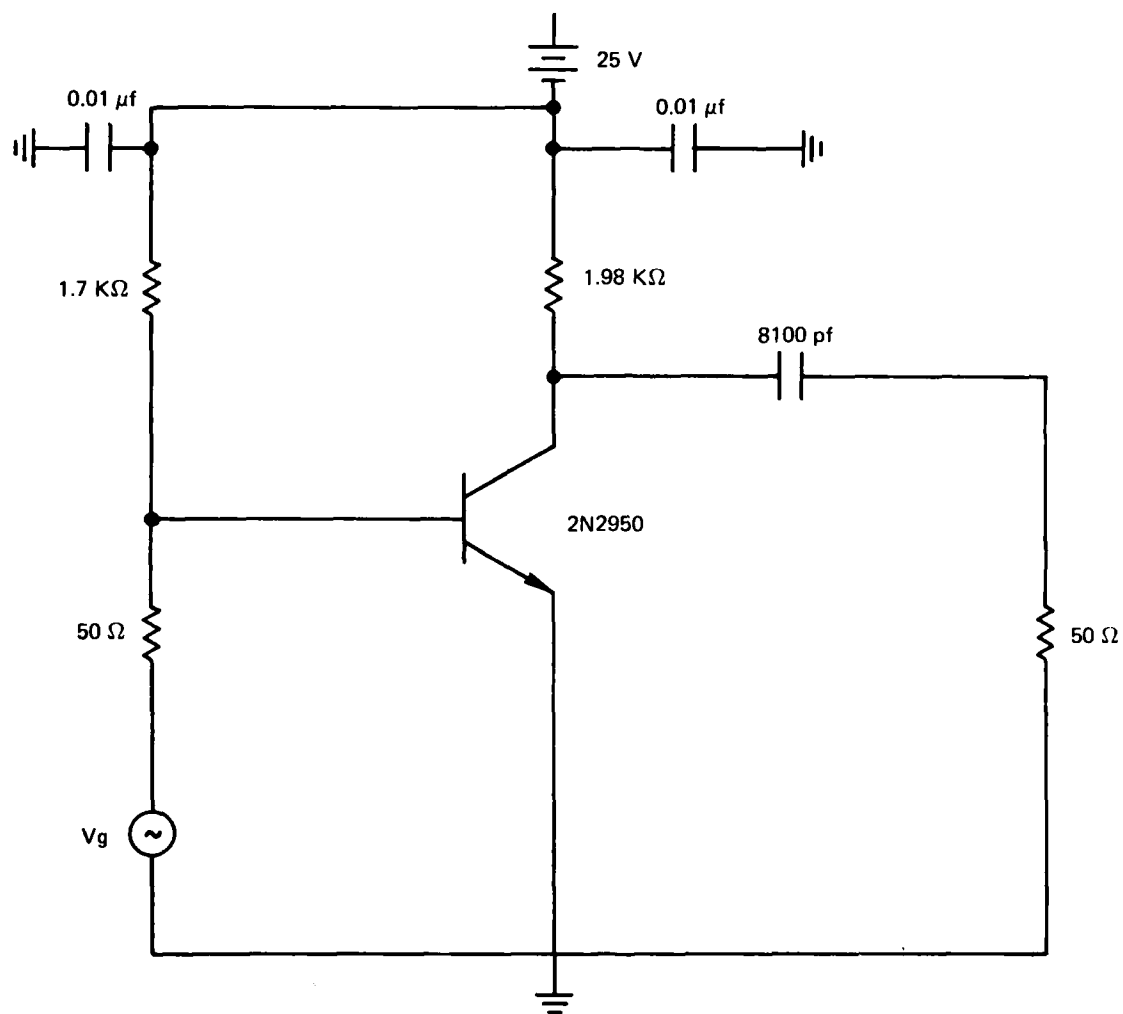


Figure 10. Common Emitter Amplifier Circuit

where $u(t)$ is a unit step function. This corresponds to the input level required to excite the circuit in Figure 10 in linear operation. The resultant output is given by

$$y(t) = \left(2.82744 \times 10^{-4} e^{-0.011550998 (2\pi \times 10^6)t} + 4.82616765 \times 10^{-2} e^{-10.616986 (2\pi \times 10^6)t} - 4.85444205 \times 10^{-2} e^{-10^7 t} \right), t \geq 0 \quad (50)$$

These representations of $y(t)$ and $x(t)$ were used in the computer program as the inputs to the A/D converter of Figure 6. To establish a baseline for performance comparison, the initial simulation run was made without an A/D converter. The accuracy of the samples was equivalent to the GP computer machine accuracy. This result is presented in Table 2 for an integration time of 9.6×10^{-6} second. Variation of the integration time and sampling interval indicated that the $9.6 \mu s$ integration time resulted in best performance. Results for other integration times/sampling intervals are shown later in this section. The results of Table 2 indicate that excellent performance is achieved using the identification technique. The percentage error on all poles and residues is less than 0.05 while the normalized mean squared error is 0.41×10^{-8} . As noted earlier, the minimum level of acceptable performance was set to the normalized mean squared error corresponding to a 10-percent error in each residue and pole. For the system described by equation (48), a 10-percent error in each pole and residue corresponds to a normalized mean square error of

$$NMSE_{(10\%)} = 0.63 \times 10^{-3}. \quad (51)$$

The simulation was exercised for various levels of A/D resolution from 8 to 24 bits and the results are presented in Table 3. Figure 11 is a plot of normalized mean squared error as a function of A/D converter resolution. Figure 11 and Table 3 results indicate that little performance improvement is gained for A/D converters in excess of 16 bits resolution. The minimum performance level is exceeded for A/D converters with 10 bits or more resolution.

TABLE 2. IDENTIFICATION RESULTS FOR PERFECT A/D CONVERTER (MACHINE ACCURACY),
DIGITAL IMPLEMENTATION (9.6 μ s INTEGRATION TIME)

System Poles (MHz)			System Residues		
Actual	Predicted	Percentage Error	Actual	Predicted	Percentage Error
0.011550998	0.011545154	-0.05	2.8069192 $\times 10^5$	2.8063116 $\times 10^5$	-0.02
10.615986	10.615088	-0.0187	-2.7368441 $\times 10^8$	-2.7362615 $\times 10^8$	-0.021
					0.411 $\times 10^{-8}$

TABLE 3. RESULTS FOR LEVELS OF A/D CONVERTER RESOLUTION FROM 8 TO 24 BITS

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Mean Squared Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz
No A/D Converter	0.011545154 10.615088	-0.05 -0.01787	2.8063166E5 -2.7362615E8	-0.02 -0.021	0.411 x 10 ⁻⁸	0.011
24	0.011550163 10.612980	-0.0072 -0.0377	2.8068418E5 -2.7357912E8	-0.00275 -0.0385	0.983 x 10 ⁻⁸	0.044
16	0.011537262 10.621939	-0.119 0.0466	2.8059592E5 -2.7381345E8	-0.0342 0.047	0.147 x 10 ⁻⁷	0.029
14	0.01157308 10.560624	0.191 -0.5308	2.8108237E5 -2.7233161E8	0.139 -0.494	0.174 x 10 ⁻⁵	0.042
12	0.011565797 10.575364	0.128 -0.392	2.8179798E5 -2.7294636E8	0.394 -0.27	0.185 x 10 ⁻⁵	0.059
10	0.010017392 10.355187	-13.2 -2.46	2.7057878E5 -2.678456E8	-3.6 -2.13	0.49 x 10 ⁻⁴	-1.06
8	0.0048517152 10.942193	-58. 3.06	-2.9278812E4 -1.1073848E8	-204.3 -59.5	0.367	Not Calculated

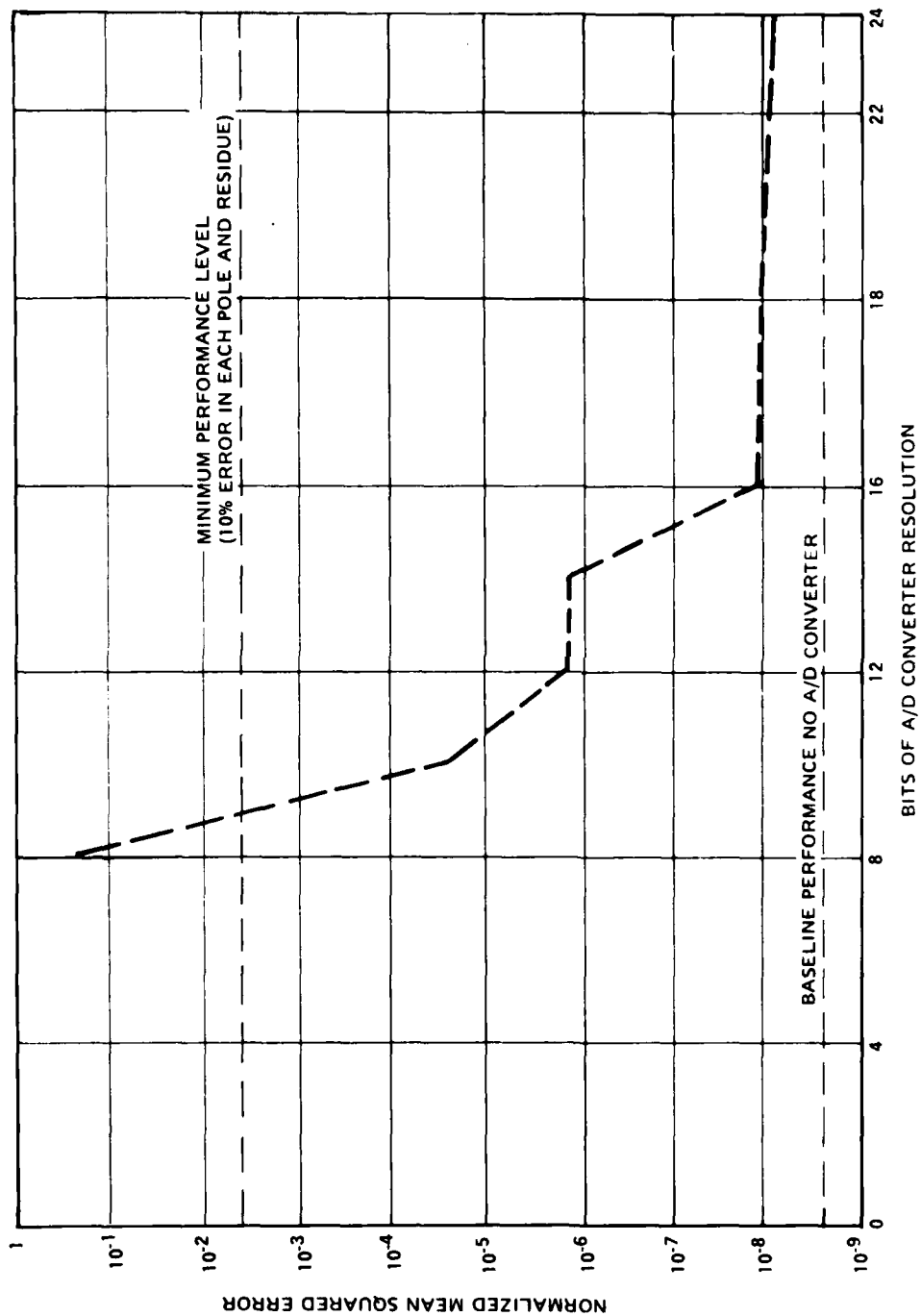


Figure 11. Normalized Mean Square Error vs. A/D Converter Resolution for Two-Pole System (Integration Time = 9.6 μ s; Sampling Time = 4 ns)

Another performance measure is indicated in Table 4 which corresponds to the maximum error in the magnitude squared of the transfer function, $|H(s)|^2$, over the frequency range of 1 kHz to 100 MHz. This error is less than 0.06 dB for A/D converters with more than 12 bits resolution.

These simulation runs were repeated for different integration time periods and sampling intervals. These results are tabulated in Tables 4 to 6 and are summarized in Figure 12. These results indicate that no significant difference in performance is noted for integration times of 4.8 μ s or greater.

(2) Four-Pole System Analysis and Results

The two-pole system identification example presented above leads to the conclusion that it is feasible to consider an experimental implementation of the identification technique (as far as A/D converter resolution requirements are concerned). It is necessary to consider systems with more than two poles because, as the system complexity (number of poles) increases, the computational load increases and it is expected that the impact of error in the input and output samples will be greater. Therefore, a four-pole system was investigated to determine the effect of system complexity on implementation requirements and performance.

The four-pole system considered had an impulse response given by

$$\begin{aligned} h(t) = & 2.8069192 \times 10^5 e^{-0.011550998 (2\pi \times 10^6)t} \\ & - 1.20 \times 10^7 e^{-0.510 (2\pi \times 10^6)t} \\ & + 1.51 \times 10^7 e^{-0.82 (2\pi \times 10^6)t} \\ & - 1.61 \times 10^8 e^{-6.50 (2\pi \times 10^6)t} \end{aligned} \quad (52)$$

Several trial runs led to the selection of a 4.8 μ s integration time and a 1.5 ns sampling interval. The initial set of results was run for a perfect A/D converter. These results are presented in Table 7. The results for this example indicate acceptable performance of the identification technique; however, the performance is not as good as was achieved for the two-pole example. There are several potential reasons for this lower level of performance. The first explanation is the increased

TABLE 4. RESULTS FOR LEVELS OF A/D CONVERTER RESOLUTION
INTEGRATION TIME = 7.2 μ s, SAMPLING INTERVAL = 3 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Mean Squared Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 1 MHz
No A/D Converter	0.011565417 10.608509	0.125 -0.08	2.8081012E5 -2.7347864E8	0.04 -0.075	0.387 x 10 ⁻⁷	-0.08
16	0.011541588 10.615394	-0.08 -0.015	2.8067905E5 -2.7365799E8	-0.0046 0.009	0.351 x 10 ⁻⁸	0.025
14	0.011528914 10.611083	-0.191 -0.055	2.8075629E5 -2.7360498E8	-0.0229 -0.029	0.734 x 10 ⁻⁷	0.067
12	0.011562609 10.600426	0.100 -0.156	2.8222309E5 -2.7367906E8	0.545 -0.00195	0.222 x 10 ⁻⁵	0.118
10	0.011648442 10.228063	0.843 3.66	2.8312792E5 -2.6608803E8	0.868 -2.77	0.132 x 10 ⁻³	-0.43
8	0.0053343822 10.863888	-53.8 2.32	-1.1663478E4 -1.6087431E8	-104.0 -41.2	0.18	Not Calculated

TABLE 5. RESULTS FOR LEVELS OF A/D CONVERTER RESOLUTION,
INTEGRATION TIME = 4.8 μ s, SAMPLING INTERVAL = 2 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Mean Squared Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz
No A/D Converter	0.011620899 10.612452	0.605 -0.0427	2.8107143E5 -2.7359561E8	0.135 -0.032	0.154 x 10 ⁻⁷	-0.177
16	0.011003596 10.644432	-4.74 0.258	2.7786488E5 -2.7420457E8	-1.0 0.19	0.622 x 10 ⁻⁶	0.968
14	0.011016497 10.642807	-4.63 0.243	2.7853243E5 -2.7432920E8	-0.77 0.236	0.404 x 10 ⁻⁶	0.985
12	0.011173364 10.633183	-3.27 0.152	2.8101151E5 -2.7447830E8	0.114 0.29	0.249 x 10 ⁻⁵	0.938
10	0.0044293814 10.813444	-61.6 1.85	2.9211082E5 -2.9038197E8	4.07 6.1	0.196 x 10 ⁻²	23.2
8	0.0027540105 10.783671	-76.2 1.57	-2.4200426E4 -1.3392599E8	-108.6 -51.	0.266	Not Calculated

TABLE 6. RESULTS FOR LEVELS OF A/D CONVERTER RESOLUTION,
INTEGRATION TIME = 2.4 μ s, SAMPLING INTERVAL = 1 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Mean Squared Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 1 MHz
No A/D Converter	0.012876359 10.613347	11.47 -0.034	2.8427807E5 -2.7372492E8	1.27 0.0148	0.204 x 10 ⁻⁶	3.96
16	0.018471276 10.595075	59.9 -0.206	3.0007228E5 -2.738586E8	6.9 0.0636	0.608 x 10 ⁻⁵	11.2
14	0.013670863 10.615823	18.35 -0.011	2.8783611E5 -2.7407198E8	2.54 0.1416	0.206 x 10 ⁻⁵	5.65
12	0.0083855822 10.630804	-27.4 0.13	2.8361473E5 -2.752973E8	1.04 0.589	0.239 x 10 ⁻⁴	10.0
10	0.015998973 10.728059	38.5 1.046	2.9614702E5 -2.793664E8	5.5 2.076	0.12 x 10 ⁻³	-9.2
8	0.013912348 10.757434	20.4 1.32	4.1746044E5 -2.9730088E8	48.7 8.63	0.526 x 10 ⁻²	-1.26

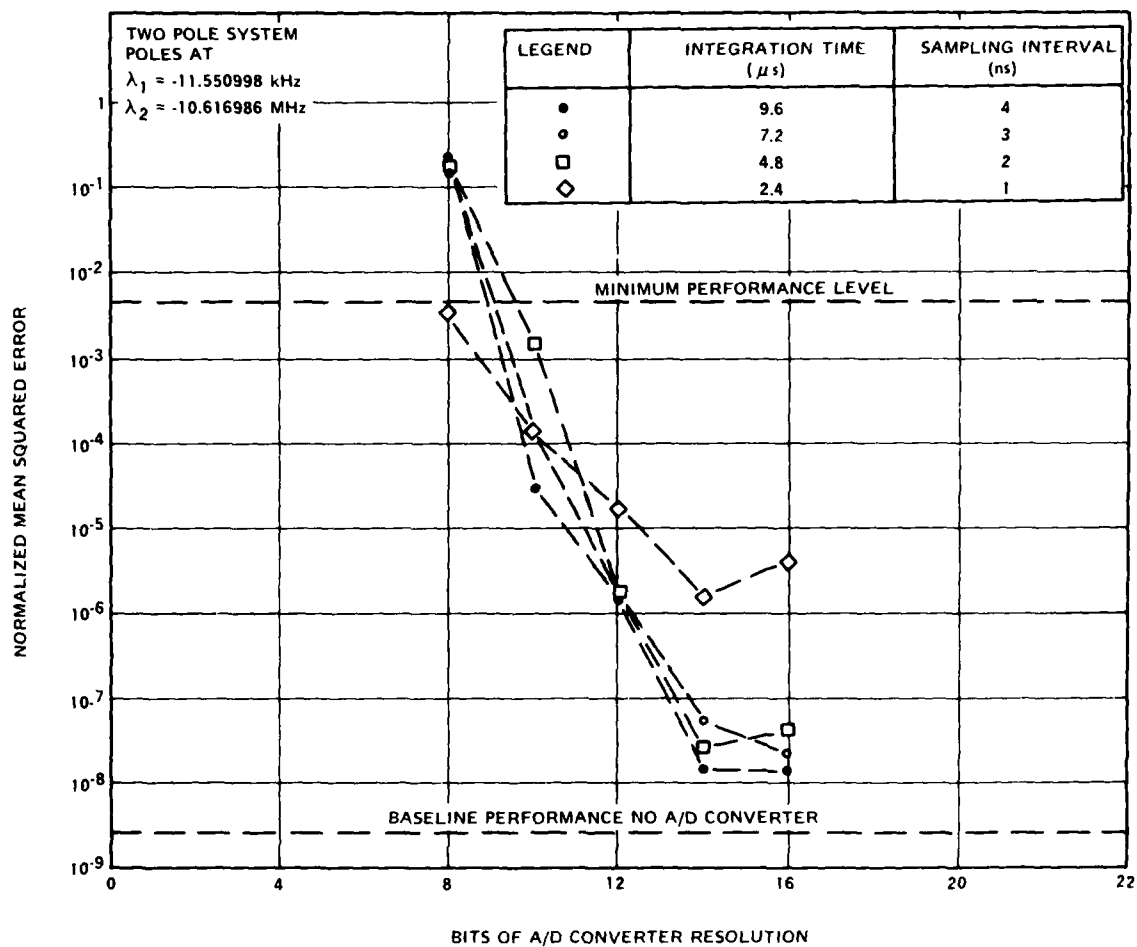


Figure 12. Results of A/D Converter Resolution
Simulation Runs for Two-Pole System

TABLE 7. FOUR-POLE SYSTEM, PERFECT A/D CONVERSION

System Poles (MHz)			System Residues			
Actual	Predicted	Percentage Error	Actual	Predicted	Percentage Error	Normalized Mean Squared Error
0.011550998	0.011279618	-2.35	2.8069192 x 10 ⁵	2.789548 x 10 ⁵	-0.619	0.103 x 10 ⁻⁴
0.510	0.4988702	-2.18	-1.20 x 10 ⁷	-1.048074 x 10 ⁷	-12.66	
0.82	0.8540037	-4.14	1.51 x 10 ⁷	1.3534398 x 10 ⁷	-10.36	
6.5	6.453757	-0.71	-1.61 x 10 ⁸	-1.594465 x 10 ⁸	-0.964	

computational load required to form the inner products for the Gram matrix. Each integration of input and output using the Simpson's rule integration technique results in a reduction of the number of samples that can be used for the next integration. This is illustrated below.

Consider the output samples $y_1(0)$, $y_1(T)$, $y_1(2T)$, ..., $y_1(2nT)$, where nT is the n th sample and T is the sampling interval. The integral of $y_1(t)$, $y_2(t)$, as obtained using Simpson's rule, is given by the samples

$$y_2(0), y_2(2T), y_2(4T), \dots, y_2(2nT).$$

It is noted that there are only nT samples of $y_2(t)$ whereas there were $2nT$ samples of $y_1(t)$. As this output is successively integrated, the time distance between samples increases and the numerical accuracy of the integration technique is expected to decrease. This will have an adverse effect on the performance of the identification technique and is a contributor to the difference in performance achieved for a two-pole system and four-pole system.

A second potential reason for the performance degradation of the identification technique is the wide-band nature of the selected system. Previous studies (Reference 6) have demonstrated that the technique should not be applied directly to wide-band systems. For these systems, the frequency range is divided into low, medium, and high ranges and the technique is applied to each range of the frequencies. The resultant transfer functions are then appropriately merged to form the total system transfer function. These are two explanations of the performance degradation experienced as system complexity increases. Part II of this study will consider this phenomenon in more detail.

The performance results of Table 7 were based on a perfect A/D conversion capability. The effect of A/D converter resolution on technique performance was investigated and the results are presented in Table 8.

The results of Table 8 indicate that satisfactory performance of the identification technique is achieved for an A/D converter with 20 bits or greater resolution. The results indicate that the identification technique predicts system poles with acceptable accuracy for a A/D converter of 12-14 bits resolution. However, the predicted residues are significantly in error for A/D conversion with less than 20 bits of resolution.

TABLE 8. SIMULATION RESULTS FOR LEVELS OF A/D CONVERTER RESOLUTION,
FOUR-POLE SYSTEM, INTEGRATION TIME = 4.8 μ s, SAMPLING INTERVAL = 1.5 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
No A/D Converter	0.011279618	-2.35	2.789548×10^5	-0.619	0.103×10^{-4}
	0.4988702	-2.18	-1.048074×10^7	-12.66	
	0.8540037	4.14	1.3534398×10^7	-10.36	
	6.453757	-0.711	-1.594465×10^8	-0.964	
24	0.011284216	-2.3	2.791535×10^5	-0.548	0.108×10^{-5}
	0.49891	-2.17	-1.062×10^7	-11.5	
	0.853884	4.13	1.39232×10^7	-7.79	
	6.454	-0.707	-1.6059077×10^8	-0.254	
20	0.011199456	-3.04	2.8101495×10^5	0.115	0.2×10^{-2}
	0.498283	-2.297	-1.2463×10^7	3.86	
	0.855755	4.36	1.921179×10^7	27.2	
	6.4495	-0.776	-1.764755×10^8	9.6	
16	0.01050104	-9.1	3.337596×10^5	18.9	0.131×10^1
	0.492189	-3.49	-5.46427×10^7	355.0	
	0.877295	6.99	1.48004×10^8	880.0	
	6.40103	-1.52	-5.695889×10^8	253.0	
14	0.01269649	5.35	4.7975×10^5	70.9	0.13×10^2
	0.4992365	-2.11	-1.634088×10^8	1261.0	
	0.86406	5.37	4.474×10^8	2862.0	
	6.455498	0.684	-1.4533×10^9	802.0	
12	0.01381562	19.6	1.6083188×10^7	5629.0	---
	0.519115	1.78	-1.480015×10^{10}	1.23×10^5	
	0.8355322	1.89	3.653195×10^{10}	2.41×10^5	
	6.473508	-0.407	-1.0109×10^{11}	6.27×10^4	

This reflects the impact of errors that result when the system output is numerically integrated four times.

These results suggest difficulty in achieving identification of systems with four or more poles because the maximum resolution of commercially available A/D converters is 16 bits.

d. A/D Conversion Time Requirements

In addition to resolution, another important A/D converter performance parameter is conversion time. The conversion time is defined as the time required for a complete measurement by an analog-to-digital converter. The conversion time defines the rate at which the input to the A/D converter can be sampled.

The sampling interval impacts the performance of the identification technique since numerical integration techniques are used to form the Gram matrix inner products. In order to determine the sampling interval required for satisfactory performance of the identification technique, the sampling rate was varied using the computer simulation. A summary of results is presented in Tables 9 through 13. Table 9 presents the results for a perfect A/D converter and Tables 10 to 13 present results for 16, 14, 12, and 10 bit A/D converters, respectively. These results are plotted in Figure 13.

These results indicate that the sampling interval should be on the order of 8 nanoseconds for the system output with an upper break frequency of 10.62 MHz. It is also observed that this interval can be increased to 24 nanoseconds with a slight degradation in performance. This implies that the sampling rate is between 41.6 and 125 MHz for a 10.6 MHz signal. If the Nyquist rate is defined as being twice the highest break frequency of the output function, or 21.2 MHz, the recommended sampling interval for the two-pole system investigated is approximately 2 to 6 times the Nyquist rate.

These conclusions could change as more complex ($N > 2$) systems are taken into consideration. This is due to the "shrinking number of samples" characteristic of the Simpson's rule of integration. It is necessary to sample the output of the system under test at a rate sufficient to generate enough samples for the final integration.

It is clear from these results that the sampling rate required is driven by the need for accuracy of numerical integration and not by Nyquist sampling constraints.

TABLE 9. SAMPLING INTERVAL RESULTS FOR PERFECT A/D CONVERSION,
INTEGRATION TIME = 9.6 μ S

Sampling Interval (ns)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz	Mean Squared Error
4	0.011545154 10.615088	-0.05 -0.01787	2.8063116E5 -2.7362615E8	-0.02 -0.021	0.011	0.411 x 10 ⁻⁸
6	0.011548873 10.589395	-0.0184 -0.26	2.8067558E5 -2.7296621E8	-0.0058 -0.26	0.022	0.457 x 10 ⁻⁶
8	0.011522691 10.581431	-0.245 -0.335	2.8040241E5 -2.7271069E8	-0.103 -0.355	0.0577	0.88 x 10 ⁻⁶
12	0.011501365 10.501581	-0.429 -1.087	2.8019682E5 -2.705512E8	-0.176 -1.145	0.1136	0.91 x 10 ⁻⁵
16	0.011498709 10.458732	-0.452 -1.49	2.8017916E5 -2.6924843E8	-0.182 -1.62	0.138	0.19 x 10 ⁻⁴
24	0.011620742 10.540133	-0.604 -0.72	2.8142850E5 -2.7047914E8	-0.262 -1.17	0.124	0.285 x 10 ⁻⁴

TABLE 10. SAMPLING INTERVAL RESULTS FOR 16-BIT A/D CONVERTER,
INTEGRATION TIME = 9.6 μ s

Sampling Interval (ns)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz	Mean Squared Error
4	0.011537262 10.621939	-0.119 0.0466	2.8059592E5 -2.7381345E8	-0.0342 0.04715	0.029	0.147×10^{-7}
6	0.011537338 10.600888	-0.1182 -0.1516	2.8056244E5 -2.7325026E8	-0.04613 0.1586	0.0279	0.172×10^{-6}
8	0.011503870 10.593911	-0.408 -0.217	2.8028340E5 -2.7304170E8	-0.1455 -0.2348	0.0978	0.399×10^{-6}
12	0.011507740 10.493012	-0.3744 -1.167	2.8026737E5 -2.7034192E8	-0.1512 -1.221	0.105	0.103×10^{-9}
16	0.011502428 10.447852	-0.42 -1.593	2.8027978E5 -2.6900207E8	-0.1468 -1.71	0.154	0.211×10^{-4}
24	0.011622867 10.535917	0.622 -0.763	2.8145425E5 -2.7037742E8	0.2715 -1.208	0.1199	0.289×10^{-4}

TABLE 11. SAMPLING INTERVAL RESULTS FOR 14-BIT A/D CONVERTER,
INTEGRATION TIME = 9.6 μ s

Sampling Interval (ns)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz	Mean Squared Error
4	0.011573082	0.191	2.8108237E5	0.139	0.0426	0.173×10^{-5}
	10.560624	-0.5308	-2.7233161E8	-0.494		
6	0.011551605	-0.00522	2.8087744E5	-0.0661	0.0385	0.139×10^{-5}
	10.566747	-0.473	-2.7246056E8	-0.447		
8	0.011522615	-0.2457	2.8066333E5	-1.018	0.0767	0.269×10^{-5}
	10.547927	-0.6504	-2.7195890E8	-0.6305		
12	0.011490437	-0.524	2.8030790E5	-0.1368	0.1477	0.103×10^{-4}
	10.488631	-1.2089	-2.7030774E8	-1.233		
16	0.011485884	-0.5637	2.8021712E5	-0.169	0.1922	0.169×10^{-4}
	10.466335	-1.42	-2.6949992E8	-1.53		
24	0.011609131	0.5032	2.8166121E5	0.345	0.1726	0.258×10^{-4}
	10.527826	-0.8397	-2.7029216E8	-1.239		

TABLE 12. SAMPLING INTERVAL RESULTS FOR 12-BIT A/D CONVERTER,
INTEGRATION TIME = 9.6 μ s

Sampling Interval (ns)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz	Mean Squared Error
4	0.01156579	0.128	2.8179798E5	0.394	0.059	0.185×10^{-5}
	10.575364	-0.392	-2.7294636E8	-0.2697		
6	0.011525954	-0.2168	2.8187778E5	0.422	0.1679	0.311×10^{-5}
	10.587743	-0.2754	-2.7342334E8	-0.095		
8	0.011539311	-0.1011	2.8081213E5	0.0428	0.057	0.105×10^{-5}
	10.574969	-0.3957	-2.7259698E8	-0.397		
12	0.011529217	-0.1885	2.8103355E5	0.1217	0.123	0.611×10^{-5}
	10.515567	-0.955	-2.7106579E8	-0.9568		
16	0.011496888	-0.468	2.8060646E5	-0.0304	0.211	0.22×10^{-4}
	10.440006	-1.667	-2.6884574E8	-1.768		
24	0.011619679	0.594	2.8208909E5	0.4977	0.19	0.3×10^{-4}
	10.490084	-1.195	-2.6939983E8	-1.565		

TABLE 13. SAMPLING INTERVAL RESULTS FOR 10-BIT A/D CONVERTER,
INTEGRATION TIME = 9.6 μ s

Sampling Interval (ns)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Maximum Error in $ H(s) ^2$ (dB) for 1 kHz to 100 MHz	Mean Squared Error
4	0.010017392 10.355187	-13.27 -2.46	2.7057878E5 -2.678456E8	-3.6 -2.13	-1.06	0.49×10^{-4}
6	0.01001557 10.316076	-13.29 -2.83	2.7150760E5 -2.6727096E8	-3.27 -2.34	-1.09	0.7×10^{-4}
8	0.01000578 10.287152	-13.37 -3.1	2.7116091E5 -2.6632976E8	-3.39 -2.69	-1.09	0.754×10^{-4}
12	0.0099222242 10.219084	-14.1 -3.74	2.7140996E5 -2.6491260E8	-3.30 -3.20	-1.21	0.11×10^{-3}
16	0.0099410072 10.164943	-13.94 -4.257	2.7199783E5 -2.6339261E8	-3.097 -3.76	-1.19	0.133×10^{-3}
24	0.010148000 10.252004	-12.15 -3.438	2.7214522E5 -2.6415126E8	-3.045 -3.48	-0.915	0.89×10^{-4}

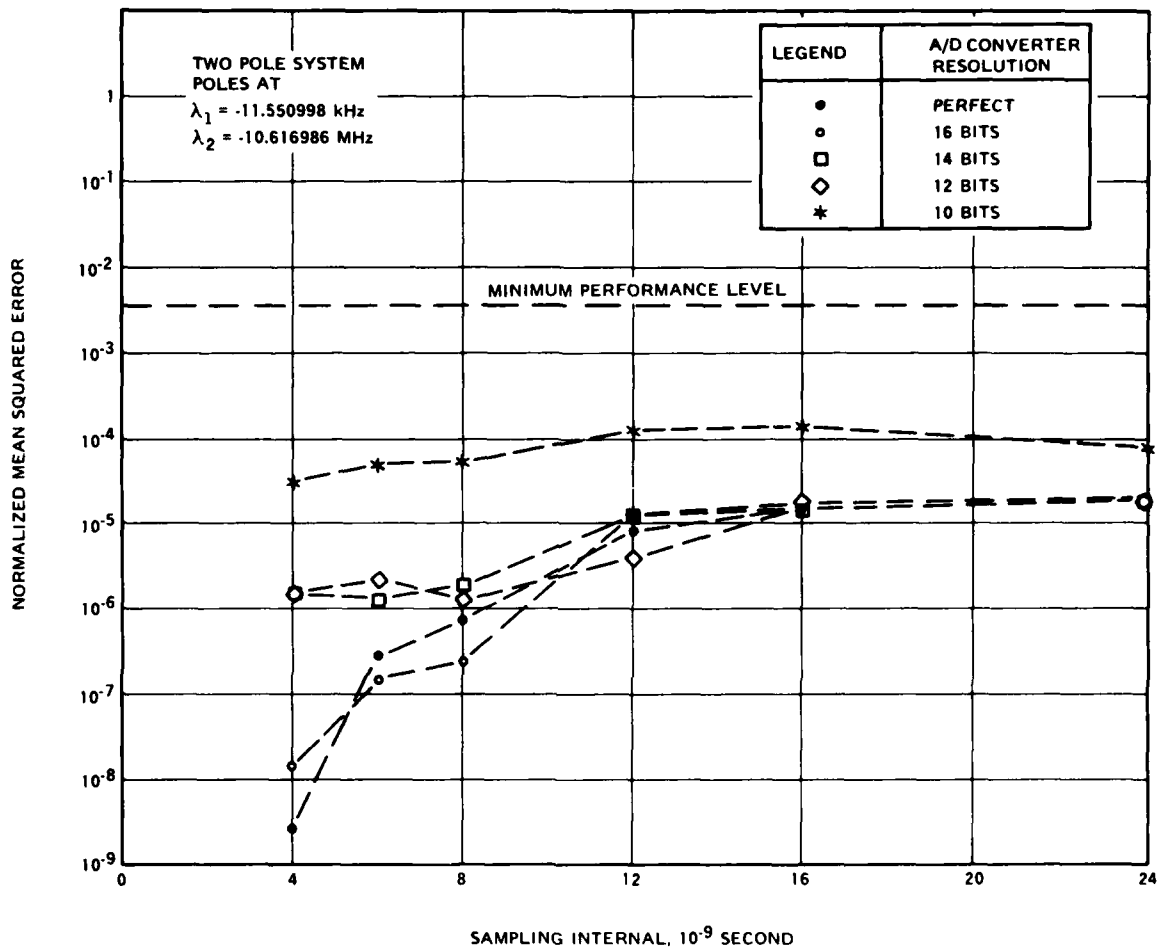


Figure 13. Results of Sampling Interval Rate Simulation for All A/D Converters

e. Second-Order Impulse Response Identification

The primary goal of the identification technique is to identify the second order impulse response of a nonlinear system. It has been shown (Reference 1) that the functional form of the second-order impulse response, $h_2(t_1, t_2)$ is given by equation (25).

The second-order response of a nonlinear system to an input given by

$$x(t) = \sum_{i=1}^L e^{-\alpha_i t} u(t) \quad (53)$$

has also been shown to be (Reference 1) given by

$$\begin{aligned} Y_2(s) = & \sum_{k_1=1}^M \sum_{k_2=1}^N \sum_{i=1}^L \sum_{j=1}^L A_{k_1 k_2} \\ & \cdot \frac{\alpha_i + \alpha_j + 2a_{k_1}}{(\alpha_j + a_{k_1})(\alpha_i + a_{k_1})(\alpha_i + \alpha_j + a_{k_1} + a_{k_2})} \frac{1}{s - (a_{k_1} + a_{k_2})} \\ & - \frac{1}{(\alpha_j + a_{k_1})(\alpha_i + a_{k_2})} \frac{1}{s + (\alpha_j - a_{k_2})} \\ & - \frac{1}{(\alpha_i + a_{k_1})(\alpha_j + a_{k_2})} \frac{1}{s + (\alpha_i - a_{k_2})} \\ & + \frac{\alpha_i + \alpha_j + 2a_{k_2}}{(\alpha_j + a_{k_2})(\alpha_i + a_{k_2})(\alpha_i + \alpha_j + a_{k_1} + a_{k_2})} \frac{1}{s + \alpha_i + \alpha_j} \end{aligned} \quad (54)$$

where

$$\begin{aligned} \alpha_j & \neq a_{k_1} & \text{for } j = 1, \dots, L; k_1 = 1, \dots, M \\ \alpha_i + \alpha_j + a_{k_1} + a_{k_2} & \neq 0 & \text{for } i, j = 1, \dots, L; k_1 = 1, \dots, M; \\ a_{k_1} & \neq a_{k_2} & \text{for } k_2 = 1, \dots, N; k_1 = N + 1, \dots, M. \end{aligned} \quad (55)$$

The quantities a_{k1} , a_{k2} , M and N were shown in Section II to be known from identification of the linear impulse response, $h_1(t)$. The remaining unknown quantities, A_{k1k2} are identifiable from the residues of the second-order response. These residues are given by the equation

$$R = C^{-1}Y \quad (56)$$

where

$$R = \text{residue matrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_N \end{bmatrix} \quad (57)$$

$$Y = \text{output matrix} = \begin{bmatrix} \bar{Y}_2(T) \\ \bar{Y}_3(T) \\ \bar{Y}_4(T) \\ \vdots \\ \bar{Y}_{N+1}(T) \end{bmatrix} \quad (58)$$

where

$\bar{y}_i = (i - 1)\text{th}$ integral of $y(t)$

$C = N \times N$ matrix whose $ij\text{th}$ element is defined by

$$C_{ij} = \frac{P_j(T)}{\lambda_j^i} - \sum_{m=1}^i \frac{x_{m+1}(T)}{(\lambda_j)^{i+1-m}} \quad (59)$$

where

$$P_j(T) = \int_0^T e^{\lambda_j \tau} x(\tau) d\tau \quad (60)$$

$x_i(T)$ = i^{th} integration of input $x(t)$ from $t = 0$ to $t = T$

$x_i(t) = \delta(t)$, unit impulse

The key impact of identifying the residues of $Y_2(s)$ on the implementation requirements is the need to accurately measure $y_2(T)$, $y_3(T)$, ..., $y_N(T)$, which are the integrated outputs of the nonlinear system. This is explained in detail below.

The system is excited by a input consisting of a sum of decaying exponential functions. The amplitude is selected to excite the linear and second-order responses of the system under test. The system output is $y_1(t) + y_2(t)$ where $y_1(t)$ is the linear response and $y_2(t)$ is the second-order response. The identification technique operates on $y_2(t)$ so there is a need to isolate $y_2(t)$ from the total response $y_1(t) + y_2(t)$. However, since the linear impulse response of the system under test will have been identified previously, the function $y_1(t)$ is known. Therefore $y_2(t)$ can be isolated from $y_1(t) + y_2(t)$. A potential problem arises because $y_2(t)$ is typically small in magnitude compared to $y_1(t)$. The implementation equipment will measure $y_1(t) + y_2(t)$ to the prescribed resolution of the A/D converter but the resolution of $y_2(t)$ will be lower because the most significant bit of the A/D will be assigned on the basis of the peak magnitude of the input $y_1(t) + y_2(t)$.

The purpose of this part of the investigation is to determine the A/D converter resolution characteristics required to accurately identify the residues of $Y_2(s)$ and subsequently the $A_{k_1 k_2}$. The second-order response of the nonlinear system to be used in this study is given by

$$y_2(t) = \left(\begin{aligned} &2.575137 \times 10^{-3} e^{-0.11550998 (2 \pi \times 10^6)t} \\ &- 1.5725176 \times 10^4 e^{-10.616986 (2 \pi \times 10^6)t} \\ &+ 2.3537485 e^{-0.023101996 (2 \pi \times 10^6)t} \\ &- 3.323955 \times 10^3 e^{-21.233972 (2 \pi \times 10^6)t} \\ &- 1.3645 e^{-1.603100 (2 \pi \times 10^6)t} \\ &+ 1.9755175 \times 10^4 e^{-12.208535 (2 \pi \times 10^6)t} \\ &- 9.051956 \times 10^2 e^{-3.1830988 (2 \pi \times 10^6)t} \end{aligned} \right) u(t)$$

(61)

where the input is

$$x(t) = 10^{-2} e^{-10^7 t} u(t). \quad (62)$$

This corresponds to an approximate representation of the second-order response of an amplifier circuit whose linear impulse response is given by (Reference 1)

$$h_1(t) = \left(2.8069192 \times 10^5 e^{-0.011550998 (2\pi \times 10^6)t} - 2.7368441 \times 10^8 e^{-10.616986 (2\pi \times 10^6)t} \right) u(t) \quad (63)$$

This is the two-pole system being considered in detail during the study. The expression for the second-order response $y_2(t)$ is approximate because the actual response would have natural frequencies at

$$\begin{aligned} s_1 &= -10.616986 (2\pi \times 10^6) \\ s_2 &= -10.628537 (2\pi \times 10^6) \end{aligned} \quad (64)$$

These poles arise from the two poles of the linear impulse response $[(s_1 = 0.011550998(2\pi \times 10^6) \text{ and } s_2 = 10.616986 (2\pi \times 10^6)]$ according to the relation

$$\begin{aligned} s_1 &= \lambda_2 \\ s_2 &= \lambda_1 + \lambda_2 \end{aligned} \quad (65)$$

Previous analyses (Reference 1) have shown that these poles will cause computational problems in evaluating the residues due to matrix inversion. In order to avoid this problem at this point of the study, $y_2(t)$ was assumed to have a single pole at $s_1 = \lambda_2$ with a residue given by the sum of the residues of λ_1 and $\lambda_1 + \lambda_2$ given in Reference 1. The problem of poles of the form $\lambda_2 + \lambda_1 = \lambda_2$ will be addressed in Part II of this study.

This second-order response was used in the computer simulation of the identification technique. The results are shown in Table 14. The most significant bit of the

TABLE 14. SECOND-ORDER RESPONSE RESULTS FOR A/D CONVERTERS, INTEGRATION
TIME = 3.2 μ s, SAMPLING INTERVAL = 1 ns, MSB SET FOR $y_1(t)$

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
No A/D Converter	0.011545154	-0.0506	0.0577	0.2×10^4	0.294×10^{-6}
	10.615088	-0.0179	-1.5745944×10^4	0.132	
	0.02309028	-0.0506	2.3011404	-2.23	
	21.230176	-0.0179	-3.33472×10^3	0.324	
	1.6030958	-0.00036	-1.455635	6.68	
	12.206638	-0.0155	1.978457×10^4	0.149	
	3.1830988	0	-9.047332×10^2	-0.051	
16	0.011537262	-0.1189	-0.378	-0.148×10^5	0.694×10^{-5}
	10.621939	0.0466	-1.593702×10^4	1.34716	
	0.02307452	-0.1189	2.8255933×10^3	20.05	
	21.243878	0.0466	-3.4012×10^3	2.32	
	1.60308669	-0.000856	-2.820757	106.72	
	12.213488	0.0406	2.00318×10^4	1.4	
	3.1830988	0	-9.003308×10^2	-0.537	
14	0.01157308	0.1911	5.56	0.2×10^6	0.207×10^{-3}
	10.5600624	-0.536	-1.4429517×10^4	-8.24	
	0.02314616	0.1911	-3.68104	256	
	21.120125	-0.536	-2.8700895×10^3	13.7	
	1.60312251	0.00138	6.4197	-570.	
	12.1516118	-0.466	1.8074497×10^4	-8.5	
	3.1830988	0	-9.3215×10^2	-2.9	

TABLE 14. SECOND-ORDER RESPONSE RESULTS FOR A/D CONVERTERS, INTEGRATION
TIME = 3.2 μ s, SAMPLING INTERVAL = 1 ns, MSB SET FOR $y_1(t)$ (Continued)

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
12	0.011565797	0.128	48.29	1.82×10^7	0.931×10^{-2}
	10.575364	-0.392	-7.29475×10^3	-53.6	
	0.02313159	0.128	-4.9039826×10^1	-2.18×10^3	
	21.150728	-0.392	-3.602589×10^2	-89.2	
	1.603115228	0.00092	5.989077×10^1	4.49×10^3	
	12.166913	-0.34	8.851946×10^4	-55.19	
	3.1830988	0	-1.132547×10^3	25.11	
10	0.010017392	-13.27	12.1	4.7×10^5	0.0242
	10.355187	-2.465	3.967×10^3	-125.2	
	0.020034784	-13.27	-21.20	-1000.	
	20.710374	-2.465	2.559803×10^3	-177.	
	1.60156682	-0.095	2.63355×10^2	-1.94×10^4	
	10.365204	-2.1	-4.729866×10^3	-123.9	
	3.1830988	0	-1.736826×10^3	91.87	
8	0.0048517152	-58.	-0.662	-2.58×10^4	0.254
	10.942193	3.06	-5.3947137×10^4	243.0	
	0.00970342	-58.	-6.08179	-358.4	
	21.884386	3.06	-1.7776168×10^4	434.8	
	1.59640114	-0.418	1.6418216×10^3	-1.21×10^4	
	12.533742	2.66	7.049036×10^4	256.8	
	3.1830998	0	-7.8×10^2	-13.8	

A/D converter was set by the peak level of the system total response, $y_1(t) + y_2(t)$. A plot of normalized mean squared error as A/D converter resolution is given in Figure 14.

These results indicate that 14 to 16 bits of resolution is required to achieve a reasonable level of performance.

Another set of simulation runs was made assuming that $y_2(t)$ could be isolated from the total response $y_1(t) + y_2(t)$ prior to A/D conversion. This makes it possible to set the most significant bit of the A/D converter based on the peak magnitude of $y_2(t)$. These results are presented in Table 15. A plot of normalized mean squared error versus A/D converter resolution is provided in Figure 15.

These results indicate that 10 to 12 bits of resolution are sufficient if $y_1(t)$ can be removed from the total response $y_1(t) + y_2(t)$ before A/D conversion. This will be, at best, difficult to achieve. This issue of separation of responses will be addressed in more detail in Part II of this study.

f. Additional A/D Converter Requirements

Previous paragraphs have concentrated on the A/D converter parameters of resolution and conversion time. (Cost considerations are provided in paragraph g. below). Conversion time and resolution are the key parameters because they place the most restrictions on the systems to which the identification technique can be applied. However, numerous other A/D converter characteristics must be taken into account when specifying an A/D converter. These include maximum rate of change of input, and absolute accuracy, among others.

The slew rate is an indication of the maximum rate of change of the input that the A/D converter can tolerate and still respond to individually important samples of the input. It is given by (Reference 7).

$$\left. \frac{dV}{dt} \right|_{\max} = 2^{-N} V_{FS} / T_{\text{convert}} \quad (66)$$

where N is the number of A/D converter bits

V_{FS} is the full scale input voltage

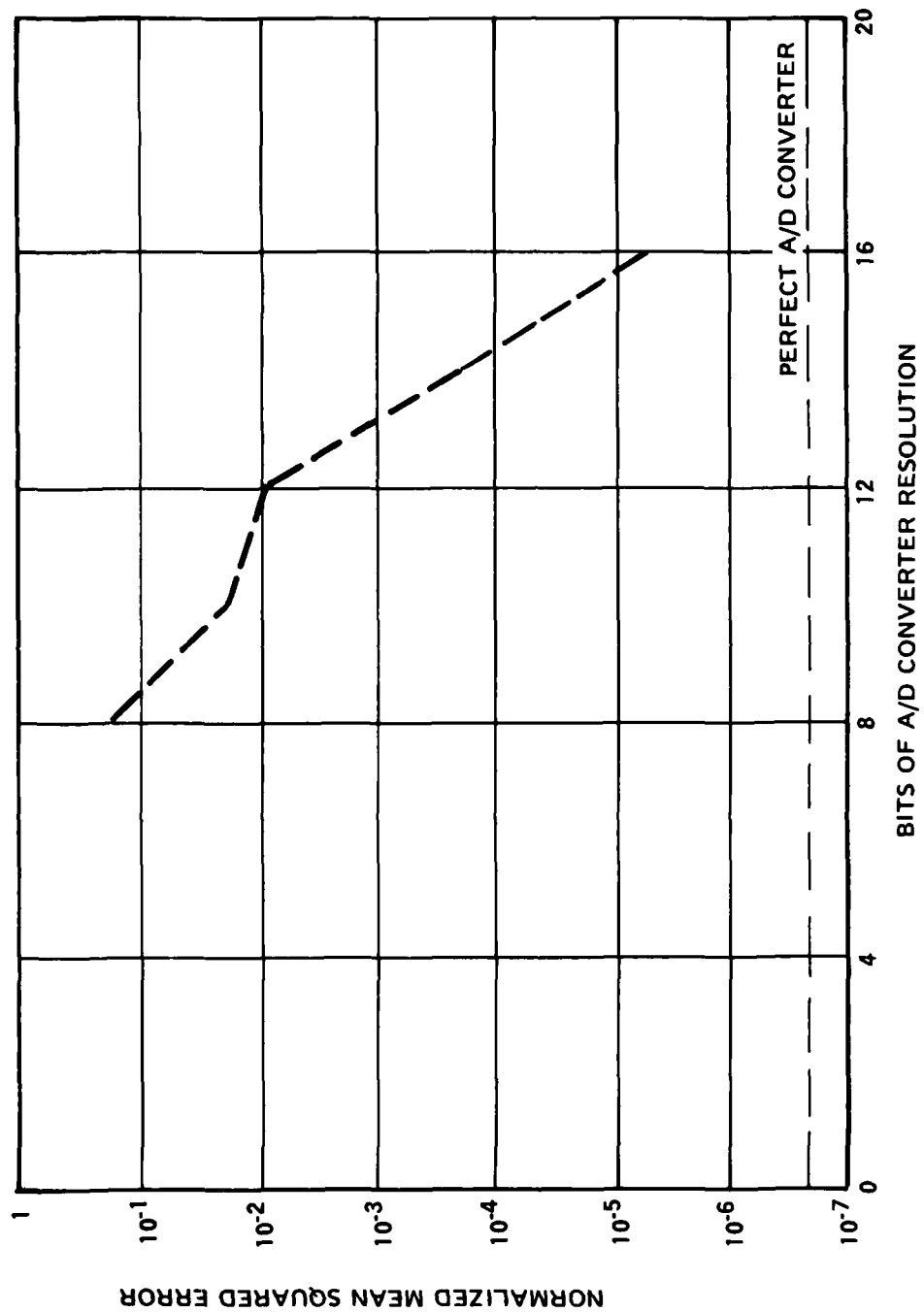


Figure 14. Second-Order System Response, Integration Time = $3.2 \mu s$,
Sampling Interval = $1 ns$, MSB Set for $y_1(t) = y_2(t)$

TABLE 15. SECOND-ORDER RESPONSE RESULTS FOR A/D CONVERTERS, INTEGRATION
TIME = 3.2 μ s, SAMPLING INTERVAL = 1 ns, MSB SET FOR $y_2(t)$

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
No A/D Converter	0.011545154	-0.0506	0.0577	2.0×10^3	0.294×10^{-6}
	10.615088	-0.0179	-1.5745944×10^4	0.132	
	0.02309028	-0.0506	2.3011404	-2.23	
	21.230176	-0.0179	-3.33472×10^3	0.324	
	1.6030958	-0.00036	-1.455635	6.68	
	12.206638	-0.0155	1.978457×10^4	0.149	
	3.1830988	0	-9.047332×10^2	-0.051	
16	0.011537262	-0.1189	-0.01467389	469.8	0.6×10^{-6}
	10.621939	0.0466	-1.5775605×10^4	0.320689	
	0.02307452	-0.1189	2.3429884	-0.457	
	21.243878	0.0466	-3.344544038×10^3	0.6194	
	1.60308669	-0.000856	-1.4023495	2.774	
	12.213488	0.0406	1.98233284×10^4	0.345	
	3.1830988	0	-9.05040566×10^2	-0.0173	
14	0.01157308	0.1911	0.264	1.01×10^4	0.292×10^{-5}
	10.5600624	-0.536	-1.55101118×10^4	-1.37	
	0.02314616	0.1911	2.1085	-10.42	
	21.120125	-0.536	-3.2665676×10^3	-2.03	
	1.60312251	0.00138	-1.9131	40.2	
	12.1516118	-0.466	1.9476245×10^4	-1.412	
	3.1830988	0	-9.0221021×10^2	-0.33	

TABLE 15. SECOND-ORDER RESPONSE RESULTS FOR A/D CONVERTERS, INTEGRATION
TIME = 3.2 μ s, SAMPLING INTERVAL = 1 ns, MSB SET FOR $y_2(t)$ (Continued)

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
12	0.011565797	0.128	0.0599084	2.2×10^4	0.681×10^{-6}
	10.575364	-0.392	-1.5615083×10^4	-0.7	
	0.02313159	0.128	2.33371	-0.85	
	21.150728	-0.392	-3.291855×10^3	-0.96572	
	1.603115228	0.00092	-2.176462348	59.2	
	12.166913	-0.34	1.961258392×10^4	-0.72	
	3.1830988	0	-9.0163858×10^2	-0.393	
10	0.010017392	-13.27	7.2988	2.8×10^5	0.247×10^{-4}
	10.355187	-2.465	-1.703214×10^4	8.311	
	0.020034784	-13.27	2.48552×10^2	1.04×10^5	
	20.710374	-2.465	-3.5576623×10^4	7.03	
	1.60156682	-0.095	-8.938179×10^2	6.54×10^4	
	10.365204	-2.1	2.1137867×10^4	6.999	
	3.1830988	0	-1.2048383×10^2	-0.00867	
8	0.0048517152	-58.	-0.002157	-16.2	0.118×10^{-3}
	10.942193	3.06	-1.5334697×10^4	-2.48	
	0.00970342	-58.	1.9260158	-18.17	
	21.884386	3.06	-3.1433888×10^3	-5.43	
	1.59640114	-0.418	1.0950628×10^1	-902.	
	12.533742	2.66	1.9252994×10^4	-2.54	
	3.1830998	0	-9.5911069×10^2	5.95	

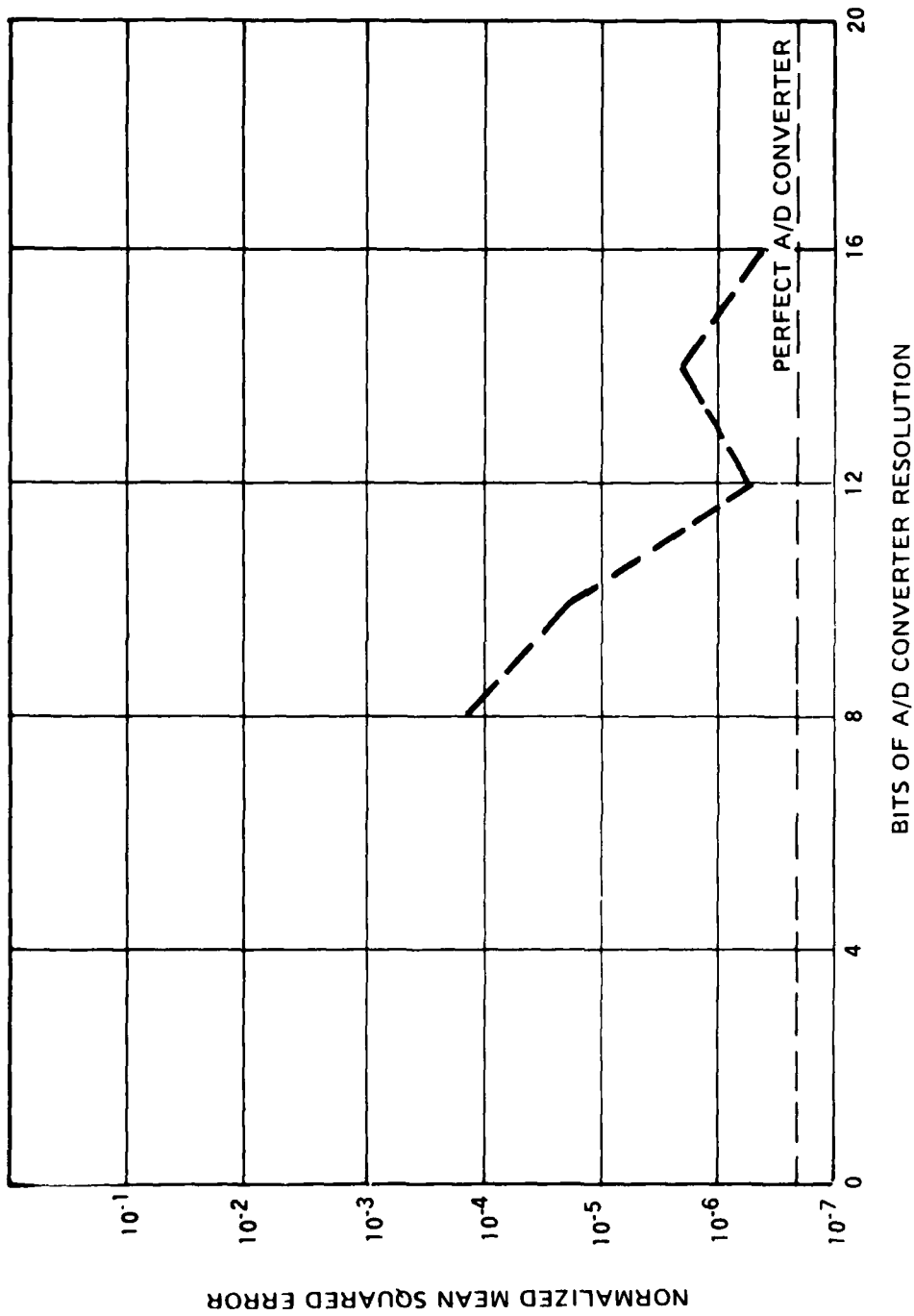


Figure 15. Second-Order System Response, Integration Time = 3.2 μ s,
Sampling Period = 1 ns, MSB Sr^+ for $y_2(t)$

T_{convert} is the conversion time

If the input signal changes at a rate faster than this maximum, 1 LSB changes in the input cannot be resolved within the sampling period. This problem can be alleviated somewhat by using a sample and hold circuit at the input to the A/D converter. Between conversions, the sample and hold acquires the input signal, and, just before conversion takes place, the signal is placed in hold, where it remains throughout the conversion.

If a sample and hold is used, the rate of change of the input is limited by

$$\left. \frac{dV}{dt} \right|_{\text{max}} = 2^{-N} V_{\text{FS}} / t_{\text{apu}} \quad (67)$$

where t_{apu} is the aperture time of the sample and hold. The aperture time of a typical sample and hold is on the order of 2 to 3 ns. This eases the problem described above. The signal conditioning using the sample and hold also tends to improve overall accuracy of the A/D conversion process.

The absolute accuracy error of an A/D converter is the difference between the analog input theoretically required to produce a given digital output code and the analog input actually required to produce that same code (Reference 7). Absolute accuracy error can be caused by several different sources of error. A good A/D converter will have an absolute accuracy of $\pm 1/2$ LSB. This implies that the performance of a 16 bit converter will lie somewhere between the performance predicted for a 16-bit converter and a 15-bit A/D converter.

Other A/D converter parameters, such as input full scale voltage and output code, are not as important as those discussed above. In general they need to be addressed for the particular system under consideration.

g. Survey of Currently Available A/D Converters

The objective of this phase of the study was to survey the characteristics of commercially available A/D converters to determine if the requirements of paragraphs B.2.c and d above can be met. The key characteristics of interest for this survey were the number of bits of resolution, the conversion time, and the cost. Many companies manufacture commercially available A/D converters, and, of course, these converters vary in performance and cost over a very broad range.

It is not warranted to present a detailed listing of all available A/D converters in this report. Therefore only the general characteristics of these converters will be listed and a selected few will be reviewed in detail. Data for this survey was obtained from A/D converter specification sheets obtained from the following manufacturers (listed in alphabetical order):

Analog Devices, Inc.
Analogic Corp.
Beckman Instruments, Inc.
Burr-Brown Research Corp.
Computer Labs, Inc.
Datel Systems, Inc.
DDC - ILC Data Device Corp.
Fairchild Semiconductor
Ferranti Electric
Hybrid Systems Inc.
Intech
Intersil, Inc.
Micro Networks Corp.
National Semiconductor
Precision Monolithics, Inc.
Teledyne Semiconductor
Texas Instruments, Inc.
TRW LSI Products
Zeltex, Inc.

The range of available A/D converters is illustrated in Figure 16, which is a plot of A/D converter resolution bits vs conversion time, for converters with less than 1 ms conversion time. Each X represents a device corresponding to a given resolution and conversion time manufactured by one of the companies listed above.

Figure 17 is a plot of minimum conversion speed for each level of resolution. The conversion time is converted to maximum input frequency in Figure 18. Figure 18 indicates that A/D converters of high resolution (14 to 16 bits) cannot accurately convert signals of frequency greater than 100 to 125 kHz. This imposes a significant restriction on the applicability of the identification technique since the results indicate that 14 to 16 bits of A/D converter resolution is generally needed for satisfactory performance. This frequency restriction is even more constrained (20 kHz) when sampling requirements are taken into account.

The general trend of Figure 17 is that the conversion time increases as the resolution increases. The current development trend seems to be directed toward the 8 to 12 bit resolution A/D converter. The 16-bit A/D converters are significantly more costly than a lower resolution converter

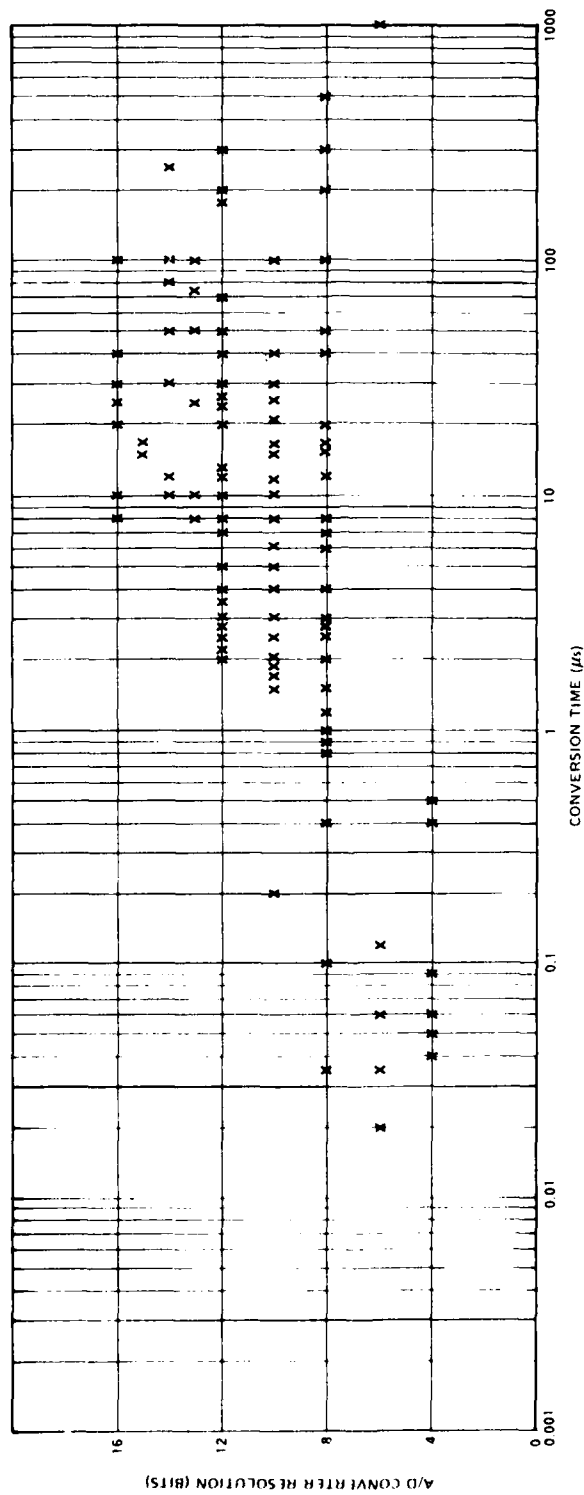


Figure 16. Available A/D Converter Characteristics

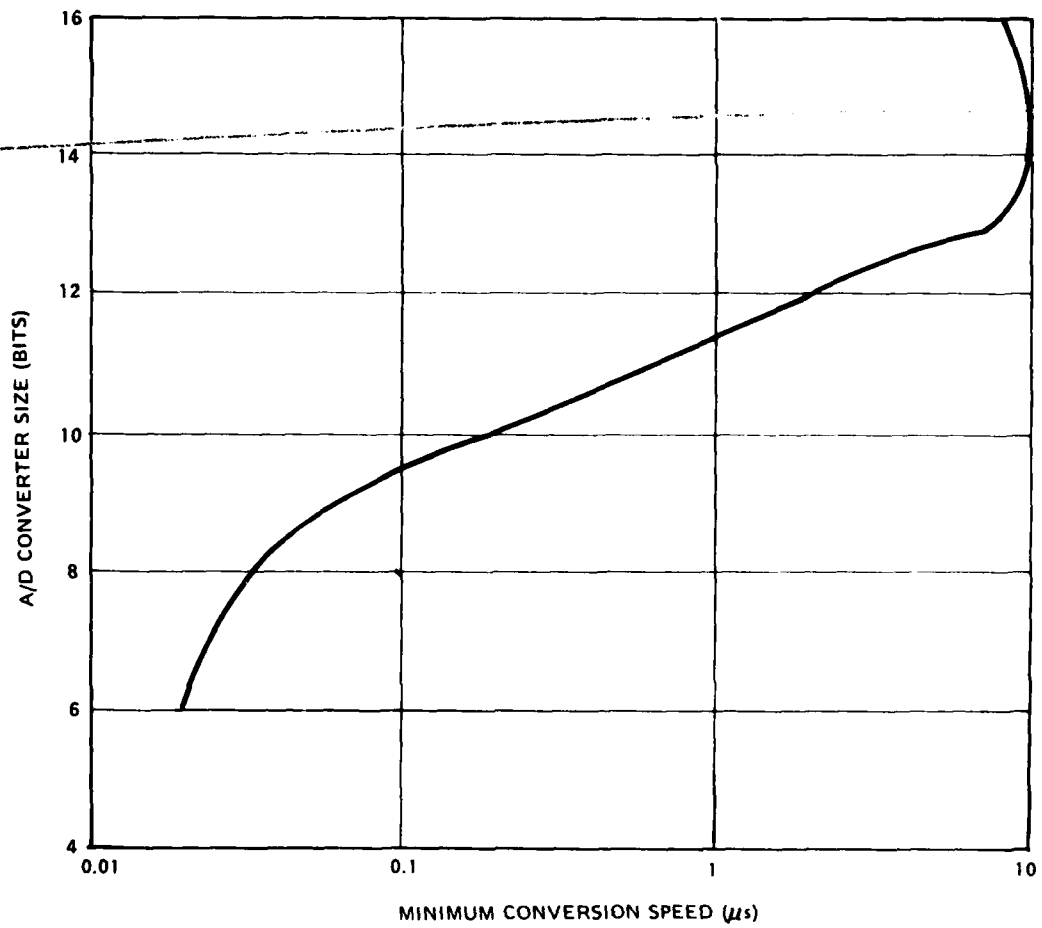


Figure 17. Current A/D Converter Conversion Speed Characteristics

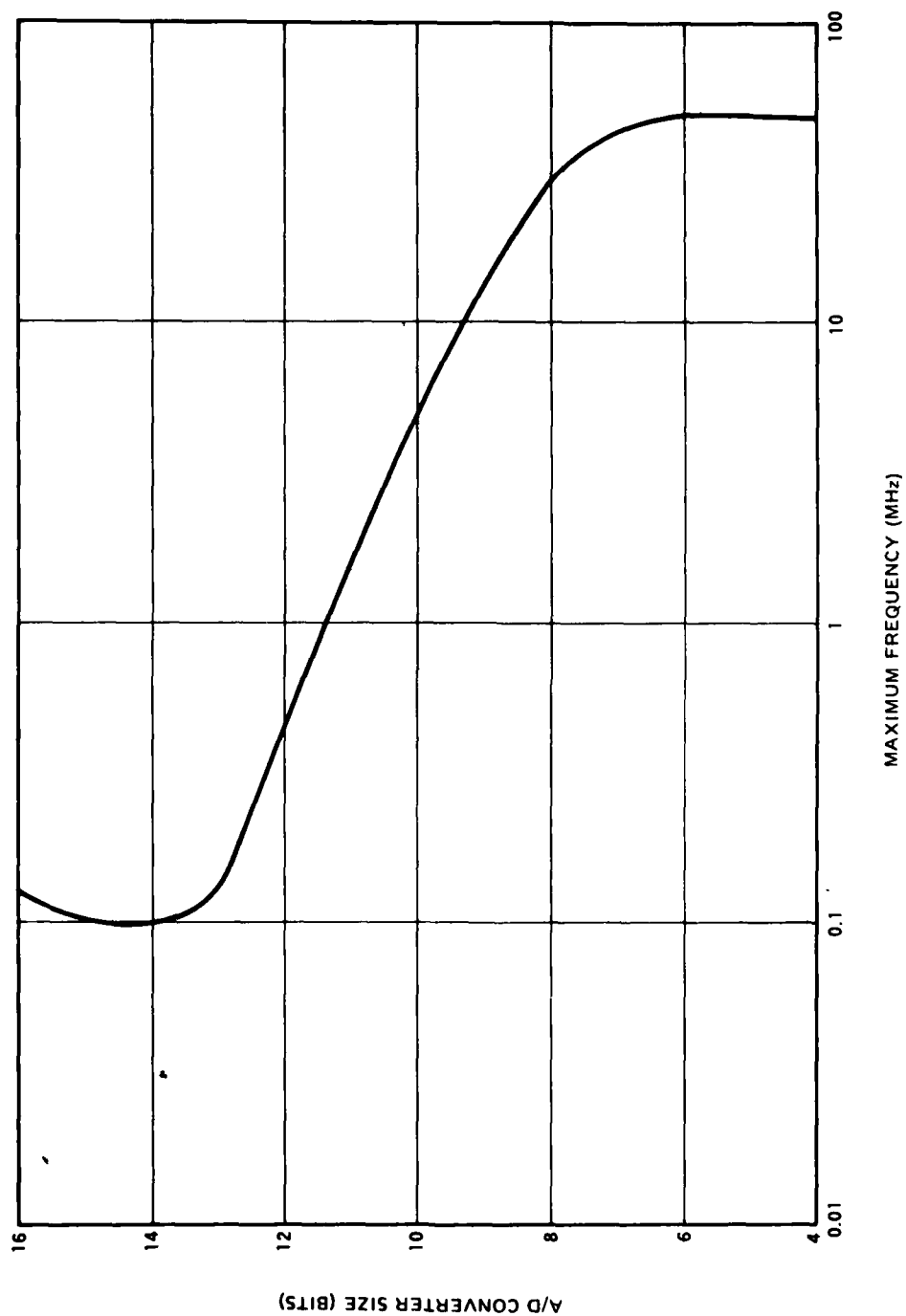


Figure 18. Current A/D Converter Frequency Characteristics

(8-12 bits) and not too many devices have appeared on the market. Commercially available 16-bit A/D converters are listed in Table 16.

TABLE 16. COMMERCIALY AVAILABLE A/D CONVERTERS

Conversion Time (ns)	Manufacturer	Cost (dollars)
8	Intech	1500.
10	Zeltex	750.
20	Zeltex	750.
25	Analogic	1395.
30	Analogic	895.
32	Zeltex	750.
40	Intech	460.
100	Micro Networks	220.
5,000	Analog Devices	1720.
100,000	Analogic	210.
200,000	Burr Brown	270.
250,000	Intersil	---

2. Amplifier Requirements

The digital implementation of the identification technique shown in Figure 6 indicates a pre-A/D converter amplifier. The purpose of this amplifier is to adjust the output of the system under test to the full-scale input level of the A/D converter. Most commercially available A/D converters have a normalized analog input signal level of ± 5 or ± 10 volts. In order to make complete use of the resolution capability of the A/D converter, it is necessary to adjust the level of the output of the system under test to this input level. For example, consider the two-pole system discussed in paragraph III.B.1.c.(2). This is a representation of the linear portion of an amplifier (Reference 1) and is valid only for low level input voltages (on the order of 1 millivolt). The peak system response to a 1

millivolt input is also on the order of millivolts. If a standard 16-bit A/D converter with a full-scale input voltage level of 10 volts were used in the digital implementation of the identification technique, the samples of the system response would be accurate to only about 5 bits of resolution.

This need for a pre-A/D converter amplifier complicates the identification process. This complication arises because the A/D converter samples the output of the cascaded nonlinear system and the amplifier. The identification technique processes these samples and will attempt to identify the total system, which consists of the system under test in series with the amplifier. This is illustrated in Figure 19.

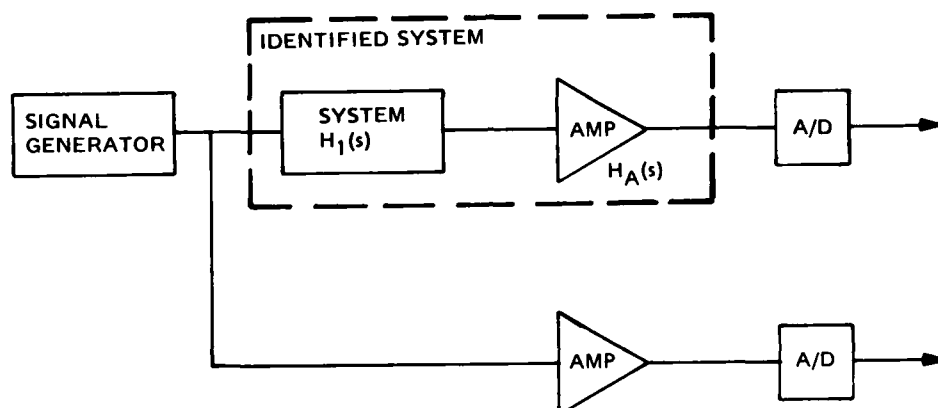


Figure 19. Identification Technique Model to Account for Amplifier

The identification technique attempts to identify the transfer function

$$H_T(s) = H_1(s) H_A(s) \quad (68)$$

where $H_1(s)$ is the transfer function of the system under test and $H_A(s)$ is the transfer function of the amplifier. The primary complication introduced by the amplifier is one of dimensionality. If $H_1(s)$ has two poles and $H_A(s)$ has two poles, the identification process will attempt to identify a four-pole system. Since $H_A(s)$ will be known, there is no difficulty in obtaining $H_1(s)$ from $H_T(s)$. It will be necessary to maintain the frequency extent of $H_A(s)$ approximately equal to that of $H_1(s)$ to avoid the wide-band problem discussed in paragraph III.B.1.c. However, the difference in the performance of the

identification technique for a two-pole system and a four-pole system was amply demonstrated in paragraph III.B.1.c.(2). These results suggest that the amplifier may unduly degrade the performance of the technique. It would therefore be advantageous to devise an approach to alleviate the complication introduced by the amplifier. Two approaches were considered during the study and these are discussed below. The first approach to solving the amplifier problem was to use an amplifier that is very wide-band compared to the system under test. This approach is based on the concept that the amplifier frequency response will not significantly affect the frequency response of interest. This is illustrated in Figure 20.

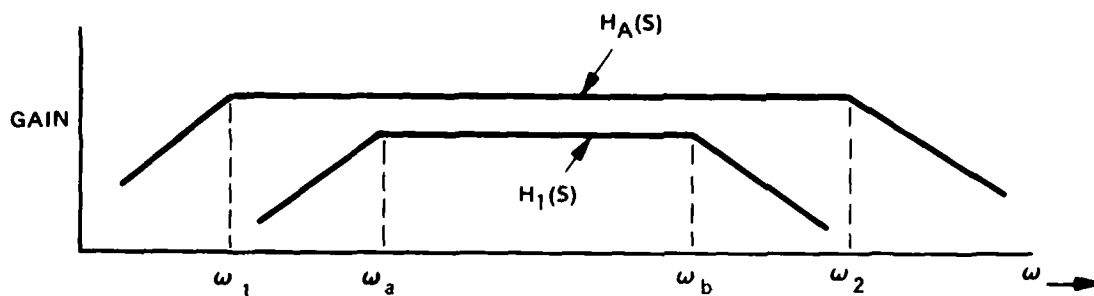


Figure 20. Frequency Extent Comparison for Test System and Amplifier

If $\omega_1 \ll \omega_a$ and $\omega_2 \gg \omega_b$, then

$$H_1(s) H_A(s) \doteq H_1(s) \quad (69)$$

Previous work on the pencil-of-functions technique (Reference 6) has demonstrated that a system can be excited such that only a limited region of its frequency extent significantly affects the output. In order to determine the bandwidth requirements of this amplifier, a set of simulation cases was run. The first amplifier considered has a transfer function given by

$$H_{A_1}(s) = \frac{100}{\left(1 + \frac{s}{\omega_1}\right)\left(1 + \frac{s}{\omega_2}\right)} \quad \omega_2 > \omega_1 \quad (70)$$

The Bode diagram of this transfer function is shown in Figure 21.

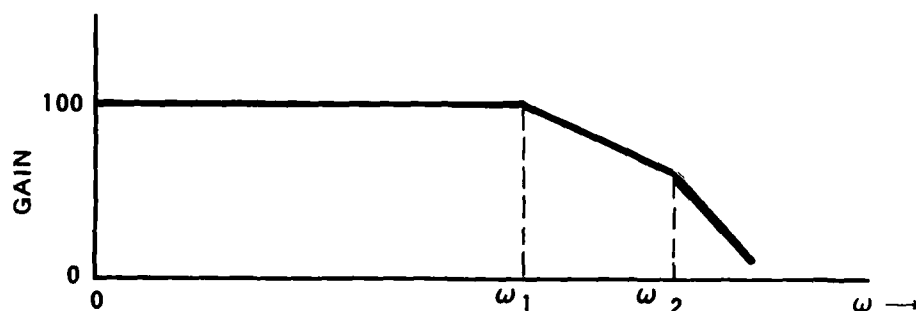


Figure 21. Bode Diagram of Amplifier with Transfer Function $H_{A_1}(s)$

This amplifier model was added to the computer simulation. The system to be identified was the two-pole system described in paragraph III.B.1.c.(1), whose transfer function is given by

$$H_1(s) = \frac{2.8069192 \times 10^5}{s + 0.011550998 (2\pi) (10^6)} - \frac{2.7368441 \times 10^8}{s + 10.616986 (2\pi) (10^6)} \quad (71)$$

The break frequencies of the amplifier, ω_1 and ω_2 , were varied and performance of the technique was evaluated. Since ω_1 determines the 3-dB bandwidth of the amplifier, it is the key parameter to be evaluated. The number of poles of the system to be identified was set to two in the simulation. Therefore, the identification process operates on the samples of the cascaded system (system under test and amplifier) and attempts to identify only two poles. The results of the simulation are shown in Table 17 for different values of ω_1 and ω_2 . The normalized mean squared error is plotted as a function of ω_1 in Figure 22.

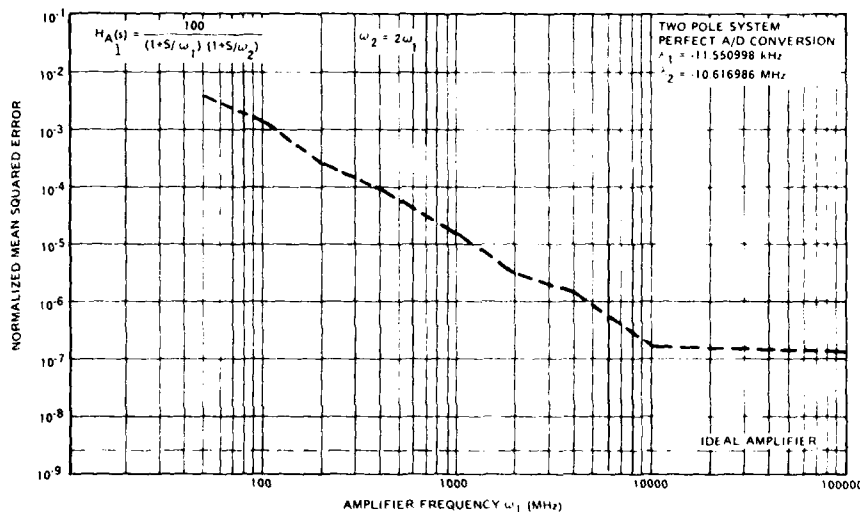


Figure 22. Normalized Mean Squared Error as a Function of ω_1

Figure 22 indicates that an upper break frequency of at least approximately 1000 times the upper break frequency of the system under test is necessary to minimize the effect of the amplifier on identification performance.

TABLE 17. IDENTIFICATION TECHNIQUE PERFORMANCE FOR DIFFERENT AMPLIFIER CONFIGURATIONS, PERFECT A/D CONVERSION, INTEGRATION
TIME = 9.6 μ s, SAMPLING INTERVAL = 4 ns

$$\text{AMPLIFIER TRANSFER FUNCTION} = \frac{1}{[1+(S/\omega_1)][1+S/\omega_2]}$$

Filter Poles (MHz) ω_1, ω_2	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
50	0.011696102	1.25	2.8233227×10^5	0.584	0.57×10^{-2}
100	8.0143939	-24.51	-2.0681498×10^8	-24.43	
100	0.011612318	0.53	2.8139416×10^5	0.25	0.155×10^{-2}
200	9.1377518	-13.93	-2.3570199×10^8	-13.88	
200	0.011576915	0.224	2.8099455×10^5	0.108	0.389×10^{-3}
400	9.8365571	-7.35	-2.537214×10^8	-7.29	
400	0.011563444	0.1077	2.808394×10^5	0.0525	0.96×10^{-4}
800	10.217587	-3.76	-2.6351158×10^8	-3.72	
800	0.01156185	0.0939	2.808174×10^5	0.0447	0.35×10^{-4}
1000	10.370997	-2.31	-2.6743464×10^8	-2.28	
1000	0.011560029	0.0782	2.807956×10^5	0.0369	0.167×10^{-4}
2000	10.447546	-1.596	-2.6938381×10^8	-1.57	
2000	0.01155798	0.06	2.8077123×10^5	0.0282	0.49×10^{-5}
4000	10.524276	-0.873	-2.7133375×10^8	-0.859	
4000	0.011550391	-0.0052	2.8068824×10^5	-0.0013	0.11×10^{-5}
8000	10.572337	-0.42	-2.7254702×10^8	-0.415	
8000	0.011554565	0.0145	2.8073282×10^5	0.031	0.708×10^{-6}
10000	10.581781	-0.33	-2.7279100×10^8	-0.33	
10000	0.011548797	-0.019	2.8067059×10^5	-0.0076	0.256×10^{-6}
20000	10.596042	-0.197	-2.7314685×10^8	-0.196	
100000	0.011554792	-0.033	2.8073499×10^5	0.0153	0.136×10^{-6}
200000	10.601552	-0.145	-2.7329259×10^8	-0.143	
∞	0.011545154	-0.05	2.8063116×10^5	-0.02	0.411×10^{-8}
∞	10.615088	-0.0187	-2.7362615×10^8	-0.021	

The low frequency characteristics required of the amplifier were investigated using an amplifier with a transfer function of the form

$$H_{A_2}(s) = \frac{K \left(1 + \frac{s}{\omega_1} \right)}{\left(1 + \frac{s}{\omega_2} \right) \left(1 + \frac{s}{\omega_3} \right)} \quad (72)$$

$$\omega_1 < \omega_2 < \omega_3$$

The Bode diagram of this transfer function is shown in Figure 23.

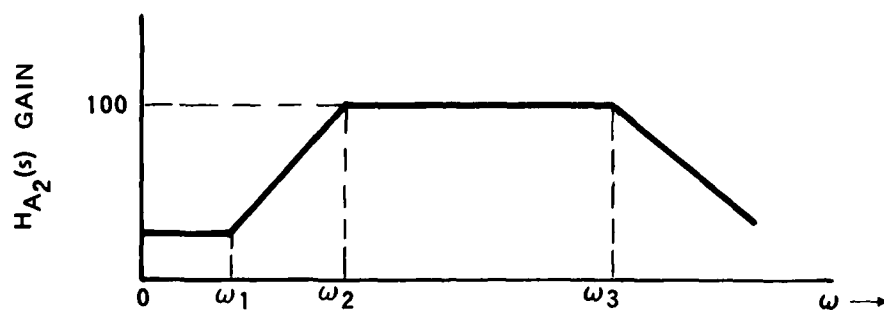


Figure 23. Bode Diagram of Amplifier with Transfer Function $H_{A_2}(s)$

The break frequencies of the amplifier were varied and the performance of the identification technique evaluated.

These results are summarized in Table 18. A plot of normalized mean squared error versus ω_2 for $\omega_1 = \omega_2/10$ and selected values of ω_3 is presented in Figure 24. These results indicate that the lower break frequency of the amplifier, ω_2 , should be, at least, approximately 1/100 of the lowest break frequency of the system under test to minimize the impact of the amplifier on the results.

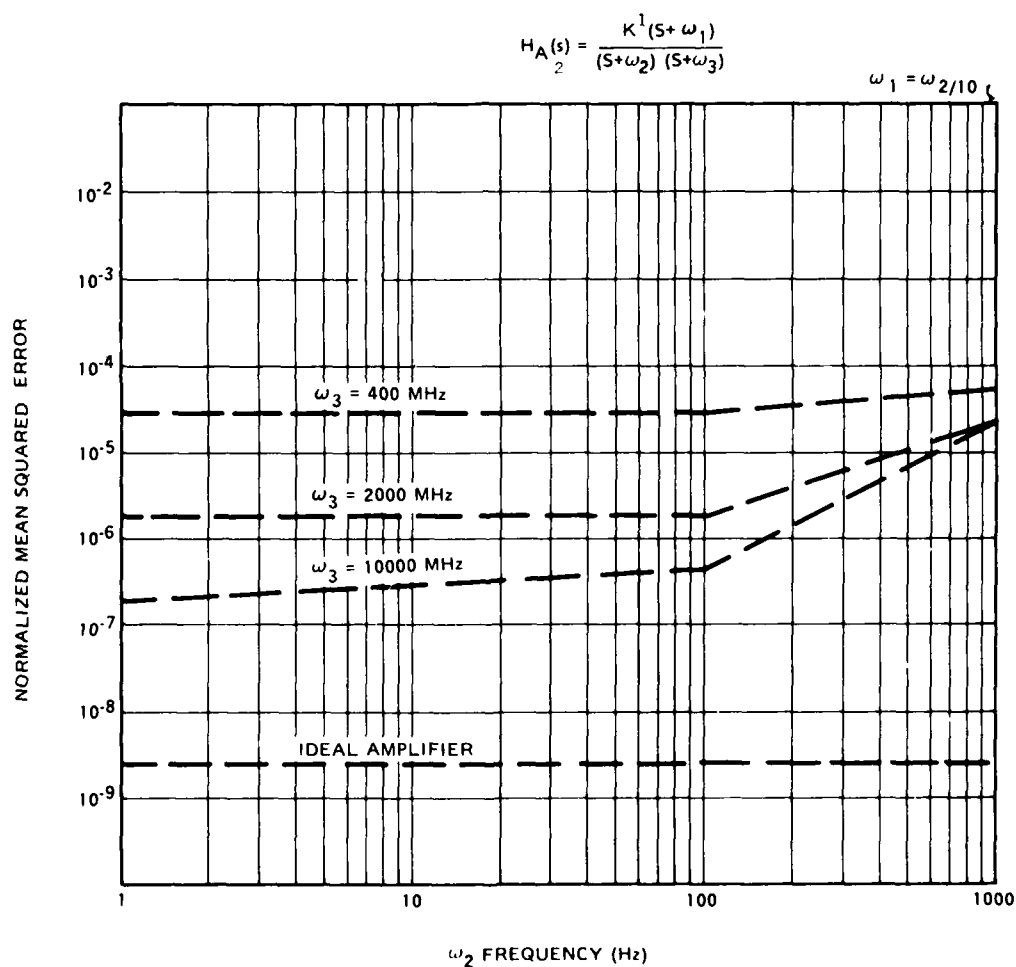


Figure 24. Identification Technique Performance as a Function of Amplifier Characteristic

AD-A081 464 GENERAL ELECTRIC CO UTICA N Y AIRCRAFT EQUIPMENT DIV F/O 9/2
NONLINEAR SYSTEM IDENTIFICATION STUDY, PART I. IMPLEMENTATION F--ETC(U)
DEC 79 E J EWEN F30602-79-C-0046

GENERAL ELECTRIC CO UTICA N Y AIRCRAFT EQUIPMENT DIV
NONLINEAR SYSTEM IDENTIFICATION STUDY, PART I, IMPL
DEC 79 E J EWEN F3060

F/O 9/2

MENTATION F--ETC(U)

F30602-79-C-0046

RADC -TR-79-199-PT-1

NL

2 of 2

ADH: 86.4

END
DATE
FILMED
4-80
DTIC

DTM

TABLE 18. IDENTIFICATION TECHNIQUE PERFORMANCE FOR DIFFERENT AMPLIFIER CONFIGURATIONS, PERFECT A/D CONVERSION, INTEGRATION
TIME = 9.6 μ s, SAMPLING INTERVAL = 4 ns

$$\text{Amplifier Transfer Function} = \frac{K[1+(S/\omega_1)]}{[1+(S/\omega_2)][1+(S/\omega_3)]}$$

Filter Frequencies (MHz)	Predicted System Poles	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
$\omega_1 = 0.0001$ $\omega_2 = 0.001$ $\omega_3 = 400$	0.011597292 10.340356	0.4 -2.6	3.0406262×10^5 -2.6666951×10^8	8.33 -2.56	0.78×10^{-4}
$\omega_1 = 0.00001$ $\omega_2 = 0.0001$ $\omega_3 = 400$	0.011554641 10.344142	0.03 -2.57	2.831267×10^5 -2.667433×10^8	0.867 -2.54	0.44×10^{-4}
$\omega_1 = 0.000001$ $\omega_2 = 0.00001$ $\omega_3 = 400$	0.011561775 10.341452	0.093 -2.59	2.8105734×10^5 -2.6667401×10^8	0.13 -2.56	0.45×10^{-4}
$\omega_1 = 0.0000001$ $\omega_2 = 0.000001$ $\omega_3 = 400$	0.011558284 10.345566	0.063 -2.56	2.8080413×10^5 -2.6677372×10^8	0.04 -2.52	0.436×10^{-4}
$\omega_1 = 0.0001$ $\omega_2 = 0.001$ $\omega_3 = 2000$	0.011589247 10.554968	0.33 -0.584	3.0396171×10^5 -2.7213314×10^8	8.29 -0.567	0.38×10^{-4}
$\omega_1 = 0.00001$ $\omega_2 = 0.0001$ $\omega_3 = 2000$	0.011548494 10.553439	-0.0217 -0.598	2.8305334×10^5 -2.7207385×10^8	0.841 -0.588	0.26×10^{-5}
$\omega_1 = 0.000001$ $\omega_2 = 0.00001$ $\omega_3 = 2000$	0.011555324 10.55293	0.0375 -0.603	2.8098128×10^5 -2.7206022×10^8	0.103 -0.593	0.234×10^{-5}
$\omega_1 = 0.0000001$ $\omega_2 = 0.000001$ $\omega_3 = 2000$	0.011552592 10.5555751	0.0138 -0.577	2.8073657×10^5 -2.7212768×10^8	0.0159 -0.569	0.215×10^{-5}
$\omega_1 = 0.0001$ $\omega_2 = 0.001$ $\omega_3 = 10000$	0.011591676 10.589241	0.352 -0.261	3.0398925×10^5 -2.7300537×10^8	8.3 0.248	0.372×10^{-4}
$\omega_1 = 0.00001$ $\omega_2 = 0.0001$ $\omega_3 = 10000$	0.011548645 10.59398	0.02 -0.217	2.8305346×10^5 -2.7310357×10^8	0.841 0.21	0.659×10^{-6}
$\omega_1 = 0.000001$ $\omega_2 = 0.00001$ $\omega_3 = 10000$	0.011558511 10.587531	0.065 -0.277	2.8101477×10^5 -2.7294168×10^8	0.115 -0.271	0.49×10^{-6}
$\omega_1 = 0.0000001$ $\omega_2 = 0.000001$ $\omega_3 = 10000$	0.011553427 10.595411	0.021 -0.2	2.8074415×10^5 -2.7313569×10^8	0.0186 -0.2	0.266×10^{-6}
$\omega_1 = 0$ $\omega_2 = 0$ $\omega_3 = 0$	0.011545154 10.615088	-0.05 -0.0187	2.8063116×10^5 -2.7362615×10^7	-0.02 -0.021	0.411×10^{-8}

This approach requires that the amplifier have a bandwidth approximately 1000 times that of the system under test. This is reasonable for test systems with bandwidths on the order of 10 to 50 kHz. However, this places severe bandwidth requirements on the amplifier for test systems with 1 MHz bandwidth or greater.

In view of these performance requirements for the amplifier, another approach was investigated.

The second approach to alleviating the amplifier problem employed linear system analysis techniques. Consider once again the digital implementation of Figure 6. The input to the A/D converter, in the Laplace domain, is given by

$$Y_O(s) = H_1(s)H_A(s)U(s) \quad (73)$$

From a linear system point of view, this is equivalent to a system with a transfer function, $H_1(s)$, being excited with an input of $H_A(s)U(s)$ or as shown in Figure 25.

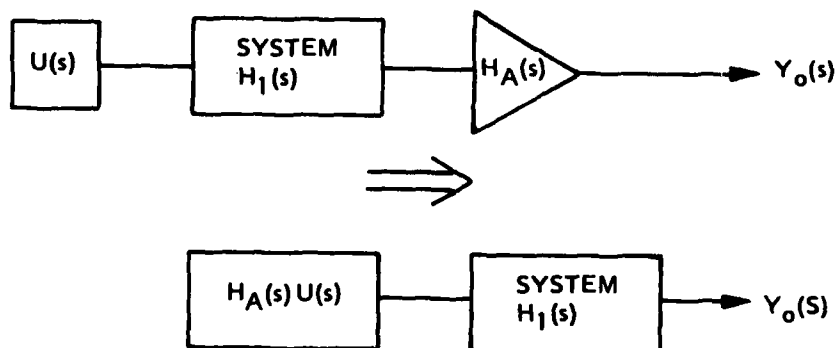


Figure 25. Linear System Equivalence Concept

This implies an equivalent technique implementation (for processing only) given by that configuration shown in Figure 26.

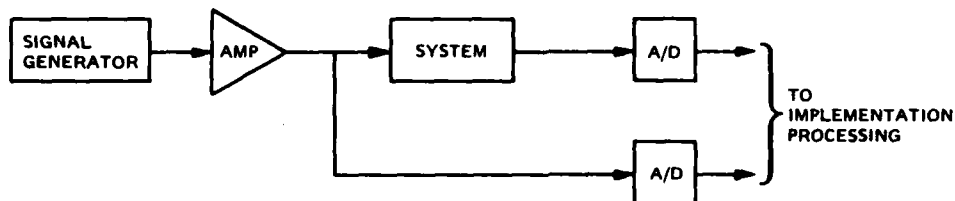


Figure 26. Equivalent Linear System Representation for Amplifier Analysis

This technique implementation was simulated for an amplifier whose transfer function was of the form

$$H_A(s) = \frac{K}{\left[1 + \left(\frac{s}{\omega_1}\right)\right] \left[1 + \frac{s}{\omega_2}\right]} \quad (74)$$

The results of the performance simulation are presented in Tables 19 and 20. Graphs of normalized mean squared error for these results are shown in Figures 27 and 28. These results suggest that best performance of the identification technique is obtained when the upper break frequency of the amplifier is approximately one to two times the upper break frequency of the system under test. The reason for this behavior is that, for the integration period needed to identify the low frequency break point of the system under test, the high frequency components of the system output generated by the high frequency components of the input $[H_A(s)U(s)]$ have a negligible effect on the inner products generated for the Gram determinant. Therefore, the amplifier must have a frequency response essentially matched in bandwidth to the system under test for this approach to yield satisfactory performance.

The results indicate that similar performance is obtained using either approach. Therefore, the approach used in an actual test setup will depend on the characteristics of the system under test.

The above analysis concentrated on the frequency response characteristics of the pre-A/D converter amplifier. These characteristics will have, as shown, a profound effect on the performance of the identification technique. There are, however, many other parameters that must be considered before selecting an amplifier. These include gain, slew rate, input impedance, common mode rejection ratio, and output voltage swing, among others. The specification of these parameters depends on the particular system under test and must be evaluated accordingly.

TABLE 19. IDENTIFICATION TECHNIQUE PERFORMANCE FOR DIFFERENT AMPLIFIER CONFIGURATIONS, PERFECT A/D CONVERSION, INTEGRATION
TIME = 9.6 μ s, SAMPLING INTERVAL = 4 ns

Filter Poles (MHz) ω_1, ω_2	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
25	0.011555324	0.0375	2.8024309×10^5	-0.16	0.134×10^{-5}
50	10.592746	-0.23	-2.7282039×10^8	-0.316	
50	0.011536579	-0.12	2.7832421×10^5	-0.84	0.196×10^{-4}
100	10.561845	-0.519	-2.7120376×10^8	-0.906	
100	0.011554257	0.028	2.7741538×10^5	-1.16	0.47×10^{-4}
200	10.497776	-1.12	-2.6912371×10^8	-1.66	
200	0.011531949	-0.164	2.8450411×10^5	1.36	0.909×10^{-4}
400	10.814967	1.86	-2.8077018×10^8	2.59	
400	0.011543181	-0.0677	2.9494878×10^5	5.08	0.962×10^{-3}
800	11.190547	5.4	-2.9555270×10^8	7.99	
700	0.011498102	-0.457	2.9802372×10^5	6.17	0.153×10^{-2}
800	11.395104	7.33	-3.0261627×10^5	10.6	
2000	0.011502276	-0.42	3.0687918×10^8	9.32	0.322×10^{-2}
4000	11.708577	10.28	-3.1528646×10^8	15.2	
No Filter	0.011545154	-0.05	2.8063116×10^5	-0.02	0.411×10^{-8}
	10.615088	-0.0187	-2.7362615×10^8	-0.021	

TABLE 20. IDENTIFICATION TECHNIQUE PERFORMANCE FOR DIFFERENT AMPLIFIER CONFIGURATIONS, PERFECT A/D CONVERSION, INTEGRATION TIME = 9.6 μ s, SAMPLING INTERVAL = 4 ns

Filter Poles (MHz) ω_1, ω_2	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
10	0.011528838	-0.1918	2.8048702×10^5	-0.0908	0.597×10^{-6}
11	10.650187	0.3127	-2.7449821×10^8	0.297	
11	0.011566252	0.132	2.8083652×10^5	0.0515	0.389×10^{-6}
12	10.590722	-0.247	-2.7302155×10^8	-0.242	
12	0.011539614	-0.098	2.8054255×10^5	-0.0532	0.148×10^{-7}
13	10.633535	1.559	-2.7407930×10^8	0.144	
13	0.011560636	0.0834	2.8076305×10^5	0.025	0.147×10^{-6}
14	10.601234	-0.148	-2.7327608×10^8	-0.199	
14	0.011588640	0.326	2.8015747×10^5	0.13	0.17×10^{-5}
15	10.561955	-0.518	-2.7230306×10^8	-0.505	
15	0.011540601	-0.09	2.805308×10^5	-0.057	0.91×10^{-7}
16	10.629664	0.119	-2.7397023×10^8	0.104	
16	0.011559574	0.074	2.8072548×10^5	0.0119	0.12×10^{-6}
17	10.603475	-0.127	-2.7331924×10^8	-0.133	
17	0.011514039	-0.3199	2.8022400×10^5	-0.166	0.133×10^{-5}
18	10.666838	0.469	-2.7487678×10^8	0.435	
18	0.011555627	0.04	2.8066038×10^5	-0.0112	0.93×10^{-7}
19	10.606738	-0.965	-2.7338636×10^8	-0.1089	
19	0.011542878	-0.0703	2.8051016×10^5	-0.0647	0.467×10^{-7}
20	10.621596	0.0434	-2.7374330×10^8	0.0215	
20	0.011528307	-0.196	2.8033879×10^5	-0.126	0.33×10^{-6}
21	10.640706	0.223	-2.7420633×10^8	0.191	

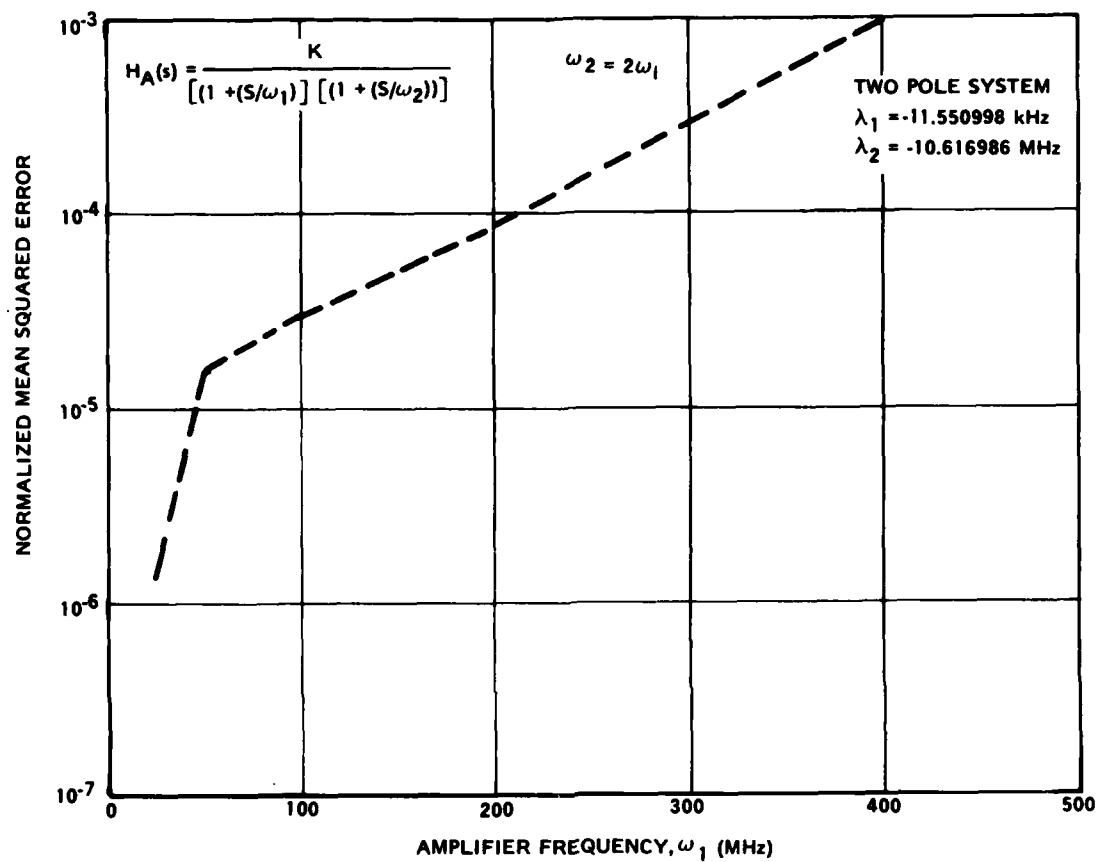


Figure 27. Identification Technique Performance as a Function of Amplifier Characteristics

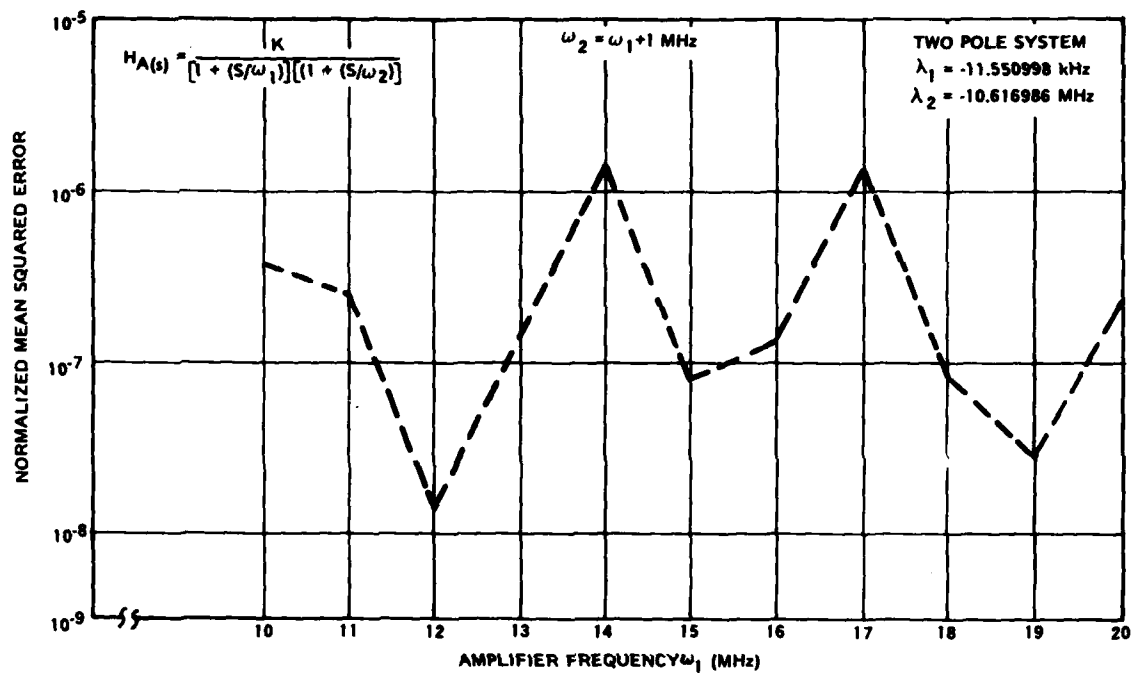


Figure 28. Identification as a Function of Amplifier Characteristics

The required gain of the amplifier is a function of the peak output voltage of the nonlinear system under test and the full-scale input voltage of the A/D converter. This gain can be achieved, if necessary, by cascading multiple amplifiers with similar frequency responses. Common mode rejection needs to be considered because of the small signals that will most likely be generated by the nonlinear system under test. Noise will be a critical factor in implementing a viable experimental measurement setup for the technique and the amplifier common mode rejection capability is a measure of how well noise can be eliminated from the amplifier output. The remaining parameters (slew rate, output voltage swing, etc.) must be specified to be compatible with the other devices of the measurement implementation and with the output of the system under test.

A survey of available amplifier characteristics reveals again a multitude of devices and manufacturers. The wide-band operational amplifiers have a gain-bandwidth product which typically lies in the range of 10 to 200 MHz. (Teledyne Philbrick and Burr Brown make operational amplifiers with a gain bandwidth product of 1000 MHz. This was the maximum gain-bandwidth product determined during the survey.) These devices typically have a frequency response which is flat from dc to $BW/10$, where $BW = 3$ dB bandwidth.

Several of these devices are recommended by the manufacturer for use as pre-A/D converter amplifiers in an implementation considered here.

The available amplifiers appear to be compatible with the existing A/D converters for application to the identification technique implementation for either of the two approaches described above. The fastest high resolution A/D converter (14 to 16 bits) was on the order of 125 kHz sampling rate. In one approach described above, the required amplifier bandwidth would be 1000 times the upper break frequency of the system under test. This upper break frequency would be less than 125 kHz due to sampling considerations but an amplifier bandwidth of 1000 (125 kHz) = 125 MHz would provide the required frequency response. This bandwidth is achievable with a gain of 8 using the 1000 MHz gain bandwidth product device described above. Cascading several of these devices will provide the required gain. If the sampling requirements are taken into account, the system under test would be limited to 20 kHz which would necessitate a 20-MHz amplifier bandwidth which is achievable with the above device having a gain of 50.

The implementation feasibility of the digital configuration of the identification technique does not appear to be limited by amplifier technology at this point in time. The A/D converter remains the critical component in the digital

implementation and its characteristics will determine the overall feasibility of implementing the identification technique.

The amplifier survey was based on amplifier specification data supplied by the following manufacturers:

Plessey Semiconductors
Analog Devices
Harris Semiconductor Products Division
Fairchild Semiconductor
National Semiconductor
Amplifier Research
M.S. Kennedy Corp.
Teledyne Philbrick
RCA
Precision Monolithics
Datel Systems, Inc.
Burr Brown Research Corp.

3. Signal Generator Requirements

The identification technique requires that an input function of the form

$$x(t) = \sum_{i=1}^N e^{-\alpha_i t} u(t) \quad (75)$$

be used to excite the system for identification of $h_2(t_1, t_2)$. The identification of $h_1(t)$ may use an arbitrary waveform (Reference 4) but it would be convenient if the same waveform generator could be used for identification of $h_1(t)$ and $h_2(t_1, t_2)$.

The input, which consists of a sum of decaying exponentials or even a single decaying exponential, is not a convenient waveform in terms of commercially available signal generators. The vast majority of waveform/signal generators provide the square wave, sine wave and triangular waveforms with pulse and frequency modulation options. No commercially available waveform generator with a specific exponential function output was found during the survey of waveform generator manufacturing companies.

The only potential signal generator candidate for use in the identification technique was a system manufactured by WAVETEK. This system was the WAVETEK Model 175 Arbitrary Waveform Generator. This device is a programmable waveform generator which permits the user to store waveforms as digital points on a 256 x 255 grid. The data points are stored in four random access memories (RAM) and are entered via a panel keyboard interface.

This signal generator is a very capable device but has some limitations that will affect its use in the identification technique. The output of the signal generator is piece-wise continuous linear between sample points which results in a distorted exponential input. The dynamic range of the device is 20 volts to 2 millivolts. This corresponds to a limitation imposed by a 13 bit A/D converter operating on an analog waveform generator. The output amplitude resolution is 1/256 which restricts the time period over which the exponential input can be used for the identification technique.

For the two-pole example of interest, the input $x(t) = 10^{-3}e^{-10^7 t}u(t)$ would be resolved only to a minimum amplitude of 3.9×10^{-6} which corresponds to a integration period of approximately 5.5×10^{-7} second. Beyond this time period, the input function would be zero.

Although the arbitrary waveform generator provides a level of capability beyond the typical commercially available waveform generator, it does not appear to meet the accuracy requirements of the identification technique.

The conclusion of the survey of commercially available signal/waveform generators is that no signal generator on the present market can generate a sum of decaying exponential functions. This requires that a circuit be designed to provide this input.

There are several options available at this point. The first option is to use the system shown in Figure 29.

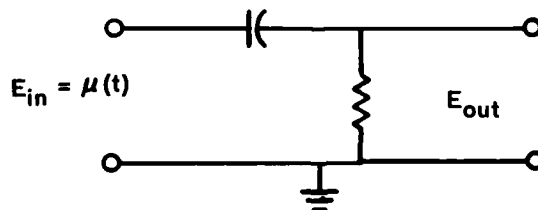


Figure 29. RC Networks

The output voltage of this system is

$$E_0(s) = \frac{1}{s + \frac{1}{RC}} \quad (76)$$

or, in the time domain

$$e_0(t) = e^{(-1/RC)t} u(t) \quad (77)$$

This is the function required -- a decaying exponential. The circuit is driven by a unit step input which is easily generated using standard equipment.

A sum of decaying exponentials can be generated using several of these passive networks in parallel with their outputs summed using an operational amplifier as a summary amplifier. The time constants ($1/RC$) of each network are set by appropriate selection of R and C . One requirement is that the input impedance of the summing amplifier not load down the passive network and effectively change the time constant.

Another implementation of the signal generator is to use an operational amplifier as shown in Figure 30.

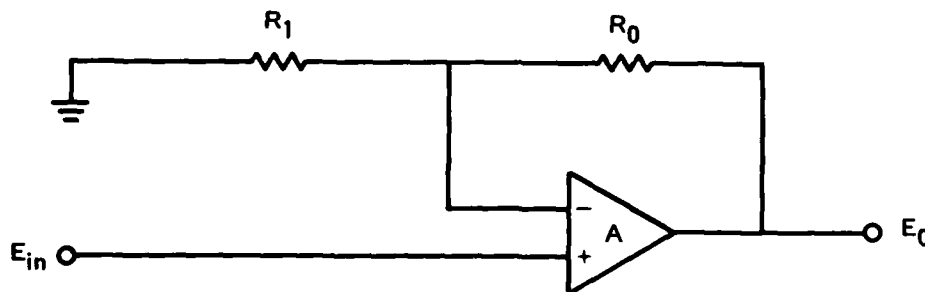


Figure 30. Operational Amplifier Network for Signal Generator

The closed loop transfer function of this amplifier circuit is

$$G_c(s) = \frac{\frac{R_0 + R_1}{R_1}}{1 + \frac{s}{2\pi f_0 A_0} \left(\frac{R_0 + R_1}{R_1} \right)} = \frac{K}{1 + \frac{s}{\omega'_0}} \quad (78)$$

where

$$K = (R_0 + R_1)/R_1, \quad \omega'_0 = 2\pi f_0 A_0/K$$

and f_0 is the high frequency cutoff of the operational amplifier.

The impulse response of this network is

$$e_0(t) = K \omega'_0 e^{-\omega'_0 t} = 2\pi f_0 A_0 e^{-\omega'_0 t} \quad (79)$$

A practical impulse is actually a short pulse of width δ . For this practical impulse, the output of the circuit is

$$\begin{aligned} e_0(t) &= (K - K e^{-\omega'_0 t}) u(t) & 0 \leq t < \delta \\ &= K (e^{\delta \omega'_0} - 1) e^{-\omega'_0 t} & t \geq \delta \end{aligned} \quad (80)$$

This output is shown in Figure 31.

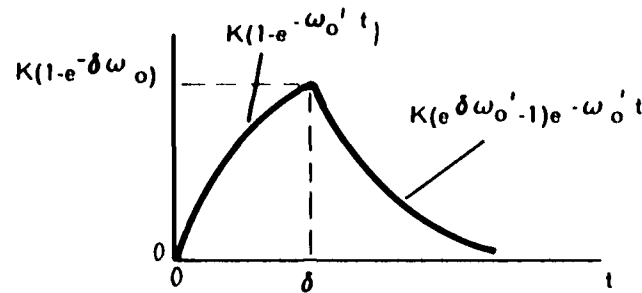


Figure 31. Short Pulse Response of Operational Amplifier Network of Figure 30

The portion of the output of the operational amplifier that represents the exponential input is for $t > \delta$. Therefore it is necessary to use an analog switch to clamp the operational amplifier from time $t = 0$ to $t = \delta$.

The amplitude of the exponential function generated by the operational amplifier will be controlled by an attenuator connected in series with the operational amplifier circuit. The unattenuated amplitude of the decaying exponential is

$$\frac{R_0 + R_1}{R_1} \left(e^{\delta \left(\frac{2\pi f_0 A_0 R_1}{R_0 + R_1} \right) - 1} \right) \quad (80a)$$

The time constant is a function of the open-loop gain (A_0) of the operational amplifier, the open-loop 3-dB bandwidth (f_0), and the resistors R_1 and R_0 . The open loop gain and bandwidth are functions of the operational amplifier selected for use while R_1 and R_0 are selectable at the discretion of the user.

A sum of decaying exponential functions can be generated by exciting a parallel set of these operational amplifier networks with the same input pulse and using a summing operational amplifier configuration to add the functions. The analog switch is then used to clamp the output until time $t = \delta$. A sample configuration for $N = 2$ is shown in Figure 32.

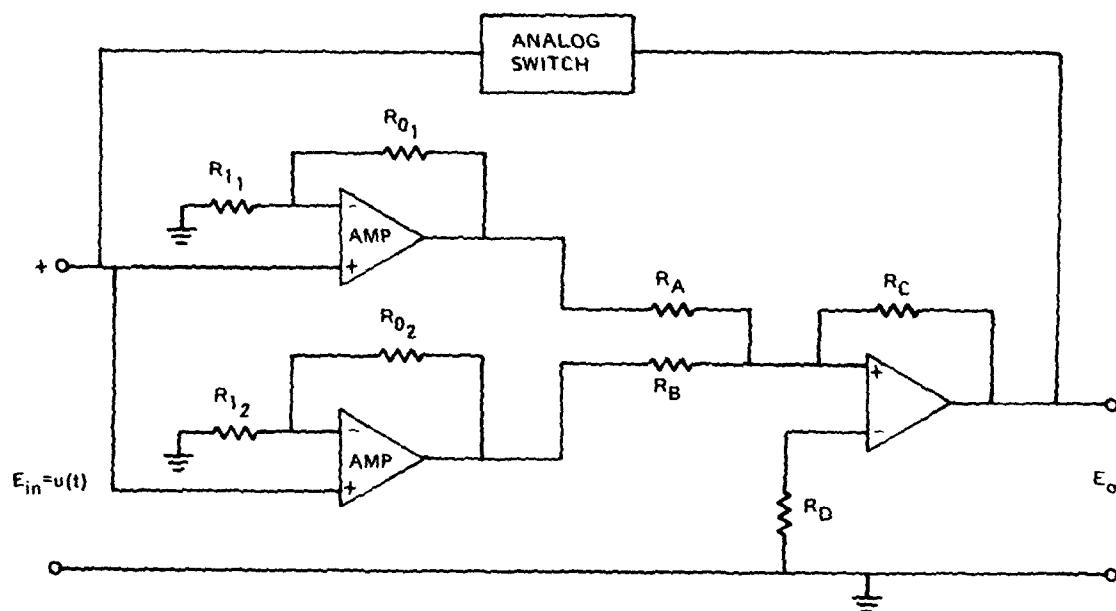


Figure 32. Signal Generator Configuration for $N = 2$

The output is the negative sum of the input exponential functions. This can be reinverted using another inverting operational amplifier. Proper selection of R_A , R_B , and R_C will serve to properly attenuate the amplitude of each exponential function prior to summing. The output of this system is

$$e_0(t) = - \left(\frac{R_C}{R_A} \frac{(R_{01} + R_{11})}{R_{01}} \left| e^{\delta\omega_{01}'t} - 1 \right| e^{-\omega_{01}'t} + \frac{R_C}{R_B} \frac{(R_{02} + R_{12})}{R_{02}} \left| e^{\delta\omega_{02}''t} - 1 \right| e^{-\omega_{02}'t} \right) \quad (81)$$

where

$$\omega_{0i}' = \frac{2\pi f_{0i} A_{0i}}{\left(\frac{R_{0i} + R_{1i}}{R_{0i}} \right)}, \quad i = 1, 2. \quad (82)$$

4. Data Storage Requirements

The data storage requirements for the digital implementation are a function of the number of A/D converter bits and the number of samples required for the input and output. The number of digital bits requiring storage for the digital implementation is given by

$$\text{Data Bits} = 2 \frac{T}{T_s} \cdot M \quad (83)$$

where

T is the integration period

T_s is the sampling interval

M is the number of A/D converter bits

The integration period and sampling interval are a function of the system under test and it is difficult to establish a fixed number for these values. For the example systems, the maximum number of samples used was 3200. This was limited by the memory capacity of the GP computer used for the simulation. For this case the number of data bits to be stored for a 16 bit A/D converter is

$$\text{Data Bits} = 2(3200)(16) = 102,400 \text{ bits}$$

An upper limit of 8000 samples seems to be reasonable for systems to which the technique can be applied with existing components. This requires a storage capacity of

$$\text{Data Bits} = 2(8000)(16) = 256,000 \text{ bits}$$

The data bits can be stored using static RAM. RAM's with a 2048 word x 8 bits organization are currently available. Two of these are required to store 2048 words of 16 bit length. To store 16,000 words of 16 bit length requires 16 of these RAM chips. These devices have a typical access time of 200 ns, which is an indication of the time required to write an 8-bit word into memory. This is easily compatible with the fastest 16-bit A/D converter presently available (conversion time = 8 μ s).

Although establishing a maximum level of data storage required is difficult, it appears that memory devices are available to handle the data storage for any practical implementation of the identification technique. As the number of samples increases and as data storage requirements increase,

it will, of course, be necessary to build a larger memory by adding chips and any necessary interface circuitry at additional cost. However, the capability is available if needed, and the data storage requirements do not appear to be technology limited at this time. The only limitation of these devices could be in the area of the access time. However, these devices are one to two orders of magnitude faster than the currently available A/D converters. This trend will probably be maintained as both memory technology and A/D converter technology advance.

Once the data is stored in the static RAM, it is necessary to transmit this data to the appropriate storage device for future non-real time processing using the GP computer. This storage device may be a time sharing data file, magnetic tape, paper tape, or punched data cards. Data transmission is most easily accomplished using a device such as a USART (Universal Synchronous/Asynchronous Receiver/Transmitter, Intel 8251A). This device is a programmable communication interface capable of transmitting the data from the RAM to the permanent storage device.

5. Digital Implementation - Conclusions

The performance evaluation of the digital implementation of the identification technique suggests that the critical components of the digital implementation are the A/D converter and the pre-A/D converter amplifier. The important conclusions impacting parameter specification of these devices are summarized below.

a. A/D Converter

- (1) The A/D converter must have 14-16 bits of resolution for adequate performance with a two-pole system. This increases to 20 to 24 bits as the number of poles increases to four. Since the highest resolution commercially available A/D has 16 bits of resolution at this point in time, any experimental validation of this implementation should be restricted to systems with two poles or fewer.
- (2) The sampling rate requirements for the A/D converter are driven by the accuracy requirements of the processing technique. For a two-pole system, the sampling rate should be 4 to 10 times slower than the highest break frequency of the system under test. The fastest 16 bit A/D converter currently available is limited to a sampling

rate of 125 kHz. This implies that the system under test must be limited to an upper break frequency of approximately 10 to 30 kHz.

b. Amplifier

- (1) Three approaches to the identification problem are designed to handle the complication of using a pre-A/D converter amplifier. The first approach is to use the identification technique directly and identify a transfer function that is the product of the transfer functions of the system under test and the amplifier. This increases the dimension of the identification problem, which will probably degrade performance of the identification technique. This approach requires an amplifier that is approximately equal in frequency extent to the system under test.

The second approach is to use a very wide-band (compared to system under test) amplifier. The lower 3-dB break frequency of the amplifier should be 1/1000 of the lower break frequency of the test system. The upper 3-dB break frequency of the amplifier should be 1000 times the upper break frequency of the test system.

The final approach is to use linear system theory to modify the required identification processing. This requires that the amplifier be approximately matched in frequency extent to the system under test.

- (2) Operational amplifiers with gain-bandwidth products up to 1000 MHz are presently commercially available. These will work well with all the processing approaches described in (1) above.
- (3) The amplifier gain requirement is determined by the peak output voltage of the system under test and the full-scale input voltage of the A/D converter.

c. Remaining Components

The remaining components of the digital implementation did not appear to be technology limited in terms of enabling implementation of the identification technique. The important conclusions are given below.

- (1) There is no commercially available waveform generator with an exponential function capability. The appropriate inputs will be generated by using operational amplifiers as low-pass filters and appropriately damping the short pulse response to obtain the exponential function.
- (2) Data storage is accomplished using static random access memory (RAM) chips and a programmable interface to transmit the data to the digital computer for nonreal-time processing.

d. Summary

The analysis of the digital implementation supports the following important conclusion. The pencil-of-functions technique requires high accuracy to perform identification satisfactorily. This accuracy is impacted by the choice of the numerical integration technique used in the processing. Simpson's rule of numerical integration results in increased error as the number of system poles increases. There is a need to determine the best integration technique for the identification processing. This issue will be dealt with in more detail during Part II of the study.

Given the constraints presented above, the digital implementation of the identification technique can feasibly be constructed and used in an experimental test setup.

A remaining issue is the problem of noise in the implementation. A 16-bit A/D converter with a 10 volt full-scale input can resolve a signal of 152 microvolts. These low level signals require careful handling to reduce the impact of noise on the performance of the technique. The devices used in the implementation must be extremely low noise components and advantage must be taken of any common mode rejection capability available in the measurement configuration.

C. HYBRID IMPLEMENTATION

During the investigation of the digital implementation, it became apparent that the numerical integration of the A/D con-

verted samples was a primary source of performance degradation. A hybrid implementation of the identification technique was postulated to possibly alleviate this performance degradation. This implementation is shown in Figure 33. The input and output of the test system are integrated N times using analog integrators. The integrator outputs are appropriately amplified and sampled by a set of A/D converters. These samples are then stored for further nonreal-time pencil-of-functions processing on the GP computer.

The potential advantages of this configuration are that it will minimize the error in the integrated outputs by eliminating the need for numerical integration of the outputs. (The inner products will still be formed using numerical integration of the products of the A/D converted samples.) In addition, this implementation eliminates the "shrinking number of samples" problem encountered using Simpson's rule as discussed in Section II.B. However, this implementation has several disadvantages when compared to the digital implementation of paragraph III.B. These are apparent from Figure 33 where it is observed that $2N + 1$ A/D converters are required for implementation as well as $2N + 1$ analog integrators and $2N + 1$ amplifiers. The digital implementation required only two A/D converters and two amplifiers. For a complex system (N large), this implementation could become complex and costly compared to the digital system. Another disadvantage lies in the fact that N must be known before the final implementation configuration is established. This complicates the procedure of evaluating N since the measurement setup must be changed for each iteration of the procedure to evaluate system order. It is recommended that the system order be established using the digital approach if possible and that the hybrid implementation be used only for pole and residue determination if its performance warrants its use. The performance of the implementation is addressed in the following sections.

1. Simulation of Hybrid Implementation

The simulation of the identification technique was modified to represent the hybrid implementation. The numerical integration process was deleted and the analytical expressions for the integrated outputs and inputs were inserted into the program. A listing of the simulation for this implementation is provided in Appendix B.

The general form of the output of the system under test is

$$y(t) = \sum_{i=1}^N C_i e^{\lambda_i t} \quad (84)$$

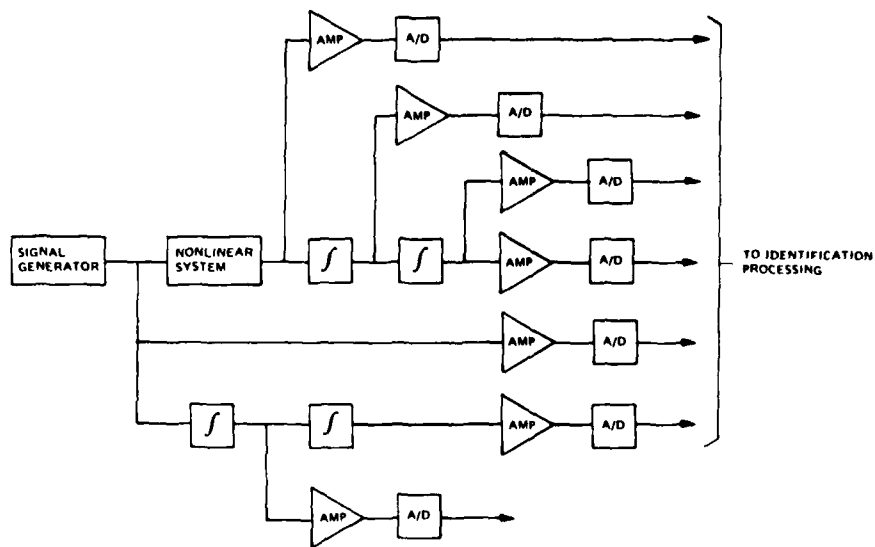


Figure 33. Hybrid Implementation

The resulting integrations yield

$$\bar{y}_2(t) = \int_0^t y(\tau) d\tau = \sum_{i=1}^N \frac{C_i}{\lambda_i} (e^{\lambda_i t} - 1) \quad (85)$$

$$\begin{aligned} \bar{y}_3(t) &= \int_0^t \bar{y}_2(\tau) d\tau \\ &= \sum_{i=1}^N \left[\frac{C_i}{\lambda_i^2} (e^{\lambda_i t} - 1) - \frac{C_i}{\lambda_i} t \right] \end{aligned} \quad (86)$$

$$\bar{y}_4(t) = \int_0^t \bar{y}_3(\tau) d\tau = \sum_{i=1}^N \left[\frac{C_i}{\lambda_i^3} (e^{\lambda_i t} - 1) - \frac{C_i}{\lambda_i^2} t - \frac{C_i}{2\lambda_i} t^2 \right] \quad (87)$$

$$\begin{aligned} \bar{y}_5(t) &= \int_0^t \bar{y}_4(\tau) d\tau = \sum_{i=1}^N \left[\frac{C_i}{\lambda_i^4} (e^{\lambda_i t} - 1) - \frac{C_i}{\lambda_i^3} t \right. \\ &\quad \left. - \frac{C_i}{\lambda_i^2} \frac{t^2}{2} - \frac{C_i}{6\lambda_i} t^3 \right] \end{aligned} \quad (88)$$

The general form of the input is

$$x(t) = A_i e^{a_i t} \quad (89)$$

and the resultant integrations are similar in form to those for $y(t)$ (setting $N = 1$).

2. Two-Pole System Analysis and Results

The two pole system investigated in paragraph III.B was used to evaluate the performance of the hybrid implementation. The performance results for this implementation are shown in Table 21 for different A/D converter resolutions. Figure 34 is a plot of normalized mean squared error vs A/D converter resolution. These results indicate that this approach does not

TABLE 21. SIMULATION RESULTS FOR HYBRID IMPLEMENTATION FOR
FOR DIFFERENT LEVELS OF A/D CONVERTER RESOLUTION,
INTEGRATION TIME = 9.6 μ s, SAMPLING INTERVAL = 4 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Square Error
No A/D Converter	0.011554792	0.0328	2.8073336×10^5	0.0147	0.356×10^{-7}
	10.60926	-0.0727	-2.7349133×10^8	-0.705	
16	0.011620211	0.599	2.8009951×10^5	-0.211	0.436×10^{-4}
	10.34654	-2.54	-2.667889×10^8	-2.52	
14	0.012683786	9.807	2.8531059×10^5	1.64	0.809×10^{-3}
	9.5219201	-10.31	-2.4551532×10^8	-10.29	
12	0.023587263	104.2	3.9067486×10^5	39.18	0.589×10^{-1}
	4.4864759	-57.7	-1.210728×10^8	-55.76	

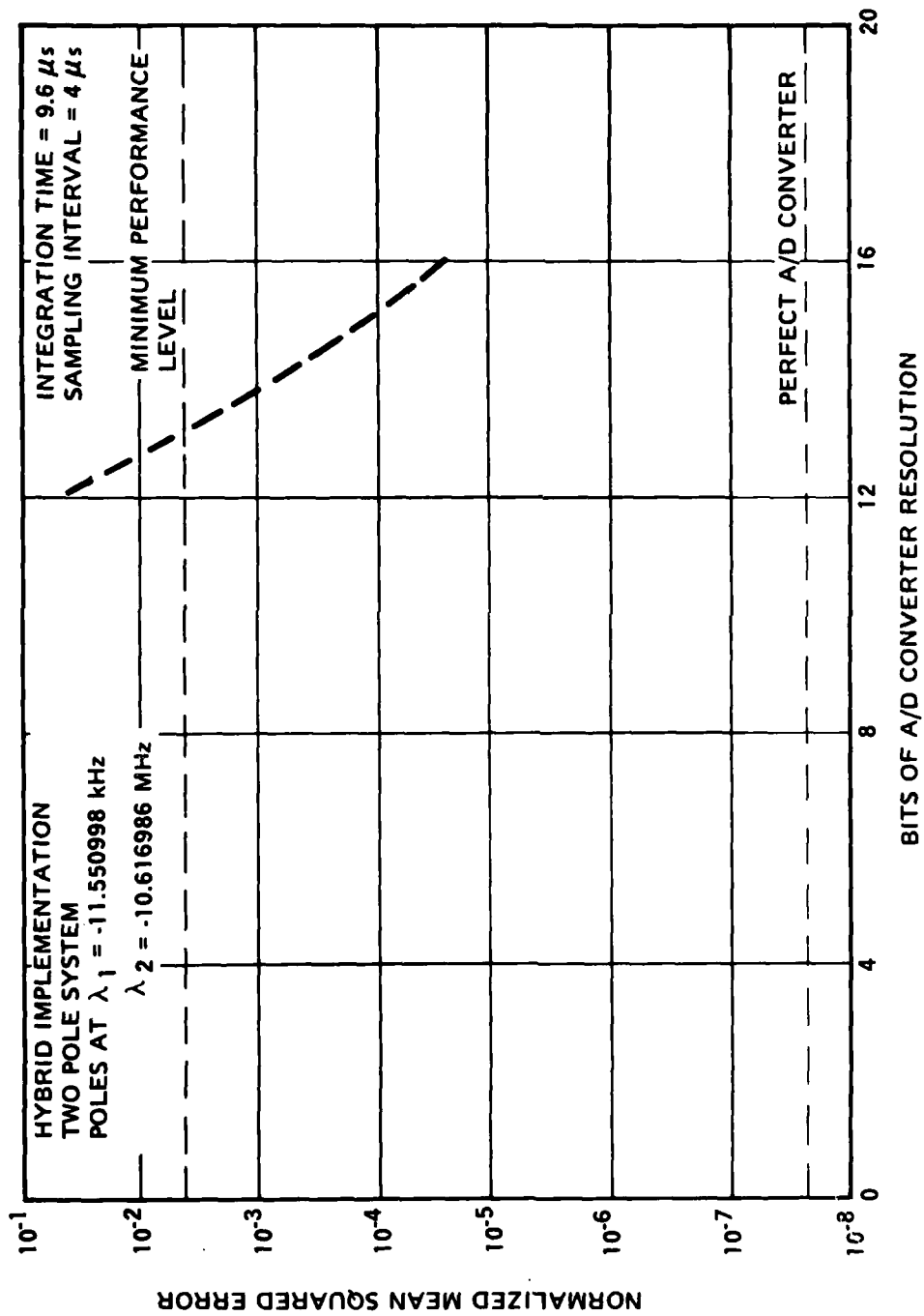


Figure 34. Hybrid Implementation Identification Performance as a Function of A/D Converter Resolution

perform as well as the digital implementation (see Table 3). Acceptable performance is obtained for A/D converters of 14 bits or greater resolution whereas, for the digital implementation, acceptable performance is obtained for A/D converters with resolution of 10 bits or greater.

The apparent reason for the poorer performance of this implementation compared to the digital implementation is that the implementation accuracy of the integrator outputs is greater for the latter than for hybrid implementation. Every sample of the integrated output and input is converted to an N-bit digital representation in the hybrid implementation. In the digital implementation, the integrated output and input are formed by numerically integrating the N-bit samples of the system input and output. The numerical integration is accomplished using the full capability of the GP computer machine accuracy. The samples are accumulated and not rounded to 16 bits as is the case for the hybrid implementation. Therefore, the digital implementation is actually more accurate than the hybrid implementation.

3. Four-Pole System Analysis and Results

Although the performance of the hybrid implementation was not as good as the digital implementation for the two-pole system, its performance for the four-pole system of paragraph III.B.1.c.(2) was investigated. The results of the investigation are presented in Table 22. The performance of the hybrid implementation is significantly better than the digital implementation for a 24-bit A/D converter and an ideal A/D converter. The performance of the hybrid implementation is slightly better than the digital implementation for a 20-bit A/D converter but its performance deteriorates significantly for a 16-bit A/D converter. With 16-bit resolution, the hybrid implementation identifies a pair of complex conjugate poles instead of poles at 0.51 and 0.82 MHz.

The results of Table 22 are useful in another respect. The performance of the hybrid implementation with an ideal A/D converter (machine accuracy) is significantly better than that of the digital implementation. This is highlighted in Table 23. These results indicate the effect of numerical integration on the performance of the pencil-of-functions approach. For systems with $N > 2$, the numerical integration using Simpson's rule appears to be a serious limitation to the performance of the technique.

TABLE 22. SIMULATION RESULTS FOR HYBRID IMPLEMENTATION FOR
DIFFERENT LEVELS OF A/D CONVERTER RESOLUTION, FOUR-POLE SYSTEM,
INTEGRATION TIME = 4.8 μ s, SAMPLING INTERVAL = 1.5 ns

Number of A/D Bits	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
No A/D Converter	0.011550994 0.5099998	-0.3×10^{-4} -0.34×10^{-4}	2.8069668×10^5 -1.200447×10^7	0.0017 0.037	0.775×10^{-8}
	0.8200003 6.499999	0.53×10^{-4} 0.8×10^{-5}	1.5111038×10^7 -1.61031116×10^8	0.073 0.0193	
24	0.011550716 0.5100335	0.00244 0.00657	2.8068844×10^5 -1.200283×10^7	-0.00123 0.0236	0.245×10^{-9}
	0.8199492 6.499999	-0.006189 -0.000321	1.510175×10^7 -1.6099426×10^8	0.0116 -0.0035	
20	0.0116280268 0.511678239	0.6668 0.329	2.82697507×10^5 -1.3764095×10^7	0.714 14.7	0.851×10^{-3}
	0.815207 6.509094	-0.584 0.140	1.900147×10^7 -1.7141638×10^8	25.8 6.47	
16	0.38824 $\pm$ 0.28327 0.0226615 7.617245	---- 96. 17.2	----	----	----

TABLE 23. COMPARISON OF HYBRID AND DIGITAL IMPLEMENTATION
PERFORMANCE FOR FOUR-POLE SYSTEM, INTEGRATION TIME = 4.8 μ s,
SAMPLING INTERVAL = 1.5 ns

Implementation	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
Hybrid	0.011550994 0.5099998	-0.3×10^{-4} -0.34×10^{-4}	2.8069668×10^5 -1.200447×10^7	0.0017 0.037	0.775×10^{-8}
	0.8200003 6.499999	0.53×10^{-4} 0.8×10^{-5}	1.5111038×10^7 -1.61031116×10^8	0.073 0.0193	
Digital	0.011279618 0.4988702	-2.35 -2.18	2.789548×10^5 -1.048074×10^7	-0.619 -12.66	0.103×10^{-4}
	0.8540037 6.453757	4.14 -0.711	1.3534398×10^7 -1.594465×10^8	-10.36 -0.964	

4. Amplifier, Signal Generator, and Data Storage Requirements

The signal generator and amplifier requirements for the hybrid implementation are essentially the same as those detailed for the digital implementation. The amplifiers in the hybrid implementation will have different gains to match the integrator outputs to the appropriate A/D converter full-scale input voltage but the frequency characteristics required are the same as those required for the digital implementation.

The data storage requirements are more complicated for the hybrid implementation because there are $2N + 1$ data streams to be recorded and stored in memory. This implies that a larger number of memory chips is required; however, there is no inherent technology limitation in meeting the data storage requirements. The cost of the data storage system will be approximately $(N + 1)/2$ times that required for the digital implementation.

5. Conclusions - Hybrid Implementation

The resolution requirements for the hybrid implementation are significantly greater than for the digital implementation. Since the hybrid implementation require A/D converters of approximately 24 bits, this implementation is beyond the current state of the art (16 bits). For systems with two poles or fewer, the hybrid implementation offers no advantages over the digital implementation. For systems with more than two poles, the hybrid implementation offers potential performance improvement over the digital approach for an A/D converter with 24 bits. This improvement, however, increases the measurement implementation complexity and cost.

It is not feasible to consider implementation of this approach for an experimental validation at this time because of the A/D converter requirement.

Future improvements in A/D converter technology may permit implementation of this approach at that time. Therefore, the approach cannot be dropped from total consideration at this time but should be relegated to a low priority position until the A/D converters become available.

D. ANALOG IMPLEMENTATION

The analog implementation for the identification technique is shown in Figure 35 for $N = 2$.

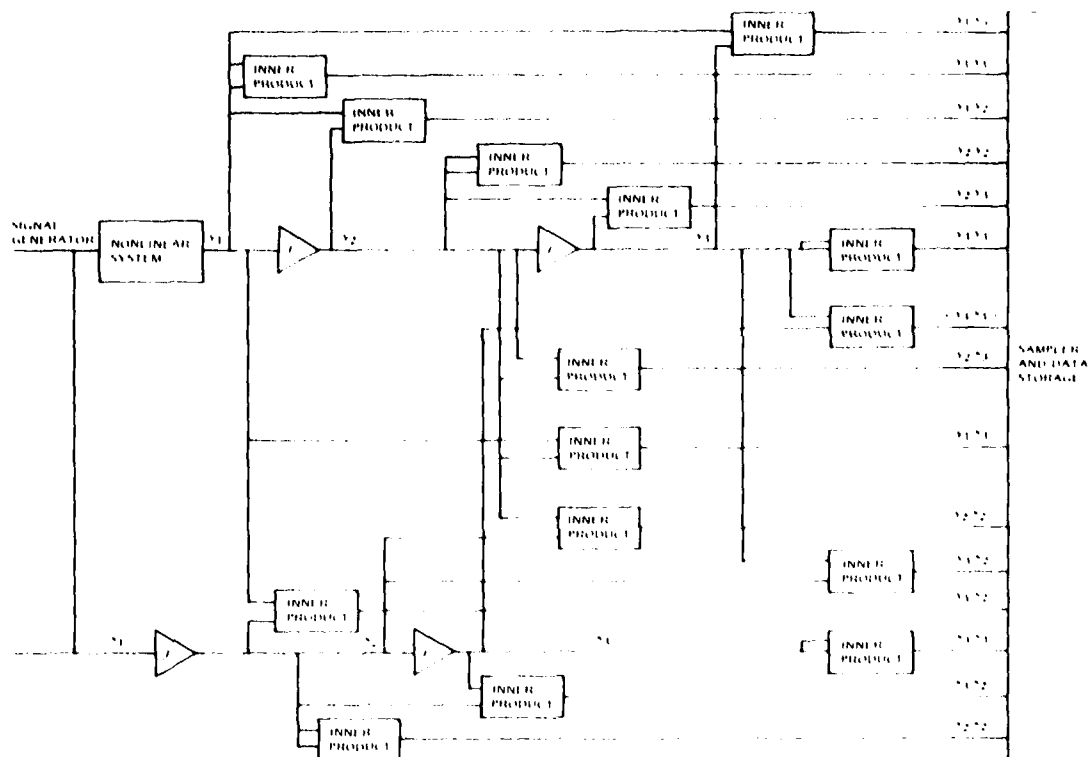


Figure 35. Analog Implementation

The block labeled "inner product" in Figure 35 performs the operation shown in Figure 36.

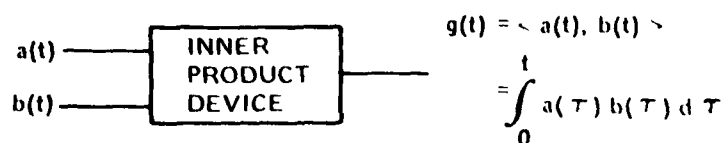


Figure 36. Inner Product Device Model

The analog implementation forms the inner products for the Gram matrix of the pencil-of-functions approach using analog devices. Analog integrators are used to obtain the integrated input and output functions, and inner product devices that multiply the two inputs and integrate the product are used to form the Gram matrix entries. The output of each inner product

device is sampled using A/D converters and stored for further processing on a general purpose computer.

1. Digital Simulation - Analog Implementation

The simulation of the identification technique was modified to represent the analog implementation. The numerical integration for the inner products was removed and replaced by the analytical functions for the inner products. The various integrated inputs, outputs, and inner products for $N = 2$ are listed in the FORTRAN computer program listing in Appendix C.

2. A/D Converter Requirements

The A/D converter requirements for the analog implementation are considered in this paragraph. Only one sample of the output of each inner product device is necessary for pencil-of-functions processing. Therefore, the conversion time requirements for the A/D converters in this implementation are considerably different from those of the digital implementation. If the output of each inner product device is sampled and held at the end of the integration period, then the A/D converters can take as long as necessary to convert the input. This is significant because, as was observed in paragraph III.B.1.g, generally the higher the resolution of the A/D converter, the slower the conversion time. Also, because only a single sample per inner product device is required, it is feasible to think of using a single A/D converter to convert all inner product outputs via multiplexing circuitry.

The resolution requirements for the A/D converters are discussed in the following section.

a. Two-Pole System Analysis and Results

The performance of the analog implementation was evaluated for the two-pole system of paragraph III.B.1.c.(1). The assumed impulse response of the system is

$$h(t) = 2.8069192 \times 10^5 e^{-0.011550998 (2 \pi \times 10^6)t} - 2.7368441 \times 10^8 e^{-10.616986 (2 \pi \times 10^6)t}, \quad t > 0 \quad (90)$$

For the initial performance evaluation, the analog integrators and inner product devices were assumed perfect. The inner

product device outputs were sampled with A/D converters whose most significant bit magnitude was established on the basis of the peak value of each inner product device. The performance results for different levels of A/D converter resolution are presented in Table 24. These results indicate that 18-20 bit resolution is required for satisfactory performance. This is beyond the current state of the art in A/D converters and restricts the feasibility of this implementation in the present time frame. In general, the performance of the analog implementation is significantly less than that of the digital implementation. Once again, the reason for this appears to be the fact that the digital implementation forms the inner products by accumulating products of N-bit numbers using full machine accuracy. Consequently, the resolution of the inner product is greater than that obtained using the analog implementation which uses a N-bit representation of the final inner product value.

b. Integrator and Inner Product Device Requirements

The A/D converter requirements were determined in section a. above, assuming perfect integrator and inner product devices. This is impossible to achieve in a practical system. This paragraph evaluates accuracy requirements for the integrators and inner product devices.

Each entry of the Gram matrix will be in error prior to A/D conversion. These error will arise from the imperfections of the integrator and the inner product device, and these will be a function of the devices and the inputs to the devices. In order to evaluate the tolerable magnitude of these errors using the computer simulation, they were assumed to be uniformly distributed between $\pm K$ percent where K is an input to the simulation. Each inner product is evaluated using the exact expression in the simulation and is then multiplied by $(1 + 100 \cdot X)$ where X is uniformly distributed between $\pm K$ percent and is selected independently for each inner product.

The performance results using this error model are provided in Table 25. These results assume a perfect A/D converter (machine accuracy). The results of Table 25 indicate that an inner product total error of less than 10^{-3} percent must be maintained to achieve satisfactory performance.

The performance of the analog implementation was also evaluated taking into account the effects of the A/D converter resolution. These results are provided in Tables 26 through 29 for A/D converters of 24, 20, 18 and 16 bits, respectively. Figure 37 is a plot of normalized mean squared error as a function of inner product error. These results support the conclusion that the total inner product error must

TABLE 24. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION FOR DIFFERENT LEVELS OF A/D CONVERTER RESOLUTION, INTEGRATION TIME = 9.6 μ s, ANALYTICAL INTEGRATION

No. of A/D Converters	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
No A/D Converter	0.01155099799 10.616986036	-0.71 x 10 ⁻⁷ 0.34 x 10 ⁻⁶	2.80691908 x 10 ⁵ -2.73684410828 x 10 ⁸	2.87 x 10 ⁻⁵ 3.03 x 10 ⁻⁵	0.545 x 10 ⁻¹⁸
24	0.01154828177 10.620821638	-0.023 0.0361	2.8060527118 x 10 ⁵ -2.737528285 x 10 ⁸	-0.0308 -0.025	0.154 x 10 ⁻⁷
22	0.01153782195 10.63620273	-0.114 0.181	2.80271658 x 10 ⁵ -2.7403180907 x 10 ⁸	-0.149 0.127	0.37 x 10 ⁻⁶
20	0.01149365755 10.70146787	-0.496 0.796	2.788648036 x 10 ⁵ -2.752142993 x 10 ⁸	-0.651 0.559	0.705 x 10 ⁻⁵
18	0.011319341 10.970104	-2.0 3.33	2.73339299 x 10 ⁵ -2.801041896 x 10 ⁸	-2.62 2.34	0.117 x 10 ⁻³
16	0.010570331 12.3080966	-8.49 15.9	2.50092183 x 10 ⁵ -3.045420229 x 10 ⁸	-10.9 11.27	0.217 x 10 ⁻²
14	0.00637772 12.766525	-44.8 491.3	1.3419145 x 10 ⁵ -1.27665 x 10 ⁹	-52.2 366.5	0.92 x 10 ⁻¹
12	0.01957734 5.8839899	69.5 -44.6	5.8572269 x 10 ⁵ -2.017712 x 10 ⁸	108.7 -26.2	0.10003

TABLE 25. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION^a FOR DIFFERENT LEVELS OF INNER PRODUCT ERROR, INTEGRATION TIME = 9.6 μ s, PERFECT A/D CONVERTER

Inner Product Error (%)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
0	0.01550998 10.61698603	-7.1 x 10 ⁻⁶ 3.4 x 10 ⁻⁵	2.806919208 x 10 ⁵ -2.7368441 x 10 ⁸	2.8 x 10 ⁻⁵ 3.03 x 10 ⁻⁵	0.545 x 10 ⁻¹⁸
10 ⁻⁵	0.011548744 10.61438876	-0.0195 -0.0244	2.80620361 x 10 ⁵ -2.73592359 x 10 ⁸	-0.025 -0.033	0.154 x 10 ⁻⁷
10 ⁻⁴	0.0115284433 10.59122218	-0.195 -0.242	2.799762306 x 10 ⁵ -2.72769677 x 10 ⁸	0.25 -0.33	0.153 x 10 ⁻⁵
10 ⁻³	0.0113241596 10.37901461	-1.96 -2.24	2.73525408 x 10 ⁵ -2.650797367 x 10 ⁸	-2.55 -3.14	0.147 x 10 ⁻³
10 ⁻²	0.0091189174 9.53705737	-21.05 -10.2	2.0748628 x 10 ⁵ -2.21483894 x 10 ⁸	-26.1 -19.1	0.1187 x 10 ⁻¹
10 ⁻¹	0.015862279 2.93256084	37.3 -72.4	4.34522529 x 10 ⁵ -8.9260807 x 10 ⁷	54.8 -67.4	0.147

TABLE 26. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION FOR DIFFERENT LEVELS OF INNER PRODUCT ERROR, INTEGRATION TIME = 9.6 μ s, 24 BIT A/D CONVERTER

Inner Product Error (%)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
0	0.01154828117	-0.023	2.8060527 x 10 ⁵	-0.0308	0.154 x 10 ⁻⁷
	10.620821638	0.0361	-2.73752828 x 10 ⁸	-0.025	
10 ⁻⁵	0.011545663	-0.046	2.8052219 x 10 ⁵	-0.0605	0.462 x 10 ⁻⁷
	10.617276	0.0027	-2.7363229 x 10 ⁸	-0.019	
10 ⁻⁴	0.011527022	-0.207	2.79930973 x 10 ⁵	-0.271	0.159 x 10 ⁻⁵
	10.5927707	-0.228	-2.7279368 x 10 ⁸	-0.325	
10 ⁻³	0.011321356	-1.99	2.734369 x 10 ⁵	-2.58	0.148 x 10 ⁻³
	10.3823478	-2.21	-2.651341 x 10 ⁸	-3.12	
10 ⁻²	0.0091152623	-21.1	2.0738214 x 10 ⁵	-26.1	0.118 x 10 ⁻¹
	9.53948438	-10.1	-2.2150356 x 10 ⁸	-19.2	
10 ⁻¹	0.0158644	37.3	4.346 x 10 ⁵	54.8	
	2.932356	-72.4	-8.9261 x 10 ⁷	-67.4	0.147

TABLE 27. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION FOR DIFFERENT LEVELS OF INNER PRODUCT ERROR, INTEGRATION TIME = 9.6 μ s, 20 BIT A/D CONVERTER

Inner Product Error (%)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
0	0.01149365755	-0.496	2.78864803 x 10 ⁵	-0.651	0.705 x 10 ⁻⁵
10 ⁻⁵	10.70146787	0.796	-2.752142993 x 10 ⁸	0.559	
	0.0114933065	-0.499	2.78853978 x 10 ⁵	-0.655	0.677 x 10 ⁻⁵
	10.6961893	0.746	-2.7507474 x 10 ⁸	0.508	
10 ⁻⁴	0.0114723125	-0.68	2.7818942 x 10 ⁵	-0.891	0.977 x 10 ⁻⁵
	10.6732183	-0.53	-2.742489 x 10 ⁸	0.206	
10 ⁻³	0.0112657849	-2.47	2.716844 x 10 ⁵	-3.21	0.173 x 10 ⁻³
	10.457255	-1.5	-2.66433697 x 10 ⁸	-2.65	
10 ⁻²	0.090454018	-21.7	2.0539462 x 10 ⁵	-26.8	0.123 x 10 ⁻¹
	9.59609748	-9.61	-2.22112449 x 10 ⁸	-18.8	
10 ⁻¹	0.015904702	37.7	4.3616566 x 10 ⁵	55.4	0.148
	2.9293634	-72.4	-8.92965 x 10 ⁷	-67.4	

TABLE 28. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION FOR DIFFERENT LEVELS OF INNER PRODUCT ERROR, INTEGRATION TIME = 9.6 μ s, 18 BIT A/D CONVERTER

Inner Product Error (%)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
0	0.011319341	-2.0	2.73339299 x 10 ⁵	-2.62	0.117 x 10 ⁻³
	10.970104	3.33	-2.801041896 x 10 ⁸	2.34	
10 ⁻⁵	0.01131668337	-2.03	2.732557712 x 10 ⁵	-2.65	0.1175 x 10 ⁻³
	10.96722873	3.3	-2.800001816 x 10 ⁸	2.31	
10 ⁻⁴	0.0112962512	-2.2	2.726142 x 10 ⁵	-2.88	0.124 x 10 ⁻³
	10.94197826	3.06	-2.791203967 x 10 ⁸	1.99	
10 ⁻³	0.0110874443	-4.01	2.660901177 x 10 ⁵	-5.2	0.338 x 10 ⁻³
	10.70635159	0.84	-2.70765252 x 10 ⁸	-1.06	
10 ⁻²	0.0088176045	-23.6	1.9895995 x 10 ⁵	-29.1	0.138 x 10 ⁻¹
	9.78349083	-78.5	-2.241044599 x 10 ⁸	-18.1	
10 ⁻¹	0.0160334969	38.8	4.4117157 x 10 ⁵	57.2	
	2.9198487	-72.5	-8.9409112 x 10 ⁷	-67.3	0.149

TABLE 29. SIMULATION RESULTS FOR ANALOG IMPLEMENTATION FOR DIFFERENT LEVELS OF INNER PRODUCT ERROR, INTEGRATION TIME = 9.6 μ s, 16 BIT A/D CONVERTER

Inner Product Error (%)	Predicted System Poles (MHz)	Percentage Error	Predicted System Residues	Percentage Error	Normalized Mean Squared Error
0	0.0105703313 12.3080966	-8.5 15.9	2.50092183 x 10 ⁵ -3.0454202296 x 10 ⁸	-10.9 11.3	.217 x 10 ⁻²
10 ⁻⁵	0.0105680356 12.30347489	-8.51 15.88	2.500225174 x 10 ⁵ -3.04398080 x 10 ⁸	-10.9 11.2	0.217 x 10 ⁻²
10 ⁻⁴	0.0105473737 12.2647352	-8.7 15.5	2.493957096 x 10 ⁵ -3.031741244 x 10 ⁸	-11.1 10.77	0.218 x 10 ⁻²
10 ⁻³	0.01032065567 11.9306556	-10.65 12.37	2.42553116 x 10 ⁵ -2.920803 x 10 ⁸	-13.6 6.72	0.259 x 10 ⁻²
10 ⁻²	0.00782470937 10.657726	-32.26 0.384	1.71728479 x 10 ⁵ -2.330022438 x 10 ⁸	-38.8 -14.8	0.228 x 10 ⁻¹
10 ⁻¹	0.0165459431 2.88236707	43.2 -72.8	4.6135407 x 10 ⁵ -8.98437394 x 10 ⁸	64.4 -67.2	0.156

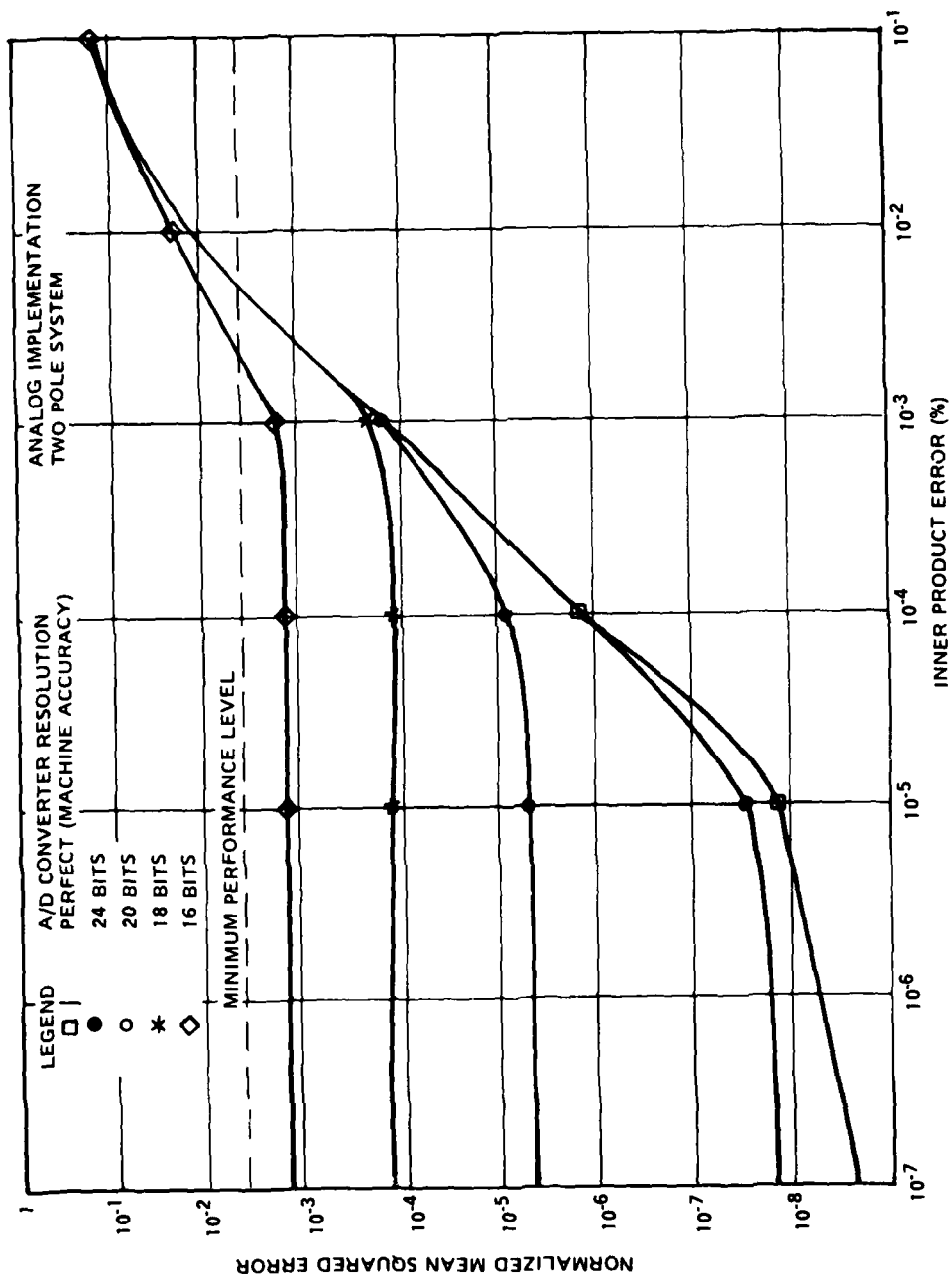


Figure 37. Identification Technique Performance as a Function of Inner Product Error-Analog Implementation

be less than 10^{-3} percent to achieve satisfactory performance for A/D converters of 18 bits or greater resolution. The analog implementation does not achieve satisfactory performance with less than an 18 bit A/D converter.

The problem now is to assess the sources of error in the analog implementation and determine the requirements for each device to meet the total system error requirement. The sources of error in the analog implementation are: (1) the integrator and (2) the inner product device.

The integrator affects the total error in two ways. First, the inputs to some of the inner product devices are the outputs of integrators. Second, the inner product device integrates the product of the two functions input to it. The error introduced by the integrator is examined below.

(1) Integrator Error

The transfer function of a true integrator ($H_{TI}(s)$) is given by

$$H_{TI}(s) = \frac{1}{s} \quad (91)$$

A practical integrator has a transfer function of the form

$$H_{PI}(s) \approx \frac{1}{s + \gamma} \quad (92)$$

A practical way of implementing an integrator is to use an operational amplifier as shown in Figure 38.

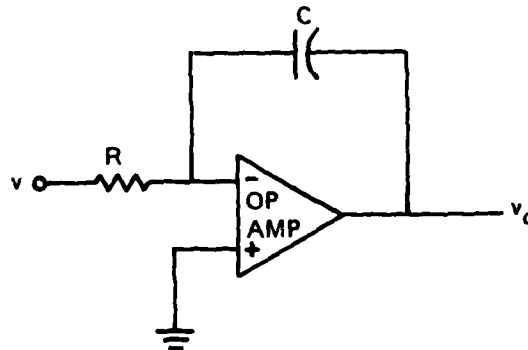


Figure 38. Practical Integrator

The transfer function of this implementation is given by (Reference 8)

$$H_0(s) = A_{v0} \frac{(2\pi f_1)}{RC} \frac{1}{|s + (1/RC A_{v0})| |s + A_{v0} (2\pi f_1)|} \quad (93)$$

where

f_1 is the high frequency pole of the operational amplifier

A_{v0} is the low frequency gain of the operational amplifier

Typically, f_1 is very high (50 to 100 MHz) and A_{v0} is 10^4 to 10^8 . Therefore the pole at $S = -2\pi f_1 A_{v0}$ is on the order of 5×10^{11} Hz. This means that, for systems in the range of 10 to 50 MHz, this pole can effectively be ignored. Then

$$H_0(s) = H_{PI}(s) = \frac{K}{s + \frac{1}{RC A_{v0}}} \quad (94)$$

where

$$\gamma = \frac{1}{RC A_{v0}}$$

and

$$K = \frac{2\pi f_1 A_{v0}}{RC}$$

Now we examine the difference in performance between an ideal and practical integrator. The output of an ideal integrator is given by

$$y_0(t) = \int_0^t x(\tau) d\tau = x(t) * u(t) \quad (95)$$

where $x(t)$ is the input function and $u(t)$ is the unit step function. The output of the ideal integrator is the convolution

of the input and a unit step function. Similarly, the double integral of $x(t)$ is given by

$$\int_0^t \int_0^\beta x(\beta) d\tau d\beta = x(t) * (t u(t)) \quad (96)$$

or the input convolved with a ramp function.

The outputs of a practical integrator are listed below

Single Integration	$x(t) * e^{-\gamma T} u(t)$
Double Integration	$x(t) * t e^{-\gamma T} u(t)$

For a single integration, the error in the integration is a function of the value of $e^{-\gamma T}$, where T is the integration period. This is a measure of the "droop" of the integrator over the integration period. This concept of "droop" also extends to the double integration. The key to reducing the error of a practical integrator is to maintain the product γT as close to zero as possible. For $\gamma T = 1. \times 10^{-13}$, the maximum difference between the true integrator impulse response and that of a practical integrator is given in Table 30 for $T = 9.6 \mu s$.

TABLE 30. PRACTICAL INTEGRATOR ERROR

<u>Integration</u>	<u>Maximum Error Magnitude</u>
Single	0.6×10^{-11}
Double	0.58×10^{-10}

For the two-pole system of interest, the integrator output levels are of the magnitudes listed in Table 31.

TABLE 31. INTEGRATED OUTPUT PEAK VALUES - TWO-POLE SYSTEM

<u>Integrator Output</u>	<u>Peak Value</u>
Single	0.13×10^{-2}
Double	0.4×10^{-2}

The errors introduced in the integrator outputs are almost negligible, and are on the order of 10^{-15} for a single integration and 2×10^{-13} for a double integration. The magnitude of these errors will be compared with those of the inner product device to determine which device is the primary contributor to the total system error.

For $\gamma T = 1 \times 10^{-13}$, it is necessary that

$$\gamma = \frac{1 \times 10^{-13}}{9.6 \times 10^{-6}} = 1.04 \times 10^{-8}$$

where

$$\gamma = \frac{1}{RC A_{v0}} \quad (97)$$

For a typical operational amplifier in use today, A_{v0} varies from 10^4 to 10^{10} . This requires that RC be in the range of 9.6×10^{-3} to 9.6×10^3 . A typical RC selection is $R = 1$ megohm and $C = 1 \mu f$ or $RC = 1$. Achieving $RC = 9600$ requires larger capacitors and larger resistors than are desired. Therefore, an operational amplifier with an A_{v0} of 10^8 to 10^{10} or higher is recommended for use with RC chosen accordingly.

(2) Inner Product Device

The remaining critical component in the analog implementation is the inner product device. One approach to implementing the inner product device is shown in Figure 39.

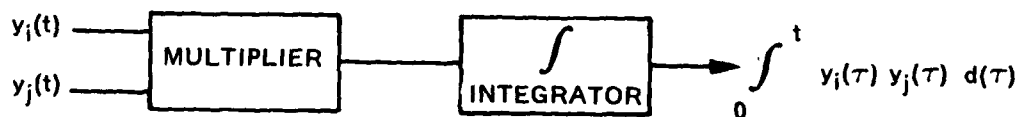


Figure 39. Inner Produce Device

As detailed above, the integrator should not contribute significantly to the error in the inner product. This means that the critical item in the inner product device is the analog multiplier.

The analog multiplier operates conceptually as shown in Figure 40.



Figure 40. Analog Multiplier Model

V_0 is a dimensional constant of a practical multiplier and is typically equal to 10 volts. There are several sources of error in a typical multiplier, some of which can be reduced or eventually eliminated by using external trimming techniques. A detailed discussion of these reducible errors is not provided in this report. However, the irreducible error in an analog multiplier is generally limited by the nonlinearity of the analog multiplier. The nonlinearity specification represents the peak difference between the multiplier output and the theoretical output. The typical range of nonlinearity errors for analog multipliers is in the range of 0.05 to 2 percent. These are specified in terms of full-scale output, i.e., a multiplier with a 0.1 percent nonlinearity error and a 10 volt full-scale maximum output has a maximum nonlinear error of 0.1 volt. The nonlinearity for each input is usually given in multiplier specifications and the nonlinearity error can be conservatively predicted from (Reference 9):

$$f(x,y) = |V_x| \epsilon_x + |V_y| \epsilon_y \quad (98)$$

where

$f(x,y)$ is the nonlinearity error

ϵ_x is the fractional nonlinearity coefficient for x input

ϵ_y is the fractional nonlinearity coefficient for y input

V_x is the x input voltage

V_y is the y input voltage

If V_x and V_y are assumed to have a maximum value of 0.1 volt, the expected product would have a maximum output of $0.1(0.1)/10 = 0.001$ volt.

If the nonlinearity error is to be maintained at less than 0.001 percent, then it is necessary that

$$f(x,y) \leq 0.00001 (V_x V_y) = 0.001 (\epsilon_x + \epsilon_y) \leq 10^{-8} \quad (99)$$

or

$$(\epsilon_x + \epsilon_y) \leq 10^{-5} \quad (100)$$

For $\epsilon_x = \epsilon_y$,

$$\epsilon_x \leq 0.5 \times 10^{-3}\% \quad (101)$$

A survey of commercially available multiplier devices indicates that the best multiplier accuracy that can be achieved is on the order of 0.05 percent. This device does not have sufficient accuracy to meet the requirements above. If the inputs to the multiplier are amplified to a peak of 10 volts, the product maximum will be 10 volts. The nonlinearity error is to be maintained to

$$f(x,y) \leq 10^{-5} (10) = 10^{-4} \quad (102)$$

This requires that

$$(\epsilon_x + \epsilon_y) \leq \frac{10^{-4}}{10} = 10^{-5} \quad (103)$$

so that no advantage is gained by amplifying the signals.

These results imply that the analog implementation is not feasible in the present time frame because of the inaccuracy of the analog multiplier. Multiplier specifications are for the maximum nonlinearity error of the output voltage. They are obtained by setting one input to a constant dc voltage. The other input is varied over the maximum input range and the output voltage measured and compared with the theoretical output. The maximum deviation is then recorded as the nonlinearity error. This complicates the error analysis undertaken in this section because the nonlinearity error specification is a worst case value. In reality, the range of voltages over which the system under test will operate may lie

in a region where the multiplier nonlinearity is much less than the worst case value. This cannot be determined from a multiplier specification sheet; indeed, it is a function of the actual devices to be used in the implementation. A significant amount of device testing would be required before the analog implementation could be considered feasible. The present indications, however, are that the multiplier errors are on the order of 0.05 percent as a minimum. This does not meet the accuracy requirements defined earlier for the inner product.

Another potential drawback for the analog multiplier is the bandwidth limitation. The analog multiplier with the 0.05 percent nonlinearity error is the Analog Devices Model #435K, which has a 3-dB bandwidth of 250 kHz. Wider bandwidth multipliers (1 to 10 MHz) are available at the price of increased nonlinearity error (minimum 0.1 to 0.5 percent).

On the basis of the extrapolation of the amplifier results of III.c, the upper 3-dB break frequency of the multiplier should be 1000 times the upper break frequency of the system under test. This implies that the system under test will be limited to 250 Hz for the 0.05 percent amplifier described above and to 1 to 10 kHz for the less accurate multipliers.

The analog implementation will also require careful consideration of the propagation time delays incurred in the circuitry. The inputs to the various inner product devices will, in many instances, have been processed by a different number of integrators. Any appreciable propagation delay will result in an error being introduced in the product of the two functions input to the analog multiplier. The results of this section clearly indicate that the error in the inner product calculation must be very small in order to obtain satisfactory performance from the identification technique. Therefore, the time delays must be compensated for as much as possible to reduce the overall inner product error. This may require a significant amount of testing prior to using the identification technique to appropriately synchronize the inputs to each inner product device.

A survey of available analog components did not reveal any analog correlator devices capable of forming the required inner product. This implies that the preferred implementation of the inner product device is the analog multiplier-integrator configuration of Figure 39.

These results support the conclusion that the analog implementation is not a feasible implementation for the identification technique at this time. Technology advancements in analog multiplier devices in the areas of increased accuracy

and bandwidth are necessary before the identification technique can be implemented using analog devices.

3. Amplifier, Signal Generator and Data Storage Requirements

The analog implementation requires $2N + 1$ amplifiers to adjust the output of the inner product devices to the full-scale input voltage of the A/D converters. The performance requirements for the amplifier are similar to those determined for the digital implementation. The gain of each amplifier must be set dependent on the inner product peak amplitude of the inner product device output and the A/D converter input characteristics.

The signal generator requirements are the same as those determined for the digital implementation.

The data storage requirements are significantly reduced for the analog implementation. The stored data consists of $4N + 1$ numerical values representing the inner product device outputs. This will permit use of smaller and fewer memory chips than are required for the digital implementation. This data storage requirement does not limit the feasibility or applicability of the identification technique as was discussed in detail in paragraph III.B.4.

4. Conclusions - Analog Implementation

The critical components in the analog implementation are the inner product device and the A/D converter. The analysis of the analog implementation has led to the following conclusions.

- (1) A minimum of 18 bits of A/D converter resolution is required to achieve minimum identification performance.
- (2) The maximum tolerable error in the inner product output is on the order of 10^{-3} percent to achieve a minimum level of identification performance for A/D converters with 18 or more bits of resolution.
- (3) Performance improvement requires less inner product error (10^{-4} to 10^{-5} percent) and increased A/D converter resolution (20 to 24 bits). However, the performance of the analog implementation is below that demonstrated for the digital implementation.

- (4) Currently available analog multipliers have an output error on the order of 0.05 percent which is approximately 50 times greater than the maximum tolerable error of 0.001 percent required for minimum performance of the identification technique.
- (5) The analog multiplier and the A/D converter requirements for the analog implementation imply that it is not feasible to consider this implementation for an experimental test setup in the present time frame. Significant technological developments for analog multipliers and A/D converters are necessary before this implementation can prove feasible.

SECTION IV

REFERENCES

1. E. Ewen, "Black Box Identification of Nonlinear Volterra Systems," PhD Dissertation, Syracuse University, Dec. 1975. Also published as TIS R77EML7, General Electric Company, Aerospace Electronic Systems Department, Mar. 1977.
2. J. Bussgang, L. Ehrman, and J. Graham, "Analysis of Nonlinear Systems with Multiple Inputs," Proc. IEEE, 62, pp 1088-1119, Aug. 1974.
3. D. George, "Continuous Nonlinear Systems," M.I.T. Research Laboratory of Electronics, Technical Report 355, 24 July 1959.
4. V. Jain, "Filter Analysis by Use of Pencil of Functions, Part I," IEEE Trans. Circuits and Systems, CAS-21, pp 574-579, Sept. 1974.
5. E. Kreyszig, Advanced Engineering Mathematics, New York: J. Wiley & Sons, Inc., 1972.
6. V. Jain and J. Osman, "Computer Routines For Use in Second-Order Volterra Identification of EMI," University of South Florida, October 1978.
7. D. Sheingold, et. al., Analog-Digital Conversion Handbook, Analog Devices, Inc. 1976.
8. J. Millman and C. Halkias, Integrated Electronics: Analog and Digital Circuits and Systems, New York: McGraw-Hill Book Co., 1972.
9. D. Sheingold, et. al., Nonlinear Circuits Handbook, Analog Devices, Inc. 1976.

APPENDIX A

COMPUTER PROGRAM LISTING FOR THE DIGITAL IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE

A listing of the computer program for the digital implementation is provided below for the two-pole system of Section III.B.

```

100     DIMENSION Y1(1,2401)
110     DIMENSION A1(4),CP(4),C1(4),AP(4)
120     DIMENSION G(10,10),Y(3,2401),U(3,2401),A(20,10),C(25,25)
130     DIMENSION ISN(10),XLX(10),COEF(10),XLAMR(10),XLAMC(10)
140     DIMENSION RR(10),CR(10),B(25),CRR(10,10),CI(10,10)
150     DIMENSION LIMIT1(10),LABEL(25),S(25,1),R(25,1)
160     INTEGER SN,STEP1,STEP2
170C    INITIALIZATION AND INPUT PARAMETERS
180     PI2=2.*3.14159
190     ERMS=0.
200     W3=10.
210     A1(3)=-1.E6*W3
220     AP(3)=A1(3)
230     DELT=4.E-3
240     SN=2
250     PRINT:"ORDER OF LINEAR SYSTEM IS",SN
260     PRINT:""
270     SN2=2*SN
280     SK=1.
290     M=2400
300     MORIG=M
310     LIMIT1(1)=M+1
320     STEP=SN+1
330     T=DELT*M
340     LL=0
350     XMSBI=1.E-3
360     XMSBO=1.5E-3
370     NBITS=0
380     PRINT:"NUMBER OF BITS IN A/D=",NBITS
390     PRINT:""
400     PRINT:"NUMBER OF WAVEFORM SAMPLES=",M
410     PRINT:""
420     PRINT:"INTEGRATION TIME IN MICROSECONDS=",T
430     PRINT:""
440     PRINT:"INTEGRATION INTERVAL IN MICROSECONDS=",DELT
450     PRINT:""
460     R3=1.E-3
470     CP(3)=R3
480 900 FORMAT(13X,E14.5)
490C    EVALUATE INPUT AND OUTPUT FUNCTIONS
500     R1=2.8069192E5
510     R2=-2.7368441E8
520     W1=PI2*.011550998
530     A1(1)=-1.E6*W1
540     W2=PI2*10.616986
550     A1(2)=-1.E6*W2
560     CA=R1*R3/((W3-W1)*1.E6)
570     C2=R2*R3/((W3-W2)*1.E6)

```

```

580      C3=R2*R3/((W2-W3)*1.E6)
590      C3=C3+R1*R3/((W1-W3)*1.E6)
600      C1(1)=-CA
610      C1(2)=-C2
620      C1(3)=-C3
630      DO 999 JK=1,M+1
640      XJK=JK
650      U(1,JK)=0.
660      XT3A=-W3*(XJK-1)*DELT
670      IF(XT3A.LT.-80.) GO TO 166
680      U(1,JK)=EXP(-W3*(XJK-1)*DELT)
690 166 CONTINUE
700      U(1,JK)=1.E-3*SK*U(1,JK)
710      CALL ATOD(U(1,JK),XMSBI,NBITS)
720      XT1=CA*EXP(-W1*(XJK-1)*DELT)
730      XT2=0.
740      XT3=0.
750      XT2A=-W2*(XJK-1)*DELT
760      XT3A=-W3*(XJK-1)*DELT
770      IF(XT3A.GT.-80.) XT3=C3*EXP(XT3A)
780      IF(XT2A.GT.-80.) XT2=C2*EXP(XT2A)
790      Y(1,JK)=XT1+XT2+XT3
800      Y(1,JK)=SK*Y(1,JK)
810      Y1(1,JK)=Y(1,JK)
820      CALL ATOD(Y(1,JK),XMSBO,NBITS)
830 999 CONTINUE
840      PRINT:"ACTUAL SYSTEM POLES ARE:",W1/PI2,W2/PI2
850      PRINT:""
860      PRINT:"ACTUAL SYSTEM RESIDUES ARE:",R1,R2
870      PRINT:""
880      M2=M
890      PRINT:"INPUT MSB=",XMSBI
900      PRINT:""
910      PRINT:"OUTPUT MSB=",XMSBO
920      PRINT:""
930      DO 138 JJ=2,SN+1
940C      INTEGRATE INPUT FUNCTION SN TIMES USING SIMPSONS RULE
950      LL=0
960      U(JJ,1)=0.
970      DO 137 II=3,M2+1,2
980      LL=LL+1
990      KK=II-LL
1000      TED1=U(JJ-1,II-2)+4.*U(JJ-1,II-1)+U(JJ-1,II)
1010      U(JJ,KK)=U(JJ,KK-1)+T/(3*M2)*TED1
1020 137 CONTINUE
1030      M2=M2/2
1040 138 CONTINUE
1050C      INTEGRATE OUTPUT WAVEFORM SN TIMES USING SIMPSONS RULE
1060      DO 101 JK=2,STEP

```

```

1070 120 LIMIT=M-1
1080     JAT=0
1090     LIMIT1(JK)=M/2+1
1100     CONST=T/(3.*M)
1110 140 J=0
1120     Y(JK,1)=0.
1130     LIM=LIMIT+2
1140     DO 13 I=3,LIM,2
1150     J=J+1
1160     KK=I-J
1170     TERM52=Y(JK-1,I-2)+4.*Y(JK-1,I-1)+Y(JK-1,I)
1180     Y(JK,KK)=Y(JK,KK-1)+CONST*TERM52
1190 13 CONTINUE
1200     M=M/2
1210 38 CONTINUE
1220 101 CONTINUE
1230C     EVALUATE INNER PRODUCTS FOR GRAM DETERMINANT
1240     M=MORIG
1250     STEP1=SN+1
1260     STEP=2*SN+1
1270     DO 100 K=1,STEP1
1280     LJ=1
1290     M1=M
1300     MSLJ=1
1310 220 LIMIT=M-2
1320     JAT=0
1330     CONST=T/(3.*M)
1340 240 CONTINUE
1350C     EVALUATE G(K,K) FOR K=1,SN
1360     DO 7 I=1,LIMIT1(K)-2,2
1370     XI=1
1380     TERM1=2**FLOAT(K-1)*CONST
1390     TERM12=Y(K,I)**2+4.*Y(K,I+1)**2+Y(K,I+2)**2
1400     TERM13=TERM1*TERM12
1410     G(K,K)=G(K,K)+TERM13
1420 7 CONTINUE
1430C     EVALUATE G(K,KM) FOR J .GT. K TO J=SN+1
1440     DO 3 J=2,STEP1
1450     MSLJ=1
1460     IF(K-J) 435,437,436
1470 435 DO 6 I=1,LIMIT1(J)-2,2
1480     XTER2=2**FLOAT(J-1)*CONST
1490     TERM22=Y(K,MSLJ)*U(J,I)
1500     HSLJ=MSLJ+2**FLOAT(J-K)
1510     TERM23=4.*Y(K,HSLJ)*U(J,I+1)
1520     KSLJ=MSLJ+2**FLOAT(J-K+1)
1530     TERM24=Y(K,KSLJ)*U(J,I+2)
1540     TERM2=XTER2*(TERM22+TERM23+TERM24)
1550     KM=J+SN
1560     G(K,KM)=G(K,KM)+TERM2
1570     MSLJ=KSLJ

```

```

1580 6 CONTINUE
1590 GO TO 3
1600C EVALUATE G(K,KM) FOR K .GT. J TO K=SN+1
1610 436 DO 176 I=1,LIMIT1(K)-2,2
1620 XTER2=2**FLOAT(K-1)*CONST
1630 TERM22=Y(K,I)*U(J,MSLJ)
1640 HSLJ=MSLJ+2**FLOAT(K-J)
1650 TERM23=4.*Y(K,I+1)*U(J,HSLJ)
1660 KSLJ=MSLJ+2**FLOAT(K-J+1)
1670 TERM24=Y(K,I+2)*U(J,KSLJ)
1680 TERM2=XTER2*(TERM22+TERM23+TERM24)
1690 KM=J+SN
1700 G(K,KM)=G(K,KM)+TERM2
1710 MSLJ=KSLJ
1720 176 CONTINUE
1730 GO TO 3
1740C EVALUATE G(K,KM) FOR K=J
1750 437 DO 276 I=1,LIMIT1(K)-2,2
1760 TERM22=Y(K,I)*U(J,I)
1770 TERM23=4.*Y(K,I+1)*U(J,I+1)
1780 TERM24=Y(K,I+2)*U(J,I+2)
1790 XTER2=2**FLOAT(K-1)*CONST
1800 TERM2=XTER2*(TERM22+TERM23+TERM24)
1810 KM=J+SN
1820 G(K,KM)=G(K,KM)+TERM2
1830 276 CONTINUE
1840 3 CONTINUE
1850 IF(K-STEP1) 236,100,100
1860 100 CONTINUE
1870C EVALUATE G(K,K) FOR K=SN+1 TO 2*SN+1
1880 DO 8 K=SN+2,SN2+1
1890 DO 9 I=1,LIMIT1(K-SN)-2,2
1900 DER1=2**FLOAT(K-SN-1)*T/(3*M)
1910 DER2=U(K-SN,I)**2+4.*U(K-SN,I+1)**2+U(K-SN,I+2)**2
1920 DER3=DER1*DER2
1930 G(K,K)=G(K,K)+DER3
1940 9 CONTINUE
1950 8 CONTINUE
1960 GO TO 425
1970 236 IF(K-SN) 237,237,100
1980C EVALUATE G(K,LK) FOR K .LT. SN, LK=K+1,SN+1
1990 237 DO 401 LK=K+1,SN+1
2000 MMW=LIMIT1(LK)-2
2010 LJ=1
2020 DO 402 LI=1,MMW,2
2030 TERM31=Y(K,LJ)*Y(LK,LI)
2040 KH1=LJ+2**FLOAT(LK-K)
2050 TERM32=Y(K,KH1)*Y(LK,LI+1)
2060 KH2=LJ+2**FLOAT(LK-K+1)
2070 TERM33=Y(K,KH2)*Y(LK,LI+2)
2080 TERM34=TERM1*2**FLOAT(LK-K)

```

```

2090      LJ=LJ+2**FLOAT(LK-K+1)
2100      TERM3=TERM34*(TERM31+4.*TERM32+TERM33)
2110      G(K,LK)=G(K,LK)+TERM3
2120 402 CONTINUE
2130 401 CONTINUE
2140 238 IF(K-1) 57,57,58
2150C     EVALUATE G(K+SN,LK+SN) FOR K=2,SN+1, LK=3,SN+1
2160 58 DO 601 LK=K+1,SN+1
2170      MMW=LIMIT1(LK)-2
2180      LJ=1
2190      DO 602 LI=1,MMW,2
2200      TEX1=U(K,LJ)*U(LK,LI)
2210      KL1=LJ+2**FLOAT(LK-K)
2220      TEX2=U(K,KL1)*U(LK,LI+1)
2230      KL2=LJ+2**FLOAT(LK-K+1)
2240      TEX3=U(K,KL2)*U(LK,LI+2)
2250      TEX4=TERM41*2**FLOAT(LK-K)
2260      LJ=LJ+2**FLOAT(LK-K+1)
2270      TEX5=TEX4*(TEX1+4.*TEX2+TEX3)
2280      G(K+SN,LK+SN)=G(K+SN,LK+SN)+TEX5
2290 602 CONTINUE
2300 601 CONTINUE
2310 57 CONTINUE
2320      GO TO 100
2330 425 CONTINUE
2340      PRINT:"UNSCALED ENTRIES IN GRAM DETERMINANT ARE"
2350          PRINT:"          ROW          COLUMN          INNER PRODUCT"
2360          PRINT:"          I              J              G(I,J)"
2370          PRINT:""
2380          STEPI=STEPI+SN
2390          DO 942 JIK=1,STEPI
2400          DO 943 KLI=1,STEPI
2410          G(KLI,JIK)=G(JIK,KLI)
2420          PRINT:JIK,KLI,G(JIK,KLI)
2430 943 CONTINUE
2440 942 CONTINUE
2450          MORIG=M
2460          N=SN2
2470C     SCALE SCALAR PRODUCTS BY 1.E6 FOR COMPUTATION FACILITY
2480          DO 1011 I=1,SN2+1
2490          DO 1021 J=1,SN2+1
2500          G(I,J)=1.E6*G(I,J)
2510 1021 CONTINUE
2520 1011 CONTINUE
2530C     EVALUATE DIAGONAL COFACTORS COEF(I)
2540          PRINT:"DIAGONAL COFACTORS ARE AS FOLLOWS"
2550          DO 300 J=1,SN2
2560          DO 310 I=1,SN2
2570          A(J,I)=G(J+1,I+1)
2580          A(I,J)=A(J,I)
2590 310 CONTINUE

```

```

2600 300 CONTINUE
2610 ID=1
2620 COEF(1)=DETE(A,N,20)
2630 PRINT:ID,COEF(1)
2640 DO 500 LKJ=1,SN
2650 DO 400 J=1,LKJ
2660 DO 410 I=1,LKJ
2670 A(J,I)=G(J,I)
2680 A(I,J)=G(I,J)
2690 410 CONTINUE
2700 400 CONTINUE
2710 DO 510 J=1,LKJ
2720 DO 520 I=LKJ+1,SN2
2730 A(J,I)=G(J,I+1)
2740 A(I,J)=A(J,I)
2750 520 CONTINUE
2760 510 CONTINUE
2770 DO 600 J=LKJ+1,SN2
2780 DO 610 I=LKJ+1,SN2
2790 A(J,I)=G(J+1,I+1)
2800 A(I,J)=A(J,I)
2810 610 CONTINUE
2820 600 CONTINUE
2830 COEF(LKJ+1)=DETE(A,N,20)
2840 PRINT:LKJ+1,COEF(LKJ+1)
2850 LNM=LKJ+1
2860 500 CONTINUE
2870 PRINT:"EIGENVALUE EQUATION COEFFICIENTS ARE"
2880C EVALUATE EIGENVALUE EQUATION COEFFICIENTS B(I)
2890 DO 640 I=1,SN+1
2900 B(I)=SQRT(ABS(COEF(SN+2-I)))
2910 PRINT:I,B(I)
2920 640 CONTINUE
2930C EVALUATE SYSTEM POLES
2940 N=SN
2950 PRINT:"POLES OF SYSTEM ARE GIVEN BELOW"
2960 PRINT:"          NUMBER    REAL(MHZ)    IMAG(MHZ)"
2970 CALL DOJNH(B,N,RR,CR)
2980 DO 650 J=1,SN
2990 XLAMR(J)=RR(J)/PI2
3000 XLAMC(J)=CR(J)/PI2
3010 AP(J)=RR(J)*1.E6
3020 PRINT:J,XLAMR(J),XLAMC(J)
3030 XLAMR(J)=RR(J)
3040 650 CONTINUE
3050C EVALUATE SYSTEM RESIDUES
3060 FLAG=1
3070 DO 660 K=1,SN+1
3080 IF(XLAMC(J)) 670,660,670
3090 670 FLAG=0
3100 660 CONTINUE

```

```

3110 IF(FLAG) 680,680,690
3120 590 DO 700 I=1,SN
3130 DO 710 J=1,SN
3140 TEMP1=1
3150 DO 711 L=1,I
3160 TEMP1=TEMP1*XLAMR(J)
3170 711 CONTINUE
3180 EWE1=0.
3190 EWE2=0.
3200 EWE=1./(TEMP1*(XLAMR(J)+W3))
3210 EWE1A=XLAMR(J)*T
3220 EWE2A=-W3*T
3230 IF(EWE2A.GT.-80.) EWE2=EXP(EWE2A)
3240 IF(EWE1A.GT.-80.) EWE1=EXP(EWE1A)
3250 C(I,J)=EWE*(EWE1-EWE2)
3260 C(I,J)=1.E-3*SK*C(I,J)
3270 TEMP=0.
3280 DO 720 K=1,I
3290 TEMP2=1.
3300 DO 721 KK=1,I+1-K
3310 TEMP2=TEMP2*XLAMR(J)
3320 721 CONTINUE
3330 TEMP=TEMP+U(K+1,LIMIT1(K+1))/TEMP2
3340 720 CONTINUE
3350 C(I,J)=C(I,J)-TEMP
3360 710 CONTINUE
3370 700 CONTINUE
3380 CALL MTINV(C,N,N,25,LABEL)
3390 DO 740 I=2,SN+1
3400 S(I-1,1)=Y(I,LIMIT1(I))
3410 740 CONTINUE
3420 CALL MTMPY(O,C,S,R,SN,SN,1)
3430 PRINT:"RESIDUES OF SYSTEM POLES ARE GIVEN BELOW"
3440 DO 751 I=1,SN
3450 PRINT:I,R(I,1)*1.E6
3460 751 CONTINUE
3470 GO TO 1000
3480 680 CONTINUE
3490 1000 CONTINUE
3491 X1=100.*(R(1,1)*1.E6-R1)/R1
3492 X2=100.*(R(2,1)*1.E6-R2)/R2
3493 PRINT:""
3494 PRINT:"PERCENTAGE ERROR-RESIDUES",X1,X2
3495 PRINT:""
3496 X1=100.*(AP(1)-A1(1))/A1(1)
3497 X2=100.*(AP(2)-A1(2))/A1(2)
3498 PRINT:"PERCENTAGE ERROR-POLES",X1,X2
3499 PRINT:""
3500 PRINT:""
3510 PRINT:" JK ACTUAL PREDICTED DIFFERENCE"
3520 SUM=0.

```

```

3530 CP(C,1)=R(C,1)*R3/(C-W3-RR(C,1))
3540 CP(C,2)=R(C,2)*R3/(C-W3-RR(C,1))
3550 CP(C,3)=R(C,1)*R3/(RR(C,1)+W3)+R(C,1)*R3/(RR(C,1)+W3)
3560 DO 247 J,K=1,MORTG+1
3570 X(J,K)=J
3580 XT11=CP(C,1)*EXP(RR(C,1)*(X(J,K)-1)*DELT)
3590 XT22=0.
3600 XT33=0.
3610 XT22A=RR(C,1)*(X(J,K)-1)*DELT
3620 XT33A=-W3*(X(J,K)-1)*DELT
3630 IF (XT33A.GT.-80.) XT33=CP(C,3)*EXP(XT33A)
3640 IF (XT22A.GT.-80.) XT22=CP(C,2)*EXP(XT22A)
3650 YOUT=-1.*(XT11+XT22+XT33)*SK
3660 DELOUT=YOUT-Y1(C,J,K)
3670 LJK=LJK+1
3680 IF (LJK.GE.100) GO TO 248
3690 GO TO 249
3700 248 LJK=0
3710 PRINT*,J,K,Y1(C,J,K),YOUT,DELOUT
3720 249 CONTINUE
3730 247 CONTINUE
3732 SUM1=0.
3740 T=T*1.,T=0.
3750 DO 242 IX=1,SN+1
3760 DO 242 JX=1,SN+1
3770 IF ((CAP(C,IX)+AP(C,JX))*T.I.T.-90.) GO TO 243
3780 SUM=SUM+CP(C,IX)*CP(C,JX)/(AP(C,IX)+AP(C,JX))*(EXP((CAP(C,IX)+AP(C,JX))*T))
3790 243 IF ((CAP(C,IX)+A1(C,JX))*T.I.T.-90.) GO TO 244
3800 SUM=SUM+.2.*CP(C,IX)*C1(C,JX)/(AP(C,IX)+A1(C,JX))*(EXP((CAP(C,IX)+A1(C,JX))*T))
3810 244 IF ((A1(C,IX)+A1(C,JX))*T.I.T.-90.) GO TO 246
3820 SUM=SUM+C1(C,IX)*C1(C,JX)/(A1(C,IX)+A1(C,JX))*(EXP((A1(C,IX)+A1(C,JX))*T))
3825 SUM1=SUM1+C1(C,IX)*C1(C,JX)/(A1(C,IX)+A1(C,JX))*(EXP((A1(C,IX)+A1(C,JX))*T))
3830 245 SUM=SUM-CP(C,IX)*CP(C,JX)/(AP(C,IX)+AP(C,JX))
3832 SUM=SUM+.2.*CP(C,IX)*C1(C,JX)/(A1(C,IX)+AP(C,IX))
3833 SUM=SUM-C1(C,IX)*C1(C,JX)/(A1(C,IX)+A1(C,JX))
3834 SUM1=SUM1-C1(C,IX)*C1(C,JX)/(A1(C,IX)+A1(C,JX))
3835 242 CONTINUE
3840 TSUM=SUM/T
3850 PRINT, ""
3860 PRINT, "MEAN SQUARED ERROR=", TSUM
3870 PRINT, ""
3872 TFORM=SUM1/T
3875 PRINT, "NORM=", TFORM
3876 PRINT, ""
3880 XMSF=TSUM/TFORM
3881 PRINT, "NORMALIZED MSF=", XMSF
3890 STOP
3900 END
3910 DEFT
3920 *****DETERMINANT EVALUATION*****

```

```

3930 FUNCTION DETE(A,NARG,IDIM)
3940 DIMENSION A(IDIM,NARG)
3950 I N=NARG
3960 SIGN=1.0
3970 NMINI=N-1
3980 IF(NMINI .EQ. 0) GO TO 401
3990 DO 391 J1 = 1,NMINI
4000*      *****FIND REMAINING ROW CONTAINING LARGEXT*****
4010*      *****ABSOLUTE VALUE IN PIVOTAL COLUMN*****
4020 101 TEMP=0.0
4030 DO 121 J2=J1,N
4040 IF(ABS(A(J2,J1)).LT.TEMP) GO TO 121
4050 TEMP=ABS(A(J2,J1))
4060 IBIG=J2
4070 121 CONTINUE
4080 IF(TEMP.NE.0.0) GO TO 201
4090*      *****PIVOTAL COLUMN CONTAINS ALL ZEROES*****
4100 DETE=0.0
4110 GO TO 5001
4120 201 IF(J1.EQ.IBIG) GO TO 301
4130*      *****INTERCHANGE ROWS AND CHANGE SIGN*****
4140 DO 221 J2=J1,N
4150 TEMP=A(J1,J2)
4160 A(J1,J2)=A(IBIG,J2)
4170 221 A(IBIG,J2)=TEMP
4180 SIGN=-SIGN
4190*      *****COMPUTE NEW COEFFICIENTS BELOW PIVOTAL R
4200*      *****AND BEYOND PIVOTAL COLUMN*****
4210 301 N1=J1+1
4220 DO 321 J2=N1,N
4230 TEMP=A(J2,J1)/A(J1,J1)
4240 DO 321 J3=N1,N
4250 321 A(J2,J3)=A(J2,J3)-A(J1,J3)*TEMP
4260 391 CONTINUE
4270*      *****ELEMENTS TIMES (-1)**NO. OF ROW INTERCHA
4280 401 DETE=1.0
4290 DO 421 J1=1,N
4300 421 DETE=DETE*A(J1,J1)
4310 DETE=DETE*SIGN
4320 5001 RETURN
4330 END
4340* ZORP2
4350* ROUTINES FOR SOLVING POLYNOMIALS
4360      SUBROUTINE POLY(N,A,R,C,PR,PC,RHO,PHI)
4370      DIMENSION A(9999)
4380      IF(RHO)10,5,10
4390 5      R=A(1)
4400      C=0.
4410      PR=A(2)
4420      PC=0.
4430      RETURN

```

```

4440 10 V1=1.
4450 V2=0.
4460 R=A(1)
4470 C=0.
4480 PR=0.
4490 PC=0.
4500 W1=RHO*COS(PHI)
4510 W2=RHO*SIN(PHI)
4520 NN=N+1
4530 DO 20 I=2,NN
4540 T1=W1*V1-W2*V2
4550 V2=W2*V1+W1*V2
4560 V1=T1
4570 R=R+A(I)*V1
4580 C=C+A(I)*V2
4590 PR=PR+A(I)*(I-1)*V1
4600 20 PC=PC+A(I)*(I-1)*V2
4610 PR=PR/RHO
4620 PC=PC/RHO
4630 5001 RETURN
4640 END
4650 SUBROUTINE ARCTA(X,Y,ANGLE)
4660 PI=3.14159265
4670 IF(X)10,30,20
4680 10 ANGLE=ATAN(Y/X)+PI*SIGN(1.,Y)
4690 RETURN
4700 20 ANGLE=ATAN(Y/X)
4710 RETURN
4720 30 IF(Y)40,60,50
4730 40 ANGLE=-PI/2.
4740 RETURN
4750 50 ANGLE=PI/2.
4760 RETURN
4770 60 ANGLE=0.
4780 RETURN
4790 END
4800 SUBROUTINE DOWNH(A,NAR,RR,CR)
4810 DIMENSION A(9999),RR(9999),CR(9999),Q(101),B(3)
4820 CALL FXOPT(67,1,1,0)
4830 J=0
4840 N=NAR
4850 NPL1=N+1
4860 ANPP=A(NPL1)
4870 DO 102 I=1,NPL1
4880 IF (A(I))103,102,103
4890 102 CONTINUE
4900 103 C=ABS(A(I)/A(NPL1))
4910 LU=120
4920 LL=-120
4930 IF(C-2.**LU)100,100,101
4940 100 IF(C-2.**LL)101,105,105

```

```

4950 101 NAR=-NAR
4960 GO TO 5001
4970 105 II=(LU+LL)/2
4980 IF(C-2.**II)110,110,109
4990 109 LL=II
5000 GO TO 111
5010 110 LU=II
5020 111 IF(LU-LL-1)5001,112,105
5030 112 IB=II/N
5040 IF(IB)114,120,114
5050 114 DO 115 I=1,NPL1
5060 II=I-1
5070 115 A(I)=A(I)*(2.**(II*IB))
5080 120 DO 121 J=1,NPL1
5090 121 A(J)=A(J)/A(NPL1)
5100 201 IF(N)2001,2001,206
5110 206 IF(A(1))301,211,301
5120 211 J=J+1
5130 RR(J)=0.
5140 CR(J)=0.
5150 DO 221 J1=1,N
5160 221 A(J1)=A(J1+1)
5170 N=N-1
5180 GO TO 201
5190 301 IF(N-2)601,501,401
5200 401 CALL GRAD(A,N,X,Y)
5210 421 IF(ABS(Y)-ABS(X*1.E-4))431,431,441
5220 431 Y=0.
5230 441 J=J+1
5240 RR(J)=X
5250 CR(J)=Y
5260 IF(Y)461,1021,461
5270 461 J=J+1
5280 RR(J)=X
5290 CR(J)=-Y
5300 GO TO 1011
5310 501 DISC=A(2)**2-4.*A(1)
5320 IF(DISC)521,541,541
5330 521 Y=SQRT(-DISC)/2.
5340 X=-A(2)/2.
5350 GO TO 421
5360 541 J=J+1
5370 RR(J)=(-A(2)+SQRT(DISC))/2.
5380 CR(J)=0.
5390 GO TO 1021
5400 601 J=J+1
5410 RR(J)=-A(1)
5420 CR(J)=0.
5430 GO TO 2001
5440 1011 B(1)=X**2+Y**2
5450 B(2)=-2.*X

```

```

5460      B(3)=1.
5470      NB=2
5480      GO TO 1041
5490 1021  B(1)=-RR(J)
5500      B(2)=1.
5510      NB=1
5520 1041  CALL DIV(A,B,N,NB,Q)
5530      DO 1061 J1=1,N
5540 1061  A(J1)=Q(J1)
5550      IF(CR(J))1081,1071,1081
5560 1071  N=N-1
5570      GO TO 201
5580 1081  N=N-2
5590      GO TO 201
5600 2001  IF(IB)2002,2005,2002
5610 2002  DO 2000 I=1,NAR
5620      RR(I)=RR(I)*(2.**IB)
5630 2000  CR(I)=CR(I)*(2.**IB)
5640 2005  NPI=NAR+1
5650      DO 2011 I=2,NPI
5660 2011  A(I)=0.
5670      A(1)=1.
5680      NA=0
5690      J=1
5700 2021  IF(CR(J))2041,2061,2041
5710 2041  NB=2
5720      B(3)=1.
5730      B(2)=-2.*RR(J)
5740      B(1)=RR(J)**2+CR(J)**2
5750      J=J+2
5760      GO TO 2081
5770 2061  NB=1
5780      B(2)=1.
5790      B(1)=-RR(J)
5800      J=J+1
5810 2081  CALL MTALGD(A,NA,B,NB,Q)
5820      NA=NB+NA
5830      NAPL1=NA+1
5840      DO 2091 I=1,NAPL1
5850 2091  A(I)=Q(I)
5860      IF(NA-NAR) 2021, 3001,3001
5870 3001  DO 3011 J2=1,NPL1
5880 3011  A(J2)=A(J2)*ANPP
5890 5001  RETURN
5900      END
5910      SUBROUTINE GRAD(A,N,XZ,YZ)
5920      DIMENSION A(9999),X(3),Y(3),RP(3),CP(3),RH0(3),PHI(3)
5930      DIMENSION ABSP(3),PR(3),PC(3)
5940      PI=3.14159265
5950      MTST=1
5960 101    XZ=0.0
5970      YZ=1.0

```

```

5980      DZ=2.
5990      RHOZ=1.
6000      PHIZ=PI/2.
6010 201  CALL POLY(N,A,RZ,CZ,PRZ,PCZ,RHOZ,PHIZ)
6020 221  SU=SQRT(PRZ**2+PCZ**2)
6030      ABSPZ=SQRT(RZ**2+CZ**2)
6040      U=2.*ABSPZ*SU
6050      PSI=ATAN(U)
6060      TOP=RZ*PCZ-CZ*PRZ
6070      BOT=-(RZ*PRZ+CZ*PCZ)
6080      CALL ARCTA(BOT,TOP,THETA)
6090      COSI=COS(THETA+PHIZ)
6100      SINE=SIN(THETA+PHIZ)
6110      IF(ABSPZ)300,5001,300
6120 300  IF(SU)301,501,301
6130 301  IF(RHOZ)321,401,321
6140 321  IF(ABSPZ/(RHOZ*SU)-1.E-7)5001,5001,701
6150 351  IF(ABSPZ/(RHOZ*SU)-10.**(-MTST))801,801,401
6160 401  DZ=DZ/8.0
6170      IM=0
6180      DO 431 I=1,3
6190      DZ=2.*DZ
6200      X(I)=XZ+DZ*COSI
6210      Y(I)=YZ+DZ*SINE
6220      RHO(I)=SQRT(X(I)**2+Y(I)**2)
6230      CALL ARCTA(X(I),Y(I),PHI(I))
6240      CALL POLY(N,A,RP(I),CP(I),PR(I),PC(I),RHO(I),PHI(I))
6250      ABSP(I)=SQRT(RP(I)**2+CP(I)**2)
6260      IF(ABSPZ-ABSP(I)) 431,431,421
6270 421  ABSPZ=ABSP(I)
6280      IM=I
6290 431  CONTINUE
6300      IF(IM) 441,441,461
6310 441  DZ=DZ/8.
6320      IF(RHOZ)443,445,443
6330 443  IF(DZ/RHOZ-1.E-7)451,451,401
6340 445  IF(DZ-1.E-7)451,451,401
6350 451  IF(SU-ABSPZ) 501,501,5001
6360 461  DZ=(2.** (IM-2))*DZ
6370      XZ=X(IM)
6380      YZ=Y(IM)
6390      PHIZ=PHI(IM)
6400      PRZ=PR(IM)
6410      PCZ=PC(IM)
6420      RHOZ=RHO(IM)
6430      RZ=RP(IM)
6440      CZ=CP(IM)
6450      GO TO 221
6460 501  DZ=1.0
6470      DTHETA=PI/10.
6480 521  THETA=0.0

```

```

6490      DO 561 I=1,20
6500      THETA=THETA+DTHETA
6510      XS=XZ+DZ*COS(PHIZ+THETA)
6520      YS=YZ+DZ*SIN(PHIZ+THETA)
6530      RHOS=SQRT(XS**2+YS**2)
6540      CALL ARCTA(XS,YS,PHIS)
6550      CALL POLY(N,A,RS,CS,PRS,PCS,RHOS,PHIS)
6560      ABSP(1)=SQRT(RS**2+CS**2)
6570      IF(ABSPZ-ABSP(1))561,561,601
6580 561  CONTINUE
6590      DZ=DZ/2.
6600      IF(RHOS)563,565,563
6610 563  IF(DZ/RHOS-1.E-7)5001,5001,521
6620 565  IF(DZ-1.E-7)5001,5001,521
6630 601  XZ=XS
6640      YZ=YS
6650      PHIZ=PHIS
6660      RHOZ=RHOS
6670      ABSPZ=ABSP(1)
6680      PRZ=PRS
6690      PCZ=PCS
6700      RZ=RS
6710      CZ=CS
6720      GO TO 221
6730 701  IF(PSI-1.E-6)711,711,351
6740 711  IF(SU-ABSPZ)501,501,351
6750 801  RHO(1)=RHOZ+BOT/SU**2
6760      IF(RHO(1))901,901,816
6770 816  PHI(1)=PHIZ+TOP/(RHOZ*SU**2)
6780 821  CALL POLY(N,A,RZ,CZ,PRZ,PCZ,RHO(1),PHI(1))
6790      ABSP(1)=SQRT(RZ**2+CZ**2)
6800      IF(ABSP(1)-ABSPZ)851,881,881
6810 841  XZ=RHOZ*COS(PHIZ)
6820      YZ=RHOZ*SIN(PHIZ)
6830      GO TO 5001
6840 851  RHOZ=RHO(1)
6850      ABSPZ=ABSP(1)
6860      PHIZ=PHI(1)
6870      TOP=RZ*PCZ-CZ*PRZ
6880      BOT=-(RZ*PRZ+CZ*PCZ)
6890      SU=SQRT(PRZ**2+PCZ**2)
6900      IF(SU)855,501,855
6910 855  U=2.*ABSPZ*SU
6920      PSI=ATAN(U)
6930      IF(ABSPZ/(RHOZ*SU)-10.**(-MTST))861,861,901
6940 861  IF(ABSPZ/(RHOZ*SU)-1.E-7)841,841,871
6950 871  IF(PSI-1.E-6)881,881,801
6960 881  IF(SU-ABSPZ)501,501,901
6970 901  DZ=ABSPZ/SU
6980      XZ=RHOZ*COS(PHIZ)
6990      YZ=RHOZ*SIN(PHIZ)

```

```

7000      MTST=MTST+1
7010      GO TO 201
7020 5001  RETURN
7030      END
7040      SUBROUTINE MTALGD(AARG,NA,BARG,NB,C)
7050      DIMENSION AARG(9999),BARG(9999),C(9999),A(101),B(101)
7060 1      NAPLI=NA+1
7070      DO 21 J1=1,NAPLI
7080 21      A(J1)=AARG(J1)
7090      NBPLI=NB+1
7100      DO 41 J1=1,NBPLI
7110 41      B(J1)=BARG(J1)
7120      NCPLI=NAPLI+NBPLI-1
7130      DO 91 J1=1,NCPLI
7140      TEMP=0.
7150      DO 81 J2=1,J1
7160      IF(J2-NAPLI) 61,61,81
7170 61      N2=J1-J2+1
7180      IF(N2-NBPLI)71,71,81
7190 71      TEMP=TEMP+A(J2)*B(N2)
7200 81      CONTINUE
7210      C(J1)=TEMP
7220 91      CONTINUE
7230      RETURN
7240      END
7250      SUBROUTINE DIV(A,B,NA,NB,Q)
7260      DIMENSION A(9999),B(9999),Q(9999)
7270      I1=NA-NB+1
7280      DO 61 J1=1,I1
7290 61      Q(J1)=0.
7300 101      KKMAX=NA-NB+1
7310      DO 391 KK=1,KKMAX
7320      K=KK-1
7330 201      TEMP=0.
7340      IF(K-1)301,211,211
7350 211      DO 291 JJ=1,K
7360      J=JJ-1
7370      I1=NB-K+J
7380      IF(I1)291,221,221
7390 221      I2=NA-NB-J
7400      TEMP=TEMP+B(I1+1)*Q(I2+1)
7410 291      CONTINUE
7420 301      I1=NA-NB-K
7430      I2=NA-K
7440 391      Q(I1+1)=A(I2+1)-TEMP
7450 5001  RETURN
7460      END
7470C      MTINV
7480*      *****MATRIX INVERSION*****
7490 SUBROUTINE MTINV(A,NRARG,NCARG,IDIM,LABEL)

```

```

7500 DIMENSION A(IDIM,NCARG),LABEL(NRARG)
7510 1 NR=NRARG
7520 NC=NCARG
7530 DO 21 J1=1,NR
7540 21 LABEL(J1)=J1
7550 DO 291 J1=1,NR
7560*      *****FIND REMAINING ROW CONTAINING LARGEST***
7570*      *****ABSOLUTE VALUE IN PIVOTAL COLUMN*****
7580 101 TEMP=0.0
7590 DO 121 J2=J1,NR
7600 IF(ABS(A(J2,J1)).LT.TEMP) GO TO 121
7610 TEMP=ABS(A(J2,J1))
7620 IBIG=J2
7630 121 CONTINUE
7640 IF(IBIG.EQ.J1)GO TO 201
7650*      *****REARRANGE ROWS TO PLACE LARGEST ABSOLUTE
7660*      *****VALUE IN PIVOT POSITION*****
7670 DO 141 J2=1,NC
7680 TEMP=A(J1,J2)
7690 A(J1,J2)=A(IBIG,J2)
7700 141 A(IBIG,J2)=TEMP
7710 I=LABEL(J1)
7720 LABEL(J1)=LABEL(IBIG)
7730 LABEL(IBIG)=I
7740*      ***:COMPUTE COEFFICIENTS IN PIVOTAL ROW***
7750 201 TEMP=A(J1,J1)
7760 A(J1,J1)=1.0
7770 DO 221 J2=1,NC
7780 221 A(J1,J2)=A(J1,J2)/TEMP
7790*      *****COMPUTE COEFFICIENTS IN OTHER ROWS*****
7800 DO 281 J2=1,NR
7810 IF(J2.EQ.J1) GO TO 281
7820 TEMP=A(J2,J1)
7830 A(J2,J1)=0.0
7840 DO 241 J3=1,NC
7850 241 A(J2,J3)=A(J2,J3)-TEMP*A(J1,J3)
7860 281 CONTINUE
7870 291 CONTINUE
7880*      *****INTERCHANGE COLUMNS ACCORDING TO*****
7890*      *****INTERCHANGES OF ROWS OF ORIGINAL MATRIX*
7900 301 NI=NR-1
7910 DO 391 J1=1,NI
7920 DO 321 J2=J1,NR
7930 IF(LABEL(J2).NE.J1) GO TO 321
7940 IF(J2.EQ.J1) GO TO 391
7950 GO TO 341
7960 321 CONTINUE
7970 341 DO 361 J3=1,NR
7980 TEMP=A(J3,J1)
7990 A(J3,J1)=A(J3,J2)
8000 361 A(J3,J2)=TEMP

```

```

8010 LABEL(J2)=LABEL(J1)
8020 391 CONTINUE
8030 5001 RETURN
8040 END
8050C MTMPY -- REV. APRIL 1971
8060* HONEYWELL TIME SHARING APPLICATIONS
8070* *****MATRIX MULTIPLICATION*****
8080 SUBROUTINE MTMPY(IND,A,B,C,LARG,MARG,N)
8090 DIMENSION A(25,999),B(25,999),C(25,999)
8100 1 L=IABS(LARG)
8110 M=IABS(MARG)
8120 I=IND+1
8130 GO TO (101,201,301,401),I
8140 101 DO 121 J1=1,L
8150 DO 121 J2=1,N
8160 C(J1,J2)=0.0
8170 DO 121 J3=1,M
8180 IF(LARG)102,5001,110
8190 102 IF(MARG)103,5001,105
8200* *****T(MATRIX) X T(MATRIX) T(A(M,L))*T(B(N,
8210 103 TEMP=A(J3,J1)*B(J2,J3)
8220 GO TO 121
8230* *****T(MATRIX) X MATRIX T(A(M,L))*B(M,N)
8240 105 TEMP=A(J3,J1)*B(J3,J2)
8250 GO TO 121
8260 110 IF(MARG)111,5001,115
8270* *****MATRIX X T(MATRIX) A(L,M)*T(B(N,M))
8280 111 TEMP=A(J1,J3)*B(J2,J3)
8290 GO TO 121
8300* *****MATRIX X MATRIX A(L,M)*B(M,N)=C(
8310 115 TEMP=A(J1,J3)*B(J3,J2)
8320 121 C(J1,J2)=C(J1,J2)+TEMP
8330 GO TO 5001
8340* *****DIAGONAL X DIAGONAL A(L,1)*B(L,1)=C(
8350 201 DO 221 J1=1,L
8360 221 C(J1,1)=A(J1,1)*B(J1,1)
8370 GO TO 5001
8380 301 DO 321 J1=1,L
8390 DO 321 J2=1,M
8400 IF(MARG)310,5001,315
8410* *****DIAGONAL X T(MATRIX) A(L,1)*T(B(M,L))
8420 310 TEMP=A(J1,1)*B(J2,J1)
8430 GO TO 321
8440* *****DIAGONAL X MATRIX A(L,1)*B(L,M)=C(
8450 315 TEMP=A(J1,1)*B(J1,J2)
8460 321 C(J1,J2)=TEMP
8470 GO TO 5001
8480 401 DO 421 J1=1,L
8490 DO 421 J2=1,M
8500 IF(LARG)410,5001,415
8510* *****T(MATRIX) X DIAGONAL T(A(M,L))*B(M,1)

```

```

8520 410 TEMP=A(J2,J1)*B(J2,1)
8530 GO TO 421
8540* *****MATRIX X DIAGONAL A(L,M)*B(M,1)=C(
8550 415 TEMP=A(J1,J2)*B(J2,1)
8560 421 C(J1,J2)=TEMP
8570 5001 RETURN
8580 END
8590 SUBROUTINE ATOD(W,XMSB,NBITS)
8600 IF(NBITS.LE.0) GO TO 80
8610 U2=0.0
8620 SIGN=-1.
8630 IF(W.GE.0.0) SIGN=1.
8640 XW=W
8650 D=2.0*XMSB
8660 XW=XW*SIGN
8670 20 DELTA2=XMSB
8680 DO 30 I=1,NBITS-1
8690 30 DELTA2=DELTA2/2.
8700 WOUT=2.*(XMSB-DELTA2)
8710 DO 5 I=1,NBITS
8720 D=D/2.0
8730 Y=ABS(XW-U2)
8740 IF(Y.LE.DELTA2) WOUT=U2
8750 X=U2-D
8760 IF(XW.GE.U2) X=U2+D
8770 5 U2=X
8780 IF(XW.LT.DELTA2) WOUT=0.0
8790 IF(XW.GE.(2.*XMSB-D/2.)) PRINT,"A/D HARD LIMITED SAMPLE"
8800 70 W=WOUT*SIGN
8810 80 RETURN
8820 END
8830 SUBROUTINE HMAG(R1A,R2A,XL1A,XL2A,R1,R2,XL1,XL2)
8840 DIMENSION LIM(5),SK(5)
8850 PI2=2.*3.14159
8860 PRINT,""
8870 PRINT,"CALCULATION OF ERROR IN H(S)**2"
8880 PRINT,""
8890 J=1
8900 PRINT:"SYSTEM POLES ARE=",XL1/PI2,XL2/PI2
8910 PRINT:""
8920 PRINT:"SYSTEM RESIDUES ARE=",R1*1.E6,R2*1.E6
8930 PRINT:""
8940 PRINT:" FREQUENCY MAGNITUDE PHASE DIFFERENCE"
8950 PI2=2.*3.14159
8960 XL1A=XL1A*1.E6
8970 XL2A=XL2A*1.E6
8980 R1=R1*1.E6
8990 R2=R2*1.E6
9000 LIM(1)=40
9010 LIM(2)=10
9020 LIM(3)=9

```

```

9030     LIM(4)=10
9040     LIM(5)=15
9050     SK(1)=1.E3
9060     SK(2)=5.E4
9070     SK(3)=1.E6
9080     SK(4)=1.E7
9090     SK(5)=5.E7
9100 200 CONTINUE
9110     DO 100 I=1,LIM(J)
9120     XI=I
9130     XI=XI*SK(J)
9140     W=PI2*XI
9150     AHMAG=W*W*((R1A+R2A)**2)+(R1A*XL2A+R2A*XL1A)**2
9160     AHMAG1=(W*W+XL1A*XL1A)*(W*W+XL2A*XL2A)
9170     AHMAGT=AHMAG/AHMAG1
9180     HMAG2=W*W*((R1+R2)**2)+(R1*XL2+R2*XL1)**2
9190     HMAG1=(W*W+XL1*XL1)*(W*W+XL2*XL2)
9200     HMAGT=HMAG2/HMAG1
9210     Y1=W*(R1+R2)
9220     Y2=R1*XL2+R2*XL1
9230     PHASE=-ATAN2(W,XL1)-ATAN2(W,XL2)+ATAN2(Y1,Y2)
9240     PHASE=PHASE*360./PI2
9250     IF(PHASE.LT.0.) PHASE=PHASE+360.
9260     DBDIF=10.*(ALOG(AHMAGT)/ALOG(10.)-ALOG(HMAGT)/ALOG(10.))
9270     PRINT,XI,HMAGT,PHASE,DBDIF
9280 100 CONTINUE
9290     J=J+1
9300     IF(J.LE.4) GO TO 200
9310     RETURN
9320     END

```

★

APPENDIX B

COMPUTER PROGRAM LISTING FOR THE HYBRID IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE

This is the listing of the computer program for the hybrid implementation. Note that the subroutines are identical to those listed in Appendix A and are not duplicated here.

```

90C      HYBRD2P
100
110      DIMENSION XTA(10),XT(10),X(10),XX(10),RI(10),W(10)
120      DIMENSION AI(10),AP(10),CP(10),CI(10),CA(10)
130      DIMENSION G(10,10),Y(3,2420),U(3,2420),A(20,10),C(25,25)
140      DIMENSION ISN(10),XLX(10),COEF(10),XLAMR(10),XLAMC(10)
150      DIMENSION RR(10),CR(10),B(25),CRR(10,10),CI(10,10)
160      DIMENSION LIMIT1(10),LABEL(25),S(25,1),R(25,1)
170      DIMENSION EID(25,25),E(25,25),CS(25,25)
175      DIMENSION XMSBI(10),XMSBO(10)
180      INTEGER SN,STEP1,STEP2
190C      INITIALIZATION AND INPUT PARAMETERS
200      PI2=2.*3.14159
210      ERMS=0.
220      DELT=4.E-3
230      SN=2
240      DO 118 I=1,SN
250      DO 118 J=1,SN
260 118 EID(I,J)=0.E0
270      DO 119 I=1,SN
280 119 EID(I,I)=1.E0
290      PRINT,"ORDER OF LINEAR SYSTEM IS",SN
300      PRINT,""
310      SN2=2*SN
320      SK=1.E0
330      M=2400
340      MORIG=M
350      LIMIT1(1)=M+1
360      STEP=SN+1
370      T=DELT*M
380      LL=0
390      XMSBI(1)=1.E-3
391      XMSBI(2)=.718E-2
392      XMSBI(3)=.476E-3
393      XMSBI(4)=.217E-2
394      XMSBI(5)=.325E-2
400      XMSBO(1)=.9125E-2
401      XMSBO(2)=.291E-1
402      XMSBO(3)=.1926E-2
403      XMSBO(4)=.685E-2
404      XMSBO(5)=.84E-2
410      NBITS=16
420      PRINT,"NUMBER OF BITS IN A/D=",NBITS
430      PRINT,""
440      PRINT,"NUMBER OF WAVEFORM SAMPLES=",M
450      PRINT,""
460      PRINT,"INTEGRATION TIME IN MICROSECONDS=",T
470      PRINT,""
480      PRINT,"INTEGRATION INTERVAL IN MICROSECONDS=",DELT

```

```

490 PRINT,""
500 900 FORMAT(18X,E14.5)
510C EVALUATE INPUT AND OUTPUT FUNCTIONS
520 R1(1)=2.8069192E5
530 R1(2)=-2.7368441E8
540 R1(3)=1.E-3
550 R1(4)=1.51E7
560 R1(5)=1.E-3
570 W(1)=.011550998*PI2
580 W(2)=10.616986*PI2
590 W(3)=10.
600 W(4)=8.2E-1*PI2
610 W(5)=.35*PI2
620 L=STEP
630 A1(SN+1)=-W(SN+1)
640 DO 901 IK=1,SN
650 A1(IK)=-W(IK)
660 CA(L)=CA(L)+R1(IK)*R1(L)/((W(IK)-W(L))*1.E6)
670 CA(IK)=R1(IK)*R1(L)/((W(L)-W(IK))*1.E6)
680 C1(IK)=-CA(IK)
690 901 CONTINUE
695 C1(L)=-CA(L)
700 DO 999 JK=1,M+1
710 XJK=JK
720 U(1,JK)=0.
730 XT3A=-W(STEP)*(XJK-1)*DELT
740 IF(XT3A.LT.-80.) GO TO 166
750 U(1,JK)=EXP(XT3A)
760 166 CONTINUE
770 U(1,JK)=1.E-3*SK*U(1,JK)
780 CALL ATOD(U(1,JK),XMSB(1),NBITS)
790 DO 902 LJ=1,STEP
800 XT(LJ)=0.
810 XTA(LJ)=-W(LJ)*(XJK-1)*DELT
820 IF(XTA(LJ).GT.-80.) XTA(LJ)=CA(LJ)*EXP(XTA(LJ))
830 902 Y(1,JK)=Y(1,JK)+XT(LJ)
840 Y(1,JK)=SK*Y(1,JK)
850 CALL ATOD(Y(1,JK),XMSB(1),NBITS)
860 999 CONTINUE
870 PRINT," ACTUAL SYSTEM POLES AND RESIDUES"
880 PRINT,""
890 PRINT," NUMBER POLE RESIDUE"
900 PRINT,""
910 PRINT,""
920 DO 446 KL=1,SN
930 PRINT,KL,W(KL)/PI2,R1(KL)
940 446 PRINT,""
950 PRINT,""
960 M2=M
970 PRINT,"INPUT MSB=",XMSB1
980 PRINT,""

```

```

990      PRINT,"OUTPUT MSB=",XMSBO
1000      PRINT,""
1002      JN=SN+1
1005      DO 137 II=1,M+1
1010      XCV=-W(SN+1)*(II-1)*DELT
1012      IF(XCV.LT.-80.) GO TO 1480
1020      U(2,II)=R1(JN)/A1(JN)*(EXP(XCV))
1025      U(3,II)=R1(JN)/(A1(JN)**2)*(EXP(XCV))
1033 1480 U(3,II)=U(3,II)-R1(JN)/A1(JN)*(II-1)*DELT
1034      U(2,II)=U(2,II)-R1(JN)/A1(JN)
1036      U(3,II)=U(3,II)-R1(JN)/(A1(JN)**2)
1100 137 CONTINUE
1110      M2=M2/2
1120 138 CONTINUE
1135      DO 101 JK=1,SN+1
1140      LIMIT1(JK)=M+1
1145 101 CONTINUE
1150      CONST=T/(3.*M)
1160      DO 13 KK=1,M+1
1170      DO 117 IL=1,SN+1
1175      XCV=-W(IL)*(KK-1)*DELT
1177      IF(XCV.LT.-80.) GO TO 1485
1180      Y(2,KK)=Y(2,KK)+CA(IL)/A1(IL)*(EXP(XCV))
1190      Y(3,KK)=Y(3,KK)+CA(IL)/(A1(IL)**2)*(EXP(XCV))
1197 1485 CONTINUE
1198      Y(2,KK)=Y(2,KK)-CA(IL)/A1(IL)
1200      Y(3,KK)=Y(3,KK)-CA(IL)/A1(IL)*(KK-1)*DELT
1210      Y(3,KK)=Y(3,KK)-CA(IL)/(A1(IL)**2)
1265 117 CONTINUE
1270      DO 39 IJK=2,SN+1
1275      CALL ATOD(U(IJK,KK),XMSBI(IJK),NBITS)
1280      CALL ATOD(Y(IJK,KK),XMSBO(IJK),NBITS)
1283 39 CONTINUE
1285 13 CONTINUE
1290 38 CONTINUE
1310C      EVALUATE INNER PRODUCTS FOR GRAM DETERMINANT
1320      M=MORIG
1330      STEP1=SN+1
1340      STEP=2*SN+1
1350      DO 100 K=1,STEP1
1360      LJ=1
1370      M1=M
1380      MSLJ=1
1390 220 LIMIT=M-2
1400      JAT=0
1410      CONST=T/(3.*M)
1420 240 CONTINUE
1430C      EVALUATE G(K,K) FOR K=1,SN

```

```

1440      DO 7 I=1,M-1,2
1450      XI=I
1460      TERM1=CONST
1470      TERM12=Y(K,I)**2+4.*Y(K,I+1)**2+Y(K,I+2)**2
1480      TERM13=TERM1*TERM12
1490      G(K,K)=G(K,K)+TERM13
1500 7 CONTINUE
1510C EVALUATE G(K,KM) FOR J .GT. K TO J=SN+1
1520 DO 3 J=2,STEP1
1530 MSLJ=1
1540 IF(K-J) 435,437,436
1550 435 DO 6 I=1,M-1,2
1560 XTER2=CONST
1570 TERM22=Y(K,MSLJ)*U(J,I)
1580 HSLJ=MSLJ+1
1590 TERM23=4.*Y(K,HSLJ)*U(J,I+1)
1600 KSLJ=MSLJ+2
1610 TERM24=Y(K,KSLJ)*U(J,I+2)
1620 TERM2=XTER2*(TERM22+TERM23+TERM24)
1630 KM=J+SN
1640 G(K,KM)=G(K,KM)+TERM2
1650 MSLJ=KSLJ
1660 6 CONTINUE
1670 GO TO 3
1680C EVALUATE G(K,KM) FOR K .GT. J TO K=SN+1
1690 436 DO 176 I=1,M-1,2
1700 XTER2=CONST
1710 TERM22=Y(K,I)*U(J,MSLJ)
1720 HSLJ=MSLJ+1
1730 TERM23=4.*Y(K,I+1)*U(J,HSLJ)
1740 KSLJ=MSLJ+2
1750 TERM24=Y(K,I+2)*U(J,KSLJ)
1760 TERM2=XTER2*(TERM22+TERM23+TERM24)
1770 KM=J+SN
1780 G(K,KM)=G(K,KM)+TERM2
1790 MSLJ=KSLJ
1800 176 CONTINUE
1810 GO TO 3
1820C EVALUATE G(K,KM) FOR K=J
1830 437 DO 276 I=1,M-1,2
1840 TERM22=Y(K,I)*U(J,I)
1850 TERM23=4.*Y(K,I+1)*U(J,I+1)
1860 TERM24=Y(K,I+2)*U(J,I+2)
1870 XTER2=CONST
1880 TERM2=XTER2*(TERM22+TERM23+TERM24)
1890 KM=J+SN
1900 G(K,KM)=G(K,KM)+TERM2
1910 276 CONTINUE
1920 3 CONTINUE
1930 IF(K-STEP1) 236,100,100
1940 100 CONTINUE
1950C EVALUATE G(K,K) FOR K=SN+1 TO 2*SN+1

```

```

1960      DO 8 K=SN+2,SN2+1
1970      DO 9 I=1,M-1,2
1980      DER1=T/(3.*M)
1990      DER2=U(K-SN,I)**2+4.*U(K-SN,I+1)**2+U(K-SN,I+2)**2
2000      DER3=DER1*DER2
2010      G(K,K)=G(K,K)+DER3
2020      9 CONTINUE
2030      8 CONTINUE
2040      GO TO 425
2050 236 IF(K-SN) 237,237,100
2060C     EVALUATE G(K,LK) FOR K .LT. SN, LK=K+1,SN+1
2070 237 DO 401 LK=K+1,SN+1
2080      MMW=LIMIT1(LK)-2
2090      LJ=1
2100      DO 402 LI=1,M-1,2
2110      TERM31=Y(K,LI)*Y(LK,LI)
2120      KH1=LI+1
2130      TERM32=Y(K,KH1)*Y(LK,LI+1)
2140      KH2=LI+2
2150      TERM33=Y(K,KH2)*Y(LK,LI+2)
2160      TERM34=TERM1
2170      LJ=LJ+2**FLOAT(LK-K+1)
2180      TERM3=TERM34*(TERM31+4.*TERM32+TERM33)
2190      G(K,LK)=G(K,LK)+TERM3
2200 402 CONTINUE
2210 401 CONTINUE
2220 238 IF(K-1) 57,57,58
2230C     EVALUATE G(K+SN,LK+SN) FOR K=2,SN+1, LK=3,SN+1
2240      58 DO 601 LK=K+1,SN+1
2250      MMW=LIMIT1(LK)-2
2260      LJ=1
2270      DO 602 LI=1,M-1,2
2280      TEX1=U(K,LI)*U(LK,LI)
2290      KL1=LI+1
2300      TEX2=U(K,KL1)*U(LK,LI+1)
2310      KL2=LI+2
2320      TEX3=U(K,KL2)*U(LK,LI+2)
2330      TEX4=TERM1
2340      LJ=LJ+2**FLOAT(LK-K+1)
2350      TEX5=TEX4*(TEX1+4.*TEX2+TEX3)
2360      G(K+SN,LK+SN)=G(K+SN,LK+SN)+TEX5
2370 602 CONTINUE
2380 601 CONTINUE
2390      57 CONTINUE
2400      GO TO 100
2410 425 CONTINUE
2420      PRINT,"UNSCALED ENTRIES IN GRAM DETERMINANT ARE"
2430      PRINT,"          ROW          COLUMN          INNER PRODUCT"
2440      PRINT,"          I          J          G(I,J)"
2450      PRINT,""
2460      STEPI=STEPI+SN

```

```

2470      DO 942 JIK=1,STEP1
2480      DO 943 KLI=1,STEP1
2490      G(KLI,JIK)=G(JIK,KLI)
2500      PRINT,JIK,KLI,G(JIK,KLI)
2510 943 CONTINUE
2520 942 CONTINUE
2530      MORIG=M
2540      N=SN2
2550C     SCALE SCALAR PRODUCTS BY 1.E6 FOR COMPUTATION FACILITY
2560      DO 1011 I=1,SN2+1
2570      DO 1021 J=1,SN2+1
2580      G(I,J)=1.E6*G(I,J)
2590 1021 CONTINUE
2600 1011 CONTINUE
2610C     EVALUATE DIAGONAL COFACTORS COEF(I)
2620      PRINT,"DIAGONAL COFACTORS ARE AS FOLLOWS"
2630      DO 300 J=1,SN2
2640      DO 310 I=1,SN2
2650      A(J,I)=G(J+1,I+1)
2660      A(I,J)=A(J,I)
2670 310 CONTINUE
2680 300 CONTINUE
2690      ID=1
2700      COEF(I)=DETE(A,N,20)
2710      PRINT,ID,COEF(I)
2720      DO 500 LKJ=1,SN
2730      DO 400 J=1,LKJ
2740      DO 410 I=1,LKJ
2750      A(J,I)=G(J,I)
2760      A(I,J)=G(I,J)
2770 410 CONTINUE
2780 400 CONTINUE
2790      DO 510 J=1,LKJ
2800      DO 520 I=LKJ+1,SN2
2810      A(J,I)=G(J,I+1)
2820      A(I,J)=A(J,I)
2830 520 CONTINUE
2840 510 CONTINUE
2850      DO 600 J=LKJ+1,SN2
2860      DO 610 I=LKJ+1,SN2
2870      A(J,I)=G(J+1,I+1)
2880      A(I,J)=A(J,I)
2890 610 CONTINUE
2900 600 CONTINUE
2910      COEF(LKJ+1)=DETE(A,N,20)
2920      PRINT,LKJ+1,COEF(LKJ+1)
2930      LNM=LKJ+1
2940 500 CONTINUE
2950      PRINT,"EIGENVALUE EQUATION COEFFICIENTS ARE"
2960C     EVALUATE EIGENVALUE EQUATION COEFFICIENTS R(I)
2970      DO 640 I=1,SN+1

```

```

2980      B(I)=SQRT(ABS(COEF(SN+2-I)))
2990      PRINT,I,B(I)
3000 640  CONTINUE
3010C    EVALUATE SYSTEM POLES
3020      N=SN
3030      PRINT,"POLES OF SYSTEM ARE GIVEN BELOW"
3040      PRINT,"          NUMBER    REAL(MHZ)    IMAG(MHZ)"
3050      CALL DOWNH(B,N,RR,CR)
3060      DO 650 J=1,SN
3070      XLAMR(J)=RR(J)/PI2
3080      XLAMC(J)=CR(J)/PI2
3090      AP(J)=RR(J)*1.E6
3095      AI(J)=1.E6*AI(J)
3100      PRINT,J,XLAMR(J),XLAMC(J)
3110      XLAMR(J)=RR(J)
3120 650  CONTINUE
3130      RR(SN+1)=-W(SN+1)
3131      AI(SN+1)=1.E6*AI(SN+1)
3140      AP(SN+1)=RR(SN+1)*1.E6
3150C    EVALUATE SYSTEM RESIDUES
3160      FLAG=1
3170      DO 660 K=1,SN+1
3180      IF(XLAMC(J)) 670,660,670
3190 670  FLAG=0
3200 660  CONTINUE
3210      IF(FLAG) 680,680,690
3220 690  DO 700 I=1,SN
3230      DO 710 J=1,SN
3240      TEMP1=1
3250      DO 711 L=1,I
3260      TEMP1=TEMP1*XLAMR(J)
3270 711  CONTINUE
3280      EWE1=0.
3290      EWE2=0.
3300      EWE=1./(TEMP1*(XLAMR(J)+W(SN+1)))
3310      EWE1A=XLAMR(J)*T
3320      EWE2A=-W(SN+1)*T
3330      IF(EWE2A.GT.-80.) EWE2=EXP(EWE2A)
3340      IF(EWE1A.GT.-80.) EWE1=EXP(EWE1A)
3350      C(I,J)=EWE*(EWE1-EWE2)
3360      C(I,J)=1.E-3*SK*C(I,J)
3370      TEMP=0.
3380      DO 720 K=1,I
3390      TEMP2=1.
3400      DO 721 KK=1,I+1-K
3410      TEMP2=TEMP2*XLAMR(J)
3420 721  CONTINUE
3430      TEMP=TEMP+U(K+1,LIMIT1(K+1))/TEMP2
3440 720  CONTINUE
3450      C(I,J)=C(I,J)-TEMP
3455      CS(I,J)=C(I,J)
3460 710  CONTINUE

```

```

3470 700 CONTINUE
3480 CALL MTINV(C,N,N,25,LABEL)
3490 ICOUNT=1
3500 EPS=1.E-3
3510 987 CALL MTMPY(O,CS,C,E,N,N,N)
3520 DO 763 I=1,SN
3530 DO 763 J=1,SN
3540 E(I,J)=EID(I,J)-E(I,J)
3550 763 CONTINUE
3560 TEMPO=0.E0
3570 DO 764 I=1,SN
3580 XNOR=0.E0
3590 DO 765 J=1,SN
3600 XNOR=XNOR+ABS(E(I,J))
3610 765 CONTINUE
3620 IF(XNOR.GT.TEMPO) TEMPO=XNOR
3630 764 CONTINUE
3640 IF(TEMPO.GT.1.0) GO TO 997
3650 IF(TEMPO.LT.EPS) GO TO 998
3660 DO 766 I=1,SN
3670 DO 767 J=1,SN
3680 767 E(I,J)=EID(I,J)+E(I,J)
3690 766 CONTINUE
3700 CALL MTMPY(O,C,E,G,N,N,N)
3710 DO 768 I=1,SN
3720 DO 768 J=1,SN
3730 C(I,J)=G(I,J)
3740 768 CONTINUE
3750 IF(ICOUNT.GT.5) GO TO 998
3760 ICOUNT=ICOUNT+1
3770 GO TO 987
3780 997 PRINT,"NORM TOO LARGE",TEMPO
3790 998 CONTINUE
3800 DO 740 I=2,SN+1
3810 S(I-1,1)=Y(I,LIMIT1(I))
3820 740 CONTINUE
3830 CALL MTMPY(O,C,S,R,SN,SN,1)
3840 PRINT,"RESIDUES OF SYSTEM POLES ARE GIVEN BELOW"
3850 DO 751 I=1,SN
3860 PRINT,I,R(I,1)*1.E6
3870 751 CONTINUE
3880 GO TO 1000
3890 680 CONTINUE
3900 1000 CONTINUE
3910 PRINT,"          PERCENTAGE ERROR"
3920 PRINT,""
3930 PRINT,"      NUMBER      POLE      RESIDUE"
3940 PRINT,""
3950 DO 346 I1=1,SN
3960 X(I1)=100.*(R(I1,1)*1.E6-R1(I1))/R1(I1)
3970 XX(I1)=100.*(AP(I1)-A1(I1))/A1(I1)

```

```

3930 346 PRINT, I1, X(I1), YX(I1)
3990 PRINT, ""
4000 PRINT, ""
4010 PRINT, "          JK          ACTUAL      PREDICTED      DIFFERENCE"
4020 SUM=0.
4025 SUM1=0.
4030 J=SN+1
4040 DO 341 JI=1, SN
4050 CP(JI)=R(JI,1)*RI(J)/(-W(J)-RR(JI))
4060 CP(J)=CP(J)+R(JI,1)*RI(J)/(W(J)+RR(JI))
4070 341 CONTINUE
4072 IJK=0
4080 DO 247 JK=1, MORIG+1
4090 XJK=JK
4100 YOUT=0.0
4110 DO 343 IJ=1, SN+1
4120 XT(IJ)=0.0
4130 XTA(IJ)=RR(IJ)*(XJK-1)*DELT
4140 IF(XTA(IJ).GT.-30.) XT(IJ)=CP(IJ)*EXP(XTA(IJ))
4150 YOUT=YOUT-GK*XT(IJ)
4160 343 CONTINUE
4170 DELOUT=YOUT-Y(1, JK)
4180 IJK=IJK+1
4190 IF(IJK.GE.100) GO TO 248
4200 GO TO 249
4210 248 IJK=0
4220 PRINT, JK, Y(1, JK), YOUT, DELOUT
4230 249 CONTINUE
4240 247 CONTINUE
4250 T=T*1.E-6
4260 DO 242 IX=1, SN+1
4270 DO 242 JX=1, SN+1
4275 SUM=SUM-CP(IX)*CP(JX)/(AP(IX)+AP(JX))
4280 IF((AP(IX)+AP(JX))*T.LT.-90.) GO TO 243
4290 SUM=SUM+CP(IX)*CP(JX)/(AP(IX)+AP(JX))*(EXP((AP(IX)+AP(JX))*T))
4295 243 SUM=SUM+2.*CP(IX)*CI(JX)/(AP(IX)+AI(JX))
4300 IF((AP(IX)+AI(JX))*T.LT.-90.) GO TO 244
4310 SUM=SUM-2.*CP(IX)*CI(JX)/(AP(IX)+AI(JX))*(EXP((AP(IX)+AI(JX))*T))
4315 244 SUM=SUM-CI(IX)*CI(JX)/(AI(IX)+AI(JX))
4317 SUM1=SUM1-CI(IX)*CI(JX)/(AI(IX)+AI(JX))
4320 IF((AI(IX)+AI(JX))*T.LT.-90.) GO TO 242
4330 SUM=SUM+CI(IX)*CI(JX)/(AI(JX)+AI(IX))*(EXP((AI(IX)+AI(JX))*T))
4335 SUM1=SUM1+CI(IX)*CI(JX)/(AI(IX)+AI(JX))*(EXP((AI(IX)+AI(JX))*T))
4340 242 CONTINUE
4350 TSUM=SUM/T
4352 SUMNOR=SUM1/T
4353 PRINT, "NORM=", SUMNOR
4354 PRINT, ""
4355 TNORM=TSUM/SUMNOR
4360 PRINT, ""
4370 PRINT, "MEAN SQUARED ERROR=", TSUM
4380 PRINT, ""
4385 PRINT, "NORMALIZED MSE=", TNORM
4390 STOP
4400 END

```

APPENDIX C

COMPUTER PROGRAM LISTING FOR THE ANALOG IMPLEMENTATION OF THE IDENTIFICATION TECHNIQUE

This is the listing of the computer program for the analog implementation. Note that the subroutines are identical to those listed in Appendix A and are not duplicated here.

```

560 PRINT,""
570 PRINT,"INTEGRATION TIME IN MICROSECONDS=",T
580 PRINT,""
590 PRINT,"INTEGRATION INTERVAL IN MICROSECONDS=",DELT
600 PRINT,""
610 900 FORMAT(18X,E14.5)
620C EVALUATE INPUT AND OUTPUT FUNCTIONS
630 R1(1)=2.8069192D5
640 R1(2)=-2.7368441D8
650 R1(3)=1.D-3
660 R1(4)=1.51D7
670 R1(5)=1.D-3
680 W(1)=1.1550998D-2*PI2
690 W(2)=1.0616986D1*PI2
700 W(3)=10.
710 W(4)=8.2D-1*PI2
720 W(5)=.35*PI2
730 L=STEP
740 A1(SN+1)=-W(SN+1)-GAMMA
750 DO 901 IK=1,SN
760 A1(IK)=-W(IK)-GAMMA
770 CA(L)=CA(L)+R1(IK)*R1(L)/((W(IK)-W(L))*1.E6)
780 CA(IK)=R1(IK)*R1(L)/((W(L)-W(IK))*1.E6)
790 C1(IK)=CA(IK)
800 901 CONTINUE
810 C1(L)=CA(L)
820 XJK=JK
830 U(1,JK)=0.
840 XT3A=-W(STEP)*(XJK-1)*DELT
850 IF(XT3A.LT.-80.) GO TO 166
860 U(1,JK)=DEXP(XT3A)
870 166 CONTINUE
880 U(1,JK)=1.D-3*SK*U(1,JK)
900 DO 902 IJ=1,STEP
910 XT(IJ)=0.
920 XTA(IJ)=-W(IJ)*(XJK-1)*DELT
930 IF(XTA(IJ).GT.-80.) XT(IJ)=CA(IJ)*DEXP(XTA(IJ))
940 902 Y(1,JK)=Y(1,JK)+XT(IJ)
950 Y(1,JK)=SK*Y(1,JK)
970 999 CONTINUE
980 PRINT," ACTUAL SYSTEM POLES AND RESIDUES"
990 PRINT,""
1000 PRINT," NUMBER POLE RESIDUE"
1010 PRINT,""
1020 PRINT,""
1030 DO 446 KL=1,SN
1040 PRINT,KL,W(KL)/PI2,R1(KL)
1050 446 PRINT,""
1060 PRINT,""
1070 M2=M
1090 PRINT,""

```

```

100C    ANALOG IMPLEMENTATION-----TWO POLES--
105C    IMPERFECT INTEGRATOR---
107C    VERSION 2-----
110    IMPLICIT DOUBLE PRECISION(A-H,O-Z)
120    DIMENSION FA(10)
130    DIMENSION XTA(10),XT(10),XAX(10),XX(10),RI(10),W(10)
140    DIMENSION AI(10),AP(10),CP(10),CI(10),CA(10)
150    DIMENSION G(10,10),Y(5,1),U(5,1),A(20,10),C(25,25)
160    DIMENSION ISN(10),XLX(10),COEF(10),XLAMR(10),XLAMC(10)
170    DIMENSION RR(10),CR(10),B(25),CRR(10,10),CI(10,10)
180    DIMENSION LIMIT1(10),LABEL(25),S(25,1),R(25,1)
190    DIMENSION EID(25,25),E(25,25),CS(25,25)
200    DIMENSION XMSBI(10),XMSBO(10)
210    INTEGER SN,STEP1,STEP2
220C    INITIALIZATION AND INPUT PARAMETERS
230    PI2=2.*3.14159
235    SG=0.
240    ERMS=0.
250    DELT=4.D-3
260    SN=2
270    DO 118 I=1,SN
280    DO 118 J=1,SN
290 118 EID(I,J)=0.DO
300    DO 119 I=1,SN
310 119 EID(I,I)=1.DO
320    PRINT,"ORDER OF LINEAR SYSTEM IS",SN
330    PRINT,""
340    SN2=2*SN
350    SK=1.DO
360    M=2400
370    MORIG=M
380    LIMIT1(1)=1
390    STEP=SN+1
400    T=DELT*M
405    GAMMA=0.
406    GAMMA=GAMMA*PI2
410    LL=0
420    XMSBI(1)=1.D-3
430    XMSBI(2)=.228D-3
440    XMSBI(3)=.1D-2
450    XMSBI(4)=.217D-2
460    XMSBI(5)=.325D-2
470    XMSBO(1)=.587D-3
480    XMSBO(2)=.127D-2
490    XMSBO(3)=4.D-3
500    XMSBO(4)=.685D-2
510    XMSBO(5)=.84D-2
520    NBITS=0
530    PRINT,"NUMBER OF BITS IN A/D=",NBITS
540    PRINT,""
550    PRINT,"NUMBER OF WAVEFORM SAMPLES=",M

```

```

1100 PRINT,"INTEGRATOR LOW FREQ CUTOFF(HERTZ)",GAMMA/(PI2*1.D-6)
1110 PRINT,""
1115 PRINT,"MULTIPLIER ERROR(%)=",SG*100.
1116 PRINT,""
1120 JN=SN+1
1125 SC=-GAMMA*T
1130 II=1
1140 XCV=-W(SN+1)*M*DELT
1150 IF(XCV.LT.-80.) GO TO 1480
1160 U(2,II)=R1(JN)/A1(JN)*(DEXP(XCV))
1170 U(3,II)=R1(JN)/(A1(JN)**2)*(DEXP(XCV))
1180 U(4,II)=R1(JN)/(A1(JN)**3)*(DEXP(XCV))
1190 U(5,II)=R1(JN)/(A1(JN)**4)*(DEXP(XCV))
1200 1480 U(3,II)=U(3,II)-R1(JN)*DEXP(SC)/A1(JN)*(II-1)*DELT
1210 U(2,II)=U(2,II)-R1(JN)*DEXP(SC)/A1(JN)
1220 U(4,II)=U(4,II)-R1(JN)*DEXP(SC)/(A1(JN)**2)*(II-1)*DELT
1230 U(3,II)=U(3,II)-R1(JN)*DEXP(SC)/(A1(JN)**2)
1240 U(4,II)=U(4,II)-R1(JN)*DEXP(SC)/(2.*A1(JN))*(II-1)*(II-1)
      *DELT*DELT
1250 U(4,II)=U(4,II)-R1(JN)*DEXP(SC)/(A1(JN)**3)
1260 U(5,II)=U(5,II)-R1(JN)*DEXP(SC)/(A1(JN)**3)*(II-1)*DELT
1270 U(5,II)=U(5,II)-R1(JN)*DEXP(SC)/(2.*A1(JN)**2)*((II-1)
      *DELT)**2)
1280 U(5,II)=U(5,II)-R1(JN)*DEXP(SC)/(6.*A1(JN))*((II-1)*DELT)**3)
1290 U(5,II)=U(5,II)-R1(JN)*DEXP(SC)/(A1(JN)**4)
1300 137 CONTINUE
1310 M2=M2/2
1320 138 CONTINUE
1340 DO 101 JK=1,SN+1
1350 LIMIT1(JK)=1
1360 101 CONTINUE
1370 CONST=T/(3.*M)
1380 KK=1
1390 DO 117 IL=1,SN+1
1400 XCV=-W(IL)*M*DELT
1410 IF(XCV.LT.-80.) GO TO 1485
1420 Y(2,KK)=Y(2,KK)+CA(IL)/A1(IL)*(DEXP(XCV))
1430 Y(3,KK)=Y(3,KK)+CA(IL)/(A1(IL)**2)*(DEXP(XCV))
1440 Y(4,KK)=Y(4,KK)+CA(IL)/(A1(IL)**3)*(DEXP(XCV))
1450 Y(5,KK)=Y(5,KK)+CA(IL)/(A1(IL)**4)*(DEXP(XCV))
1460 1485 CONTINUE
1470 Y(2,KK)=Y(2,KK)-CA(IL)*DEXP(SC)/A1(IL)
1480 Y(3,KK)=Y(3,KK)-CA(IL)*DEXP(SC)/A1(IL)*(KK-1)*DELT
1490 Y(3,KK)=Y(3,KK)-CA(IL)*DEXP(SC)/(A1(IL)**2)
1500 Y(4,KK)=Y(4,KK)-CA(IL)*DEXP(SC)/(2.*A1(IL))*((KK-1)*DELT)**2
1510 Y(4,KK)=Y(4,KK)-CA(IL)*DEXP(SC)/(A1(IL)**2)*(KK-1)*DELT
1520 Y(4,KK)=Y(4,KK)-CA(IL)*DEXP(SC)/(A1(IL)**3)
1530 Y(5,KK)=Y(5,KK)-CA(IL)*DEXP(SC)/(A1(IL)**3)*(KK-1)*DELT
1540 Y(5,KK)=Y(5,KK)-CA(IL)*DEXP(SC)/(6.*A1(IL))*((KK-1)*DELT)**3

```

```

1550      Y(5,KK)=Y(5,KK)-CA(IL)*DEXP(SC)/(2.*A1(IL)**2)*(((KK-1)
      *DELT)**2)
1560      Y(5,KK)=Y(5,KK)-CA(IL)*DEXP(SC)/(A1(IL)**4)
1570 117 CONTINUE
1580 13 CONTINUE
1590C      EVALUATE INNER PRODUCTS FOR GRAM DETERMINANT
1600      M=MORIG
1610      STEP1=SN+1
1620      STEP=2*SN+1
1630      SUM=0.
1640      DO 1 I=1,STEP1
1650      DO 1 J=1,SN+1
1660      SUM=SUM-C1(I)*C1(J)/(A1(I)+A1(J))
1670      S3=(A1(I)+GAMMA+A1(J)+GAMMA)*T
1680      IF(S3.LT.-90.) GO TO 1
1690      SUM=SUM+C1(I)*C1(J)/(A1(I)+A1(J))*DEXP(S3)
1700 1 CONTINUE
1710      G(1,1)=SUM
1715      G(1,1)=(1.+SG*UNIFM2(10.,0.,2.))*G(1,1)
1720      SUM=0.
1730      DO 2 I=1,STEP1
1740      SUM=SUM+C1(I)*C1(I)/(2.*A1(I)**2)
1750      X1=X1+C1(I)/A1(I)
1755      S3=2.*(A1(I)+GAMMA)*T
1770      IF(S3.LT.-90.) GO TO 2
1780      SUM=SUM+C1(I)*C1(I)/(2.*A1(I)**2)*DEXP(S3)
1790 2 CONTINUE
1800      DO 3 I=1,SN+1
1810      S3=A1(I)*T
1820      IF(S3.GT.-90.) SUM=SUM-X1*C1(I)/A1(I)*DEXP(S3)
1830      DO 3 J=1,STEP1
1840      IF(J.LE.I) GO TO 3
1850      SUM=SUM+C1(I)*C1(J)/(A1(I)*A1(J))
1860      S3=(A1(I)+GAMMA+A1(J)+GAMMA)*T
1870      IF(S3.LT.-90.) GO TO 3
1880      SUM=SUM+C1(I)*C1(J)/(A1(I)+A1(J))*DEXP(S3)
1890 3 CONTINUE
1900      G(1,2)=SUM
1905      G(1,2)=(1.+SG*UNIFM2(10.,0.,2.))*G(1,2)
1910      SUM=0.
1920      DO 4 I=1,STEP1
1930      DO 4 J=1,STEP1
1940      S1=A1(I)+A1(J)
1950      S2=A1(I)*A1(J)
1960      SUM=SUM+C1(I)*C1(J)/S2*(1./A1(I)+1./A1(J)+T-1./S1)
1970      S3=(A1(I)+GAMMA+A1(J)+GAMMA)*T
1980      IF(S3.LT.-90.) GO TO 41
1990      SUM=SUM+C1(I)*C1(J)/(A1(I)*A1(J))*(1./(A1(I)+A1(J)))*DEXP(S3)
2000 41 S3=A1(I)*T
2010      IF(S3.LT.-90.) GO TO 42
2020      SUM=SUM-C1(I)*C1(J)/S2*DEXP(S3)/A1(I)

```

```

2030 42 S3=A1(J)*T
2040 IF(S3.LT.-90.) GO TO 4
2050 SUM=SUM-C1(I)*C1(J)/S2*DEXP(S3)/A1(J)
2060 4 CONTINUE
2070
2270 G(2,2)=SUM
2275 G(2,2)=(1.+SG*UNIFM2(10.,0.,2.))*G(2,2)
2280 SUM=0.
2290 F=C1(1)/A1(1)+C1(2)/A1(2)+C1(3)/A1(3)
2300 FA(1)=C1(1)/A1(1)*(C1(2)/(A1(1)*A1(2))+C1(3)/(A1(1)*A1(3)))
2310 FA(1)=FA(1)-C1(1)/A1(1)*(C1(2)/(A1(2)**2)+C1(3)/(A1(3)**2))
2320 FA(1)=FA(1)-F*T*C1(1)/A1(1)
2330 FA(2)=C1(2)/A1(2)*(C1(1)/(A1(1)*A1(2))+C1(3)/(A1(2)*A1(3)))
2340 FA(2)=FA(2)-C1(2)/A1(2)*(C1(1)/(A1(2)**2)+C1(3)/(A1(3)**2))
2350 FA(2)=FA(2)-F*T*C1(2)/A1(2)
2360 FA(3)=C1(3)/A1(3)*(C1(1)/(A1(1)*A1(3))+C1(2)/(A1(2)*A1(3)))
2370 FA(3)=FA(3)-C1(3)/A1(3)*(C1(2)/(A1(2)**2)+C1(1)/(A1(1)**2))
2380 FA(3)=FA(3)-F*T*C1(3)/A1(3)
2390 DO 10 I=1,SN+1
2400 S3=A1(I)*T
2410 IF(S3.LT.-90.) GO TO 9
2420 SUM=SUM+FA(I)*DEXP(S3)
2430 9 CONTINUE
2440 S3=2.*(A1(I)+GAMMA)*T
2450 IF(S3.LT.-90.) GO TO 10
2460 SUM=SUM+C1(I)*C1(I)/(2.*A1(I)**3)*DEXP(S3)
2470 10 SUM=SUM-C1(I)*C1(I)/(2.*A1(I)**3)
2480 DO 11 I=1,SN+1
2490 DO 11 J=1,SN+1
2500 IF(J.LE.I) GO TO 11
2510 X1=C1(I)*C1(J)/(A1(I)+A1(J))*(1./(A1(I)**2)+1./(A1(J)**2))
2520 SUM=SUM-X1
2530 S3=(A1(I)+GAMMA+A1(J)+GAMMA)*T
2540 IF(S3.LT.-90.) GO TO 11
2550 SUM=SUM+X1*DEXP(S3)
2560 11 CONTINUE
2570 G(1,3)=SUM
2575 G(1,3)=(1.+SG*UNIFM2(10.,0.,2.))*G(1,3)
2580 X=C1(1)/A1(1)
2590 YA=C1(2)/A1(2)
2600 Z=C1(3)/A1(3)
2610 TOT=X+YA+Z
2620 F1=X*X+2.*X*YA+2.*X*Z+YA*YA+2.*YA*Z+Z*Z
2630 F2=2.*X*X/A1(1)+2.*X*YA/A1(1)+2.*X*YA/A1(2)+2.*X*Z/A1(1)+2.*
    *X*Z/A1(3)
2640 F2=F2+2.*YA*YA/A1(2)+2.*YA*Z/A1(2)+2.*YA*Z/A1(3)+2.*Z*Z/A1(3)
2650 F3=X*X/(A1(1)**2)+2.*X*YA/(A1(1)*A1(2))+2.*X*Z/(A1(1)*A1(3))
2660 F3=F3+YA*YA/(A1(2)**2)+2.*YA*Z/(A1(2)*A1(3))+Z*Z/(A1(3)**2)
2670 F4=X*X+X*YA+X*Z
2680 F4=-2.*F4/A1(1)
2690 F5=-2./A1(2)*(X*YA+YA*Z+YA*YA)

```

```

2700      E5=-2./ZALC3)*(X**YA*(1+1**))
2710      E7=-2.*X*(X/ZALC1)**2)+YA/(CALC1)*ALC2)**2/ZALC1)*ALC3))
2720      E8=-2.*YA*(X/ZALC1)*ALC2)**YA/(CALC2)*ALC2)**2/ZALC2)*ALC3))
2730      E9=-2.*7*(X/ZALC1)*ALC3)**YA/(CALC2)*ALC3)**2/ZALC3)*ALC3))
2740      SUM=(E1+E2+E3+E4+E5+E6+E7+E8+E9)/ZALC1**2)
2750      SUM=(SUM+E5/ZALC2**2)+(E6/ZALC3**2)-(1/ZALC1)-E8/ZALC2)-E9/ZALC3)
2760      E1=ALC1*E
2770      IF (3.1415926-90.0) 30 TO 21
2780      SUM=(SUM+G*G*(ZALC1)-1./ZALC1)**2)+(1/ZALC1)**2)*XPC1)
2790      21 E1=ALC1*E
2800      IF (3.1415926-90.0) 30 TO 22
2810      SUM=(SUM+G*G*(ZALC2)-1./ZALC2**2)+(1/ZALC2)**2)*XPC2)
2820      22 E2=ALC2*E
2830      IF (3.1415926-90.0) 30 TO 23
2840      SUM=(SUM+G*G*(ZALC3)-1./ZALC3**2)+(1/ZALC3)**2)*XPC3)
2850      23 CONTINUE
2860      DO 24 I=1,SN+1
2870      S1=ALC1**2*(C2.*ALC1**2)
2880      SUM=SUM+S1
2890      S3=2.*CALC1*GAMMA*E
2900      IF (3.1415926-90.0) 30 TO 24
2910      SUM=(SUM+S3)*XPC3)
2920      24 CONTINUE
2930      S3=ALC1*GAMMA*ALC2*GAMMA*E
2940      S4=2.*YA/(CALC1)*ALC2)*CALC1)*ALC2))
2950      SUM=SUM+S4
2960      IF (3.1415926-90.0) 30 TO 25
2970      SUM=(SUM+S4)*XPC3)
2980      25 S3=ALC1*GAMMA*ALC3*GAMMA*E
2990      S4=2.*X/(CALC1)*ALC3)*CALC1)*ALC3))
3000      SUM=SUM+S4
3010      IF (3.1415926-90.0) 30 TO 26
3020      SUM=(SUM+S4)*XPC3)
3030      S3=ALC2*GAMMA*ALC3*GAMMA*E
3040      S4=2.*YA/(CALC2)*ALC3)*CALC2)*ALC3))
3050      SUM=SUM+S4
3060      IF (3.1415926-90.0) 30 TO 27
3070      SUM=(SUM+S4)*XPC3)
3080      27 CONTINUE
3090      G(3,3)=0
3100      G(3,3)=(1.*S3*UNIT*PC1**2)+(2.*S4*G(3,3)
3110      G1=X*(C-2.*X/ZALC1)-YA/ZALC2)-1/ZALC3)-YA/ZALC1)-1/ZALC2)
3120      G2=YA*(C-2.*YA/ZALC2)-1/ZALC3)-X/ZALC1)-X/ZALC2)-1/ZALC3)
3130      G3=X*(C-2.*1/ZALC3)-X/ZALC1)-YA/ZALC2)-X/ZALC3)-YA/ZALC2)
3140      SUM=0.
3150      DO 12 I=1,SN+1
3160      SUM=(SUM+ALC1/ZALC1)*DEZ(CALC1)**2)
3170      S3=ALC1*E
3180      IF (3.1415926-90.0) 30 TO 12
3190      SUM=(SUM+ALC1/ZALC1)*DEZ(CALC1/ZALC1)-1./ZALC1**2)**2)*XPC3)
3200      CONTINUE

```

```

3200 D1=X**2/A1(1)+X*YA/A1(2)+X*Z/A1(3)+YA**2/A1(2)+YA*Z/A1(3)
3210 D1=D1+Z*Z/A1(3)
3220 D1=D1+X*YA/A1(1)+X*Z/A1(1)+YA*Z/A1(2)
3230 SUM=SUM+D1*T
3240 B1=X*X+2.*X*YA+2.*X*Z+YA*YA+2.*YA*Z+Z*Z
3250 SUM=SUM+B1*T*T/2.
3260 SUM=SUM-G1/A1(1)-G2/A1(2)-G3/A1(3)
3270 S3=A1(1)*T
3280 IF(S3.LT.-90.) GO TO 40
3290 SUM=SUM+G1/A1(1)*DEXP(S3)
3300 40 S3=A1(2)*T
3310 IF(S3.LT.-90.) GO TO 14
3320 SUM=SUM+G2/A1(2)*DEXP(S3)
3330 14 S3=A1(3)*T
3340 IF(S3.LT.-90.) GO TO 15
3350 SUM=SUM+G3/A1(3)*DEXP(S3)
3360 15 CONTINUE
3370 DO 16 I=1,SN+1
3380 SUM=SUM-C1(I)*C1(I)/(2.*A1(I)**4)
3390 S3=2.*(A1(I)+GAMMA)*T
3400 IF(S3.LT.-90.) GO TO 16
3410 SUM=SUM+C1(I)*C1(I)/(2.*A1(I)**4)*DEXP(S3)
3420 16 CONTINUE
3430 S3=(A1(1)+GAMMA+A1(2)+GAMMA)*T
3440 IF(S3.LT.-90.) GO TO 61
3450 SUM=SUM+X*YA*(1./A1(2)+1./A1(1))*DEXP(S3)
3460 61 S3=(A1(1)+GAMMA+A1(3)+GAMMA)*T
3470 IF(S3.LT.-90.) GO TO 62
3480 SUM=SUM+X*Z*(1./A1(3)+1./A1(1))*DEXP(S3)
3490 62 S3=(A1(2)+GAMMA+A1(3)+GAMMA)*T
3500 IF(S3.LT.-90.) GO TO 63
3510 SUM=SUM+YA*Z*(1./A1(3)+1./A1(2))*DEXP(S3)
3520 63 CONTINUE
3521 SUM=SUM-X*YA*(1./A1(2)+1./A1(1))*(1./(A1(1)+A1(2)))
3523 SUM=SUM-X*Z*(1./A1(3)+1./A1(1))*(1./(A1(1)+A1(3)))
3525 SUM=SUM-YA*Z*(1./A1(3)+1./A1(2))*(1./(A1(2)+A1(3)))
3530 G(2,3)=SUM
3535 G(2,3)=(1.+SG*UNIFM2(10.,0.,2.))*G(2,3)
3540 SUM=0.
3550 KN=SN+1
3560 X=R1(KN)/A1(KN)
3570 DO 28 I=1,SN+1
3580 SUM=SUM-X*(C1(I)/(A1(I)+A1(KN))-C1(I)/A1(I))
3590 S3=(A1(I)+GAMMA+A1(KN)+GAMMA)*T
3600 IF(S3.LT.-90.) GO TO 29
3610 SUM=SUM+X*C1(I)/(A1(I)+A1(KN))*DEXP(S3)
3620 29 S3=A1(I)*T
3630 IF(S3.LT.-90.) GO TO 28
3640 SUM=SUM-X*C1(I)/A1(I)*DEXP(S3)
3650 28 CONTINUE
3660 G(1,4)=SUM

```

```

3665      G(1,4)=(1.+SG*UNIFM2(10.,0.,2.))*G(1,4)
3670      SUM=G(1,4)/A1(KN)
3680      DO 30 I=1,KN
3690      SUM=SUM-X*C1(I)/(A1(I)**2)
3700      S3=A1(I)*T
3710      IF(S3.LT.-90.) GO TO 30
3720      SUM=SUM-X*C1(I)*(T/A1(I)-1./(A1(I)**2))*DEXP(S3)
3730 30 CONTINUE
3740      G(1,5)=SUM
3745      G(1,5)=(1.+SG*UNIFM2(10.,0.,2.))*G(1,5)
3750      SUM=0.
3760      DO 32 I=1,KN
3770      X=R1(KN)/A1(KN)
3780      SUM=SUM+X*C1(I)/A1(I)*(1./A1(I)-1./(A1(I)+A1(KN))+1./A1(KN)+T)
3790      S3=A1(I)*T
3800      IF(S3.LT.-90.) GO TO 33
3810      SUM=SUM-R1(KN)/A1(KN)*C1(I)/(A1(I)**2)*DEXP(S3)
3820 33 S3=(A1(I)+GAMMA+A1(KN)+GAMMA)*T
3830      IF(S3.LT.-90.) GO TO 51
3840      SUM=SUM+R1(KN)/A1(KN)*C1(I)/(A1(I)*(A1(I)+A1(KN)))*DEXP(S3)
3850 51 S3=A1(KN)*T
3860      IF(S3.LT.-90.) GO TO 45
3870      SUM=SUM-R1(KN)/A1(KN)*C1(I)/(A1(I)*A1(KN))*DEXP(S3)
3880 45 CONTINUE
3890 32 CONTINUE
3900      G(2,4)=SUM
3905      G(2,4)=(1.+SG*UNIFM2(10.,0.,2.))*G(2,4)
3910      SUM=G(2,4)/A1(KN)
3920      DO 34 I=1,KN
3930      SUM=SUM+T*T/2.*R1(KN)*C1(I)/(A1(I)*A1(KN))
3940      SUM=SUM-R1(KN)*C1(I)/(A1(I)*A1(KN))/(A1(I)**2)
3950      S3=A1(I)*T
3960      IF(S3.LT.-90.) GO TO 34
3970      SUM=SUM-R1(KN)/A1(KN)*C1(I)/A1(I)*(T/A1(I)-1./(A1(I)**2))
      *DEXP(S3)
3980 34 CONTINUE
3990      G(2,5)=SUM
3995      G(2,5)=(1.+SG*UNIFM2(10.,0.,2.))*G(2,5)
4000      SUM=0.
4010      X=R1(KN)/A1(KN)
4020      DO 35 I=1,SN+1
4030      SUM=SUM+X*C1(I)/A1(I)*(1./(A1(I)*A1(KN))-1./(A1(KN)**2))
4040      SUM=SUM+X*C1(I)/A1(I)*(T/A1(I)-1./(A1(I)*(A1(I)+A1(KN)))+
      (T**2)/2.)
4050      SUM=SUM+X*C1(I)/(A1(I)**3)
4060      S3=(A1(I)+GAMMA+A1(KN)+GAMMA)*T
4070      IF(S3.LT.-90.) GO TO 52
4080      SUM=SUM+X*C1(I)/A1(I)*DEXP(S3)/(A1(I)*(A1(I)+A1(KN)))
4090 52 S3=A1(KN)*T
4100      IF(S3.LT.-90.) GO TO 53
4110      TER=1./(A1(KN)**2)-T/A1(KN)-1./(A1(I)*A1(KN))

```

```

4120      SUM=SUM+X*C1(I)/A1(I)*TER*DEXP(S3)
4130 53 S3=A1(I)*T
4140      IF(S3.LT.-90.) GO TO 35
4150      SUM=SUM-X*C1(I)/(A1(I)**3)*DEXP(S3)
4160 35 CONTINUE
4170      G(3,4)=SUM
4175      G(3,4)=(1.+SG*UNIFM2(10.,0.,2.))*G(3,4)
4180      SUM=G(3,4)/A1(KN)
4190      DO 36 I=1,KN
4200      X=R1(KN)/A1(KN)*C1(I)/A1(I)
4210      YA=X
4220      SUM=SUM-X/(A1(I)**3)+YA*(T*T/(2.*A1(I))+(T**3)/3.)
4230      S3=A1(I)*T
4240      IF(S3.LT.-90.) GO TO 36
4250      SUM=SUM-X*(T/(A1(I)**2)-1./(A1(I)**3))*DEXP(S3)
4260 36 CONTINUE
4270      G(3,5)=SUM
4275      G(3,5)=(1.+SG*UNIFM2(10.,0.,2.))*G(3,5)
4280      SUM=0.
4290      X=R1(KN)/A1(KN)
4300      SUM=X*X*(T+2./A1(KN)-1./(2.*A1(KN)))
4310      S3=A1(KN)*T
4320      IF(S3.LT.-90.) GO TO 37
4330      SUM=SUM-X*X*2./A1(KN)*DEXP(S3)
4340 37 S3=2.*(A1(KN)+GAMMA)*T
4350      IF(S3.LT.-90.) GO TO 44
4360      SUM=SUM+X*X/(2.*A1(KN))*DEXP(S3)
4370 44 CONTINUE
4380      G(4,4)=SUM
4385      G(4,4)=(1.+SG*UNIFM2(10.,0.,2.))*G(4,4)
4390      SUM=G(4,4)/A1(KN)
4400      SUM=SUM-X*X*(1./(A1(KN)**2)-T*T/2.)
4410      S3=A1(KN)*T
4420      IF(S3.LT.-90.) GO TO 38
4430      SUM=SUM-X*X*(T/A1(KN)-1./(A1(KN)**2))*DEXP(S3)
4440 38 CONTINUE
4450      G(4,5)=SUM
4455      G(4,5)=(1.+SG*UNIFM2(10.,0.,2.))*G(4,5)
4460      SUM=G(4,4)/(A1(KN)**2)
4470      SUM=SUM+X*X*(T**3)/3.
4480      SUM=SUM+2.*X*X*((T**2)/(2.*A1(KN))-1./(A1(KN)**3))
4490      S3=A1(KN)*T
4500      IF(S3.LT.-90.) GO TO 39
4510      SUM=SUM-2.*X*X*(T/(A1(KN)**2)-1./(A1(KN)**3))*DEXP(S3)
4520 39 CONTINUE
4530      G(5,5)=SUM
4535      G(5,5)=(1.+SG*UNIFM2(10.,0.,2.))*G(5,5)
4540      PRINT,"UNSCALED ENTRIES IN GRAM DETERMINANT ARE"
4550      PRINT,"          ROW          COLUMN      INNER PRODUCT"
4560      PRINT,"          I          J          G(I,J)"
4570      PRINT,""

```

```

4580     STEP1=STEP1+SN
4590     DO 942 JIK=1,STEP1
4600     DO 943 KLI=1,STEP1
4601
4602     XMSGI=DABS(G(JIK,KLI))*(1.+1./(2.**NBITS))
4605     CALL ATOD(G(JIK,KLI),XMSGI,NBITS)
4610     G(KLI,JIK)=G(JIK,KLI)
4620     PRINT,JIK,KLI,G(JIK,KLI)
4630 943 CONTINUE
4640 942 CONTINUE
4650     MORIG=M
4660     N=SN2
4670C    SCALE SCALAR PRODUCTS BY 1.E6 FOR COMPUTATION FACILITY
4680     DO 1011 I=1,SN2+1
4690     DO 1021 J=1,SN2+1
4700     G(I,J)=1.E9*G(I,J)
4710 1021 CONTINUE
4720 1011 CONTINUE
4730C    EVALUATE DIAGONAL COFACTORS COEFA(I)
4740     PRINT,"DIAGONAL COFACTORS ARE AS FOLLOWS"
4750     DO 300 J=1,SN2
4760     DO 310 I=1,SN2
4770     A(J,I)=G(J+1,I+1)
4780     A(I,J)=A(J,I)
4790 310 CONTINUE
4800 300 CONTINUE
4810     ID=1
4820     COEF(1)=DETE(A,N,20)
4830     PRINT,ID,COEF(ID)
4840     DO 500 LKJ=1,SN
4850     DO 400 J=1,LKJ
4860     DO 410 I=1,LKJ
4870     A(J,I)=G(J,I)
4880     A(I,J)=G(I,J)
4890 410 CONTINUE
4900 400 CONTINUE
4910     DO 510 J=1,LKJ
4920     DO 520 I=LKJ+1,SN2
4930     A(J,I)=G(J,I+1)
4940     A(I,J)=A(J,I)
4950 520 CONTINUE
4960 510 CONTINUE
4970     DO 600 J=LKJ+1,SN2
4980     DO 610 I=LKJ+1,SN2
4990     A(J,I)=G(J+1,I+1)
5000     A(I,J)=A(J,I)
5010 610 CONTINUE
5020 600 CONTINUE
5030     COEF(LKJ+1)=DETE(A,N,20)
5040     PRINT,LKJ+1,COEF(LKJ+1)
5050     LNM=LKJ+1

```

```

5060 500 CONTINUE
5070 PRINT,"EIGENVALUE EQUATION COEFFICIENTS ARE"
5080C EVALUATE EIGENVALUE EQUATION COEFFICIENTS B(I)
5090 DO 640 I=1,SN+1
5100 B(I)=DSQRT(DABS(COEF(SN+2-I)))
5110 PRINT,I,B(I)
5120 640 CONTINUE
5130C EVALUATE SYSTEM POLES
5140 N=SN
5150 PRINT,"POLES OF SYSTEM ARE GIVEN BELOW"
5160 PRINT,"          NUMBER    REAL(MHZ)    IMAG(MHZ)"
5170 CALL DOWNH(B,N,RR,CR)
5180 DO 650 J=1,SN
5190 XLAMR(J)=RR(J)/PI2
5200 XLAMC(J)=CR(J)/PI2
5210 AP(J)=RR(J)*1.E6
5220 AI(J)=-1.D6*W(J)
5230 PRINT,J,XLAMR(J),XLAMC(J)
5240 XLAMR(J)=RR(J)
5250 650 CONTINUE
5260 RR(SN+1)=-W(SN+1)
5270 AI(SN+1)=-1.D6*W(SN+1)
5280 AP(SN+1)=RR(SN+1)*1.E6
5290C EVALUATE SYSTEM RESIDUES
5300 FLAG=1
5310 DO 660 K=1,SN+1
5320 IF(XLAMC(J)) 670,660,670
5330 670 FLAG=0
5340 660 CONTINUE
5350 IF(FLAG) 680,680,690
5360 690 DO 700 I=1,SN
5370 DO 710 J=1,SN
5380 TEMP1=1
5390 DO 711 L=1,I
5400 TEMP1=TEMP1*XLAMR(J)
5410 711 CONTINUE
5420 EWE1=0.
5430 EWE2=0.
5440 EWE=1./(TEMP1*(XLAMR(J)+W(SN+1)))
5450 EWE1A=XLAMR(J)*T
5460 EWE2A=-W(SN+1)*T
5470 IF(EWE2A.GT.-80.) EWE2=DEXP(EWE2A)
5480 IF(EWE1A.GT.-80.) EWE1=DEXP(EWE1A)
5490 C(I,J)=EWE*(EWE1-EWE2)
5500 C(I,J)=1.E-3*SK*C(I,J)
5510 TEMP=0.
5520 DO 720 K=1,I
5530 TEMP2=1.
5540 DO 721 KK=1,I+1-K
5550 TEMP2=TEMP2*XLAMR(J)
5560 721 CONTINUE
5570 TEMP=TEMP+U(K+1,LIMIT1(K+1))/TEMP2

```

```

5580 720 CONTINUE
5590      C(I,J)=C(I,J)-TEMP
5600      CS(I,J)=C(I,J)
5610 710 CONTINUE
5620 700 CONTINUE
5630      CALL MTINV(C,N,N,25,LABEL)
5640      ICOUNT=1
5650      EPS=1.D-3
5660 987 CALL MTMPY(O,CS,C,E,N,N,N)
5670      DO 763 I=1,SN
5680      DO 763 J=1,SN
5690      E(I,J)=EID(I,J)-E(I,J)
5700 763 CONTINUE
5710      TEMPO=0.00
5720      DO 764 I=1,SN
5730      XNOR=0.00
5740      DO 765 J=1,SN
5750      XNOR=XNOR+DABS(E(I,J))
5760 765 CONTINUE
5770      IF(XNOR.GT.TEMPO) TEMPO=XNOR
5780 764 CONTINUE
5790      IF(TEMPO.GT.1.0) GO TO 997
5800      IF(TEMPO.LT.EPS) GO TO 998
5810      DO 766 I=1,SN
5820      DO 767 J=1,SN
5830 767 E(I,J)=EID(I,J)+E(I,J)
5840 766 CONTINUE
5850      CALL MTMPY(O,C,E,G,N,N,N)
5860      DO 768 I=1,SN
5870      DO 768 J=1,SN
5880      C(I,J)=G(I,J)
5890 768 CONTINUE
5900      IF(ICOUNT.GT.5) GO TO 998
5910      ICOUNT=ICOUNT+1
5920      GO TO 987
5930 997 PRINT,"NORM TOO LARGE",TEMPO
5940 998 CONTINUE
5950      DO 740 I=2,SN+1
5960      S(I-1,1)=Y(I,LIMIT(I))
5970 740 CONTINUE
5980      CALL MTMPY(O,C,S,R,SN,SN,1)
5990      PRINT,"RESIDUES OF SYSTEM POLES ARE GIVEN BELOW"
6000      DO 751 I=1,SN
6005
6010      PRINT,I,R(I,1)*1.E3
6020 751 CONTINUE
6030      GO TO 1000
6040 680 CONTINUE
6050 1000 CONTINUE
6060      PRINT,"          PERCENTAGE ERROR"
6070      PRINT,""

```

```

6080      PRINT,"      NUMBER      POLE      RESIDUE"
6090      PRINT,""
6100      DO 346 I1=1,SN
6110      XAX(I1)=100.*(R(I1,1)*1.E6-R1(I1))/R1(I1)
6120      XX(I1)=100.*(AP(I1)-A1(I1))/A1(I1)
6130 346 PRINT,I1,XAX(I1),XX(I1)
6140      PRINT,""
6150      PRINT,""
6170      SUM=0.
6180      SUM1=0.
6190      J=SN+1
6200      DO 341 JI=1,SN
6201      CI(JI)=-CI(JI)
6210      CP(JI)=R(JI,1)*R1(J)/(-W(J)-RR(JI))
6220      CP(J)=CP(J)+R(JI,1)*R1(J)/(W(J)+RR(JI))
6230 341 CONTINUE
6231      CI(J)=-CI(J)
6240      IJK=0
6249      MORIG=1
6250      DO 247 JK=1,MORIG+1
6260      XJK=JK
6270      YOUT=0.0
6280      DO 343 IJ=1,SN+1
6290      XT(IJ)=0.0
6300      XTA(IJ)=RR(IJ)*(XJK-1)*DELT
6310      IF(XTA(IJ).GT.-80.) XT(IJ)=CP(IJ)*DEXP(XTA(IJ))
6320      YOUT=YOUT-SK*XT(IJ)
6330 343 CONTINUE
6340      DELOUT=YOUT-Y(1,JK)
6350      IJK=IJK+1
6360      IF(IJK.GE.100) GO TO 248
6370      GO TO 249
6380 248 IJK=0
6400 249 CONTINUE
6410 247 CONTINUE
6420      T=T*1.E-6
6430      DO 242 IX=1,SN+1
6440      DO 242 JX=1,SN+1
6450      SUM=SUM-CP(IX)*CP(JX)/(AP(IX)+AP(JX))
6460      IF((AP(IX)+AP(JX))*T.LT.-90.) GO TO 243
6470      SUM=SUM+CP(IX)*CP(JX)/(AP(IX)+AP(JX))*(DEXP((AP(IX)+AP(JX))*T))
6480 243 SUM=SUM+2.*CP(IX)*CI(JX)/(AP(IX)+A1(JX))
6490      IF((AP(IX)+A1(JX))*T.LT.-90.) GO TO 244
6500 SUM=SUM-2.*CP(IX)*CI(JX)/(AP(IX)+A1(JX))*(DEXP((AP(IX)+A1(JX))*T))
6510 244 SUM=SUM-CI(IX)*CI(JX)/(A1(IX)+A1(JX))
6520      SUM1=SUM1-CI(IX)*CI(JX)/(A1(IX)+A1(JX))
6530      IF((A1(IX)+A1(JX))*T.LT.-90.) GO TO 242
6540      SUM=SUM+CI(IX)*CI(JX)/(A1(JX)+A1(IX))*(DEXP((A1(IX)+A1(JX))*T))
6550 SUM1=SUM1+CI(IX)*CI(JX)/(A1(IX)+A1(JX))*(DEXP((A1(IX)+A1(JX))*T))
6560 242 CONTINUE
6570      TSUM=SUM/T
6580      SUMNOR=SUM1/T

```

```
6590 PRINT,"NORM=",SUMNOR
6600 PRINT,""
6610 TNORM=TSUM/SUMNOR
6620 PRINT,""
6630 PRINT,"MEAN SQUARED ERROR=",TSUM
6640 PRINT,""
6650 PRINT,"NORMALIZED MSE=",TNORM
6660 STOP
6670 END
```


*MISSION
of
Rome Air Development Center*

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C³I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

**NE
DATE
ILME**