

This dissertation has been 64-4038  
microfilmed exactly as received

WAIT, John Vary, 1932-  
A HYBRID ANALOG-DIGITAL DIFFERENTIAL  
ANALYZER SYSTEM.

University of Arizona, Ph.D., 1963  
Engineering, electrical

University Microfilms, Inc., Ann Arbor, Michigan

A HYBRID ANALOG-DIGITAL DIFFERENTIAL  
ANALYZER SYSTEM

by

John V. Wait

---

A Dissertation Submitted to the Faculty of the  
DEPARTMENT OF ELECTRICAL ENGINEERING  
In Partial Fulfillment of the Requirements  
For the Degree of  
DOCTOR OF PHILOSOPHY  
In the Graduate College  
University of Arizona

1963

# STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in The University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in their judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: John V. Wait

THE UNIVERSITY OF ARIZONA

GRADUATE COLLEGE

I hereby recommend that this dissertation prepared under my  
direction by John V. Wait

entitled A Hybrid Analog-Digital Differential Analyzer System

be accepted as fulfilling the dissertation requirement of the  
degree of Doctor of Philosophy

G. A. Korn  
Dissertation Director  
Granino A. Korn

July 11, 1963  
Date

After inspection of the dissertation, the following members  
of the Final Examination Committee concur in its' approval and  
recommend its acceptance:\*

<u>Joseph F. Forter</u>	<u>18 July 1963</u>
<u>Donald J. Vorobue</u>	<u>20 July 1963</u>
<u>Walter H. Evans</u>	<u>19 July 1963</u>
<u>C. R. Haesembaum</u>	<u>20 July 1963</u>

\*This approval and acceptance is contingent on the candidate's  
adequate performance and defense of this dissertation at the  
final oral examination. The inclusion of this sheet bound into  
the library copy of the dissertation is evidence of satisfactory  
performance at the final examination.

## ACKNOWLEDGEMENT

The author must first of all express gratitude to Professor G. A. Korn for his guidance and inspiration throughout this study. Messrs. R. L. Maybach and T. A. Brubaker provided invaluable assistance in the design of the master timing clock and read-out systems. Messrs. R. L. T. Hampton, B. A. Mitchell, E. P. O'Grady, and H. R. Eckes assisted in the development of special solid-state circuits. Messrs. E. Guenther, M. P. Mitchell, R. Babcock, L. Earhart, and D. Patterson also are credited with the construction of major portions of the experimental equipment. The author is indebted to Miss M. Pio and Mr. E. Guenther for the excellent typing and drafting work on the final copies of the Dissertation.

The author is also grateful to the Office of Aerospace Research, Information Research Division, Air Force Office of Scientific Research and to the Office of Space Sciences, National Aeronautics and Space Administration for their continuing support of this study under joint grant AF-AFOSR-89-63; and to Drs. T. L. Martin, Dean of Engineering and P. E. Russell, Head, Electrical Engineering Department, for their encouragement and contribution of University facilities.

# TABLE OF CONTENTS

		Page
Chapter 1	DESCRIPTION OF HYBRID SYSTEM. . . . .	1
1.1	Introduction. . . . .	1
1.2	System Notation . . . . .	5
1.3	Sequence of Operations. . . . .	11
Chapter 2	HYBRID INTEGRATION PRINCIPLES . . . . .	20
2.1	Introduction. . . . .	20
2.2	Scaling . . . . .	20
2.3	Basic Operations. . . . .	22
2.4	Method of Approach: "DDA-Plus-Interpolation". . . . .	24
2.5	The Digital System. . . . .	25
2.6	The Analog System . . . . .	26
2.7	Polarity Inversion. . . . .	28
2.8	Interpolating Waveforms . . . . .	28
2.9	Timing Sequence . . . . .	28
2.10	Control Clocks. . . . .	30
2.11	Computer Read-Out . . . . .	30
Chapter 3	OTHER HYBRID COMPONENTS . . . . .	38
3.1	Zero-Memory Elements. . . . .	38
3.2	Summing . . . . .	38
3.3	Multiplication. . . . .	40
3.4	Coefficient Changing. . . . .	42
3.5	Generating Function of One Variable . . . . .	42
3.6	Hybrid Division . . . . .	44
Chapter 4	EXPERIMENTAL RESULTS. . . . .	47
4.1	Experimental Procedure. . . . .	47
4.2	Free-Fall Parabolic Trajectory. . . . .	48
4.3	Decaying Exponential. . . . .	51
4.4	Undamped Sinusoid . . . . .	54
4.5	Effect of the Hold Interval . . . . .	58
4.6	Comparator Accuracy . . . . .	58
Chapter 5	COMPARISON OF DIFFERENTIAL ANALYZER SYSTEMS . . . . .	71
5.1	Introduction. . . . .	71
5.2	Analog Differential Analyzers . . . . .	73
5.3	DDA Capabilities. . . . .	75
5.4	Hybrid Differential Analyzers: Theoretical Performance. . . . .	79

		Page
Chapter 6	CONCLUSIONS: PROJECTED APPLICATIONS AND FUTURE STUDIES. . . . .	85
6.1	Summary of Performance of the Hybrid System . . . .	85
6.2	Future Areas of Development: Projected Capabilities . . . . .	88
6.3	Relative Cost . . . . .	90
6.4	Applications. . . . .	91
6.5	Conclusions . . . . .	92
Appendix		
A	DESCRIPTION OF DIGITAL SYSTEM, COMPONENTS, CODING .	93
B	DESCRIPTION OF ANALOG SYSTEM COMPONENTS: COMPARATORS, DIGITAL-ANALOG CONVERTERS. . . . .	102
C	DERIVATION OF OPERATIONS REQUIRED IN A HYBRID SUMMING COMPONENT. . . . .	108
D	MULTIPLYING AND FUNCTION GENERATING WITH HYBRID COMPONENTS: ERRORS IN HYBRID FUNCTION GENERATORS.	111
E	ANALYSIS OF ERRORS IN DIGITAL DIFFERENTIAL ANALYZERS. . . . .	123
F	ROOT PERTURBATION EFFECTS . . . . .	135
REFERENCES . . . . .		139

## LIST OF FIGURES

Figure		Page
1.1a	Hybrid Integrator. . . . .	15
1.1b	Hybrid Multiplier. . . . .	16
1.1c	Hybrid Function Generator. . . . .	17
1.2	Hybrid Variable Representation . . . . .	18
1.3	Representation of the Independent Variable . . . . .	19
2.1	Sine Loop Configuration. . . . .	33
2.2	Graphical Representation of Hybrid Integration . . . . .	34
2.3	Digital Section. . . . .	35
2.4	Analog Section . . . . .	36
2.5	Timing Sequence. . . . .	37
3.1	Summing Component. . . . .	45
3.2	Coefficient Changer. . . . .	46
4.1	$X_D$ , $X_A$ and Their Sum . . . . .	62
4.2	$X_A$ Only. . . . .	63
4.3	Typical Solution Errors. . . . .	64
4.4	Effect of Removing Sawtooth And/or Analog Integrator . . . . .	65
4.5	Solution Error Due to Artificial Analog Errors . . . . .	66
4.6	DC Offset Error. . . . .	67
4.7	Growth of Normal Sinusoidal Solution . . . . .	68
4.8	Effect of Summing Amplifier Gain . . . . .	69
4.9	Variation of $T_H$ . . . . .	70
5.1	Gross Accuracy-Bandwidth Comparison of Differential Analyzers . . . . .	82

5.2	Damping Characteristics of Differential Analyzers. . .	83
5.3	Frequency Errors in Digital and Hybrid Sine Loops. . .	84
B.1	Digital-To-Analog Converters . . . . .	107

# LIST OF TABLES

Table		Page
1.1	Summary of Notation. . . . .	14
2.1	Control Clock Parameters . . . . .	32
4.1	Calibration Test Results . . . . .	48
4.2	Effect of Artificial Errors on Computer Solution . . .	59
4.3	Solution Errors Caused by Errors in Analog Channel: Sine-Wave Problem . . . . .	60
4.4	Root Sensitivity to Analog Channel Errors: Sine-Wave Problem . . . . .	61
5.1	Root Perturbation Due to Truncation Error in the Digital Solution of Differential Equations . . . . .	81

## Chapter 1

### DESCRIPTION OF HYBRID SYSTEM

#### 1.1 INTRODUCTION

The important role of differential analyzers in the field of computation, system simulation, and analysis is well established. Electronic differential analyzers, organized in an all-parallel, operational fashion, provide the system engineer or analyst with a "live mathematical model." These machines not only aid considerably in the reduction of hand computation and analytical effort, but they also provide new insight into system operations.

Electronic analog computers have remained at the forefront in those areas where high accuracy is not required, but where their flexibility, simplicity, ease of programming, and low cost make them a highly useful "every-day tool."

Of course, there are many cases where a higher computation accuracy is required, or where memory and decision-making functions predominate; it has normally been necessary to use digital computers in these instances. In order to provide machines organized in a more useful operational fashion, special purpose digital differential analyzers (DDA's) have been developed.

While promising a potentially higher accuracy (due to the essentially unlimited number of digital bits which can be used to represent a variable), the DDA suffers from a number of readily apparent disadvantages. The cost and complexity of an all-parallel DDA is unquestionably greater than an equivalent analog machine. What is even worse from a functional standpoint, the DDA has the inherent difficulties of all systems which utilize quantized variable representations, that is, truncation and round-off errors. Practical fast DDA's use relatively simple integration algorithms, usually an open trapezoidal rule (see Ref. 25). To achieve a desired minimum error in the final solution, the DDA programmer must often exercise a considerable amount of skill in constructing his program. At best, he may be forced to operate at a frustratingly slow time scale, in order to provide the fineness of iteration necessary to achieve a desired degree of accuracy. In many cases, this measure alone will not necessarily insure that some mechanism has not yielded an unpredicted error in the result.

In recent years, considerable effort has been expended toward combining the capabilities of digital and analog systems. It is relatively easy to provide analog computers with a fair measure of memory and decision-making capability. Systems using combined analog and digital computing elements with D-A and A-D converters for data conversion have been successfully employed in many simulation and computation installations. In these installations, the digital system serves for accurate and relatively slow operations most suited to its

nature, e. g., coordinate conversions, table look-up, precision function generation, data storage, decision making, etc. The analog elements are used to perform linear operations such as integration and summation, and wherever greater speed and less accuracy is required (Ref. 1, 4, 6, and 17).

The concept of blending analog and digital elements can be extended to a system of true hybrid computing elements. To clarify what is meant, consider the possibilities provided by a system wherein the values of variables (and parameters) are each represented by a combination of a coarse digital word together with a continuous analog interpolation voltage. In theory, at least, it would appear possible that the accuracy of the analog channel would be improved by roughly  $1/2^n$ , where  $n$  is the number of digital bits used in the digital representation. For example, using analog components with an accuracy of 1 per cent of full scale with a 7-bit digital word, the resulting full scale accuracy of the hybrid variables would be one part in  $100 \times 2^7$ , i.e., about 0.01 per cent. A little thought will bring to light the obvious restriction that one must trade speed (whether viewed in terms of frequency, rise time, or slewing rate) for accuracy. This same limitation, of course, applies to DDA's, and to some extent, to analog computing systems. The hybrid differential analyzer may be regarded as a relatively inexpensive parallel DDA whose truncation and round-off errors are essentially eliminated through interpolation with analog computing elements. This feature, rather than accuracy or speed as such, is considered to be the salient advantage of such a system.

The successful implementation of a true hybrid system depends upon the development of suitable transducers or data converters between the analog and digital interfaces. Basically, the problem is one of developing fast comparators and analog switches.

Under the assumption that the practical problems can be solved, it is then a fairly straight forward matter to outline the required system interconnections to implement the operational elements of a true hybrid system. Considerable work has already been done in this area by Skramstad, Schmid, Korn, and others (Ref. 1, 4, 17, 29, and 31). The purpose of this study was to explore in detail the practical aspects of hybrid differential analyzers, and to verify experimentally the capabilities and limitations of a typical system.

In order to be useful in practical application, a hybrid differential analyzer system must be capable of performing at least those operations now within the capabilities of conventional analog and digital systems. One class of problems for which hybrid systems should be useful is for space-vehicle trajectory calculations. This type of problem involves nonlinear differential equations, and the input data normally must undergo numerous coordinate transformations. It is apparent that a useful hybrid computing system must not only contain linear operational elements, such as integrators and summers, but also must contain elements capable of performing multiplication, division, precise coefficient setting, polar-rectangular conversion (resolving), and general function generation.

The major portion of the experimental work associated with this thesis is devoted to the performance of hybrid integrators. However, later sections will discuss the requirements of hybrid elements for performing all of the above operations. The actual system is designed so as to retain as much flexibility as possible in the manner in which these operations are implemented. Figure 1.1 shows abbreviated block diagrams of typical hybrid computing elements; details of the design of these elements appear in subsequent chapters.

## 1.2 SYSTEM NOTATION<sup>1</sup>

### 1.2.1 HYBRID REPRESENTATION OF VARIABLES: ACCURACY

Consider a system where each problem variable  $x$  is represented by a machine variable  $X = a_x x$ , appearing in the form

$$X = X_D + X_A$$

where  $X_D$  is a digital word with  $n$  binary digits plus sign bit;  $X_A$  is an analog voltage between  $-E$  and  $+E$  volts. Either 1 binary digit or  $E$  volts represents 1 machine unit (m.u.). We note that  $X_A$  is an interpolating voltage representing the fractional part of  $X$  (Fig. 1.2).

In this study,  $n = 3$ , ( $2^n = 8$ ) and  $E = 10$  volts.

Assuming analog-computer accuracy within  $p$  per cent of  $E$ , this representation yields  $100 \frac{2^n}{p}$  distinguishable increments of  $X$

---

<sup>1</sup>The material presented in this section is primarily derived from discussions with Prof. G. A. Korn, whose suggested notation is used throughout this paper. For a notation summary, see Table 1.1.

between 0 and  $2^n$ , or a half-scale accuracy of  $2^{-n_p}$  per cent.

For any analog voltage  $e$  between 0 and 0.5 machine unit, note that both  $X_D + e$  and  $(X_D + 1) + (e - 1)$  represent the same value of the hybrid machine variable  $X$ . Although this redundancy halves the analog resolution, it permits us to use relatively inaccurate analog comparators to generate carries.

### 1.2.2 REPRESENTATION OF THE INDEPENDENT OR TIME VARIABLE (Fig. 1.3)

The range of the independent variable  $t \geq 0$  is divided into equal increments  $\Delta t$ , so that

$$t = (k-1) \Delta t + \frac{1}{\alpha_t} \tau; \quad (k = 1, 2, \dots) \quad (2)$$

where  $\tau$  varies periodically between 0 and  $\alpha_t \Delta t$  as  $t$  increases. Each interval of length  $\Delta t$  will correspond to an individual analog-computing period of duration  $T$ , during which

$$\tau = \alpha_t \left[ t - (k-1) \Delta t \right] \geq 0; \quad (k = 1, 2, \dots) \quad (3)$$

is the computer time (real time);  $\alpha_t$  is a time scale factor suitably chosen so that

$$\alpha_t \Delta t = T \quad (4)$$

After each run, a holding interval of length,  $T_H$ , is used for performing digital updating operations, generating analog carries and resets, etc. This is a significant departure from earlier hybrid differential analyzer systems. The interruptions in the computation

complicate the introduction of real-time data inputs, but otherwise does not place any major restrictions on system capabilities.

### 1.2.3 VARIATION OF THE MACHINE VARIABLES WITH TIME: THE ANALOG-COMPUTING PERIOD

At the start of the  $k^{\text{th}}$  computing period  $[t = (k-1)\Delta t, \tau = 0]$ , the digital component  $X_D$  and the analog component  $X_A$  of each machine variable  $X$  are reset to their correct values<sup>1</sup>.

$${}^k X_D = X_D \left[ (k-1) \Delta t \right] \quad (k = 1, 2, \dots) \quad (5)$$

$${}^k X_A (0) = X_A \left[ (k-1) \Delta t \right] \quad (k = 1, 2, \dots) \quad (6)$$

Each digital component  $X_D$  remains constant during the entire computing period, while each analog component

$$X_A(t) = X_A \left[ (k-1)\Delta t + \frac{1}{\alpha_t} \tau \right] = {}^k X_A(\tau) \quad (k = 1, 2, \dots; \tau \geq 0) \quad (7)$$

varies as a function of the computer time  $\tau$  as dictated by the computing interconnections for the given problem.

### 1.2.4 THE DIGITAL-COMPUTING AND CARRY-GENERATING PERIODS

At the end of the  $k^{\text{th}}$  analog-computing period, each analog voltage is held and generates a positive or negative carry ( $\pm 1$  m.u. increment) if

$$\left| {}^k X_A(\alpha_t \Delta t) \right| > \frac{1}{2} \text{ m.u.} \quad (k = 1, 2, \dots) \quad (8)$$

<sup>1</sup>Note that the actual values of  $X_A$  and  $X_D$  do not necessarily differ between the end of one run and the beginning of another; they will differ only if a carry is made.

The carries are used as digital  $\pm 1$ -bit increments to update the digital components  $^k x_D$ , also new digital components  $^{k+1} x_D$  are computed digitally.

During the same holding period, the positive or negative carry machine units are subtracted from the corresponding analog voltages (which can, therefore, never exceed 1 m.u.); and precise fractional parts of the digital components  $^{k+1} x_D$  are computed digitally, to be introduced into the next analog computation. Note that the digital computing elements accept n-digit words, but effectively produce longer words.

#### 1.2.5 COMPUTING SPEED (See Also Section 5.2 and Appendices B and E)

No machine variable  $X = X(t)$  may be allowed to increase or decrease by more than  $1/2$  m.u. ( $1/2$  bit) during any one computing period of  $T$  seconds; hence we must scale so that

$$\left| \frac{dX}{dt} \right| < \frac{1}{2\Delta t} = \frac{\alpha t}{T} \text{ m.u./sec} \quad (9)$$

If the computer time  $\tau$  is to represent  $t$  on a 1:1 time scale ( $\alpha_t = 1$ ) during each analog computing period, then we have  $T = \Delta t$ , and the maximum absolute rate of  $X$  is given by

$$\left| \frac{dX}{dt} \right| < \frac{1}{2T} \text{ m.u./sec} \quad (10)$$

We can, in this case, represent a full-scale sinusoid

$$X(t) = 2^n \sin 2\pi f t$$

with

$$\left| \frac{dX}{dt} \right|_{\max} = 2\pi f \cdot 2^n \leq \frac{1}{2T} \text{ m.u./sec} \quad (11)$$

if

$$f \leq \frac{1}{4\pi T \cdot 2^n} = B_{\text{HYBRID}} \text{ cps} \quad (12)$$

We will call  $B_{\text{HYBRID}}$  the full-scale bandwidth of the hybrid computer.

A given full-scale bandwidth  $B_{\text{HYBRID}}$  requires

$$T \leq \frac{1}{4\pi B_{\text{HYBRID}} 2^n} \text{ sec} \quad (13)$$

The analog computing elements of the hybrid computer must permit the full rate of change

$$\frac{dX_A}{dt} = 2\pi B_{\text{HYBRID}} 2^n = \frac{1}{2T} \text{ m.u./sec}$$

i.e., the analog computing element must be able to produce a full-scale analog sinusoid

$$X_A(t) = \sin 2\pi B_{\text{ANALOG}} t$$

with

$$B_{\text{ANALOG}} \geq 2^n B_{\text{HYBRID}} = \frac{1}{4\pi T} \text{ cps} \quad (14)$$

Note that increased digital accuracy necessarily requires a proportional increase in the required analog bandwidth once  $B_{\text{HYBRID}}$  is given.

In the system to be described in this paper, a  $\pm 10$  volt analog signal range was selected so that the majority of analog operations could be easily implemented with modern fast transistor operational amplifiers. The speed of the system is basically determined by an estimate of the maximum rate of change of the analog variables which could successfully be accommodated by the analog computing elements, and by anticipated timing errors in the control and read-out equipment.

As discussed in Appendix B, it was estimated that the analog systems should be able to maintain one per cent of half-scale component accuracy if the variables have a maximum rate-of-change of 4000 v/sec. This corresponds to a full-scale  $B_{ANALOG}$  of about 64 cps; this was felt to be a conservative limit on computing speed using modern analog computing elements (see also Section 4.2).

Using the relationship

$$\left| \frac{dx}{dt} \right| \leq \frac{1}{2T} \leq 4000 \text{ v/sec}$$

one finds that  $T$  must be greater than 1250 microseconds.

Further considerations associated with obtaining a convenient gain constant for the hybrid integrator (see Chapter 2) led to a choice of  $T = 1250$  microseconds.

In order to insure ample time for performing digital operations, the analog holding interval,  $T_H$ , was made equal to  $0.04T = 50$

microseconds. If  $T_H$  were zero, the system could keep up with full-scale sine waves of angular frequency 50 rad/sec. Inclusion of  $T_H$ , however, reduces this figure to 48 rad/sec. or about 7.65 cps.

To keep up with a real-time full-scale sinusoid at  $f$  cps with  $n = 3$ , requires  $B_{HYBRID} \geq 1.04f$  cps,  $B_{ANALOG} \geq 8.32f$  cps, if we allow  $0.04T$  extra seconds per computing period for the digital computation and resetting. The speed-accuracy ratio of the hybrid computer is  $100/2pT$  distinguishable increments/sec. If we reduce this by  $1/25$  to allow for the digital-computing periods, we have  $625/13pT = 38,500$  distinguishable increments/sec., which permits a crude comparison to modern incremental digital differential analyzers.

### 1.3 SEQUENCE OF OPERATIONS (Fig. 1.1, 1.4, 2.6)

The computing operations involved in a hybrid differential analyzer system must be performed in the proper sequence. In the present system, the major steps are as follows:

#### a. Initial Hold

Prior to the beginning of a computer run, the INITIAL HOLD state (analogous to HOLD in an analog computer) is established by putting the proper digital and analog initial conditions into all integrators. At this time, a particular Total Run Time may be selected, which will stop computation and command solution read-out.

#### b. Run

When the computer run is initiated, all analog subsystems are made operative for the duration of the first computing interval  $T$ .

A holding interval,  $T_H$ , is then initiated. During  $T_H$ , all analog integrators are in HOLD, and the following operations are performed:

1. Digital Integration (updating) simultaneously in all integrators, and transmission of carries from integrators to other computing subsystems.
2. Digital Summing at the input of all summing devices and carry transmission.
3. Digital Operations in multipliers, coefficient setters, function generators, and any other zero-memory devices; transmission of carries.

Some of these operations may overlap in time, but it is essential that digital updating in all integrators be performed and the necessary carries transmitted to subsequent elements in the computing loop. Digital data transfer will normally be incremental ternary transfer (carry pulse and DC carry sign signal). All digital operations are under the control of a subroutine clock, which can be expanded to drive a large number of digital subsystems simultaneously.

After the digital operations are completed, the new states of the digital system will automatically create step transients in the analog channels, through their effect on various D/A converters. Analog interpolation voltages will also be reset to zero (see Chapter II). After these transients subside, another analog computing interval,  $T$ , may be initiated.

The above operations are repeated for the desired number of computing intervals. Figure 4.1 shows how the analog and digital

parts of the solution combine to form the complete variable.

c. Read-Out

At the end of the desired number of computing steps, the computation is stopped. The present system permits setting the total run time from 0.01 to 999.99 computing intervals (total active machine time of 999.99 T seconds). When this time is reached, the read-out system displays the value of a pre-selected machine variable. A high-speed sample-hold system and a digital voltmeter provide a digital display of the analog variable at the read-out time. A continuous full-analog signal is also provided for display purposes.

d. Reset

Initiation of the Reset interval returns the computer to the Initial Hold state described in (a) above. This includes resetting all digital variables to zero, and then inserting new initial conditions, as desired.

TABLE 1.1

## SUMMARY OF NOTATION

$X_D$	Digital part of a variable
$X_A$	Analog part of a variable
$E$	Number of volts in one m.u.
$n$	Number of bits in $X_D$ exclusive of sign ( $n = 3$ in the system being discussed)
$P$	o/o half-scale analog accuracy
$k$	Index number for computing periods, $K = 1, 2, \dots$
$t$	Independent problem time variable
$\Delta t$	Problem time increments
$T$	Duration of machine periods
$\tau$	Machine time ( $0 \leq \tau \leq T$ )
$\alpha_t$	Time scale factor

$$\alpha_t \Delta t = T : T = \frac{\alpha_t}{\Delta t}$$

$$0 \leq \tau \leq T$$

$$t = (k - 1) \Delta t \text{ at } \tau = 0$$

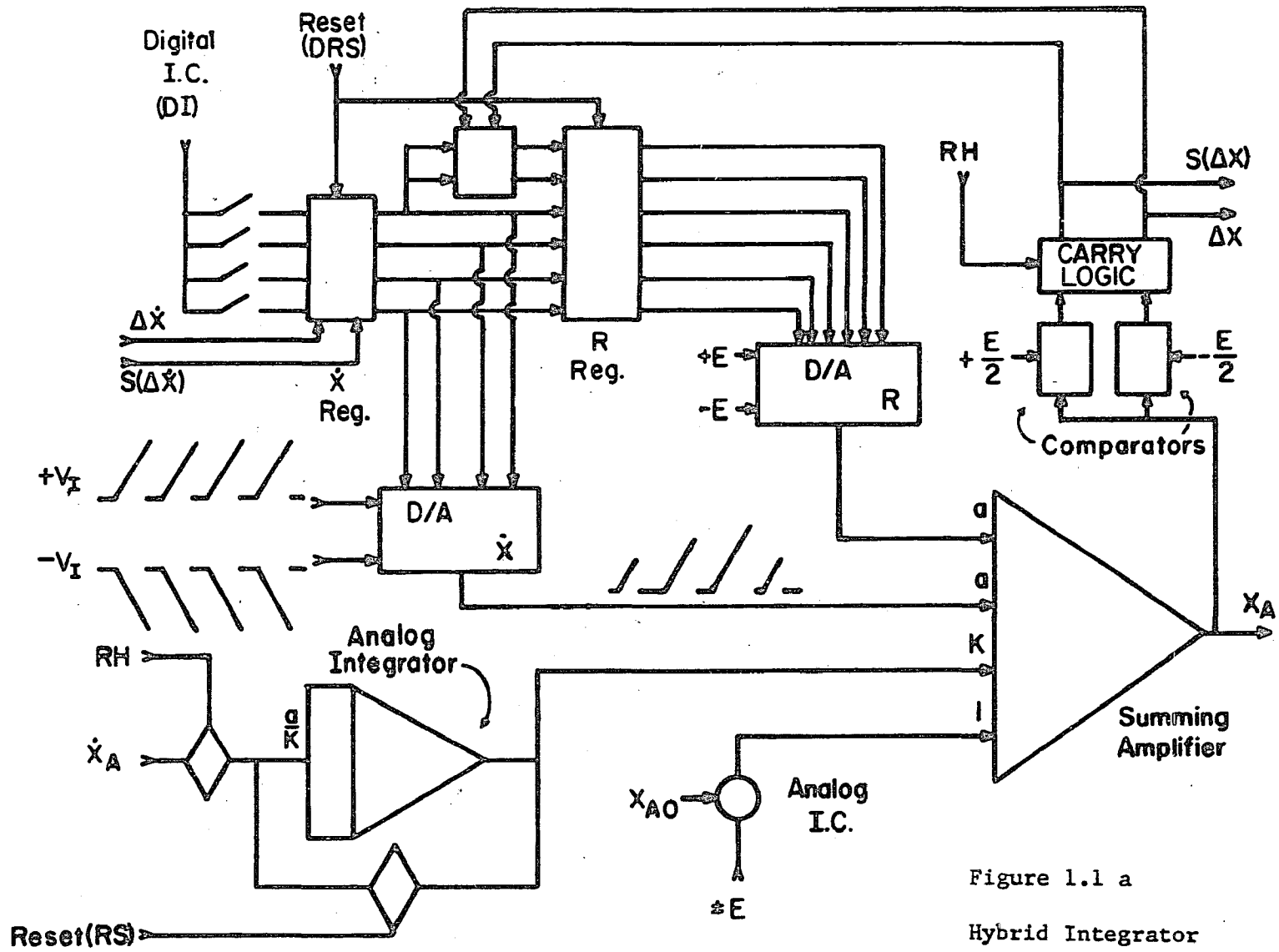


Figure 1.1 a  
Hybrid Integrator

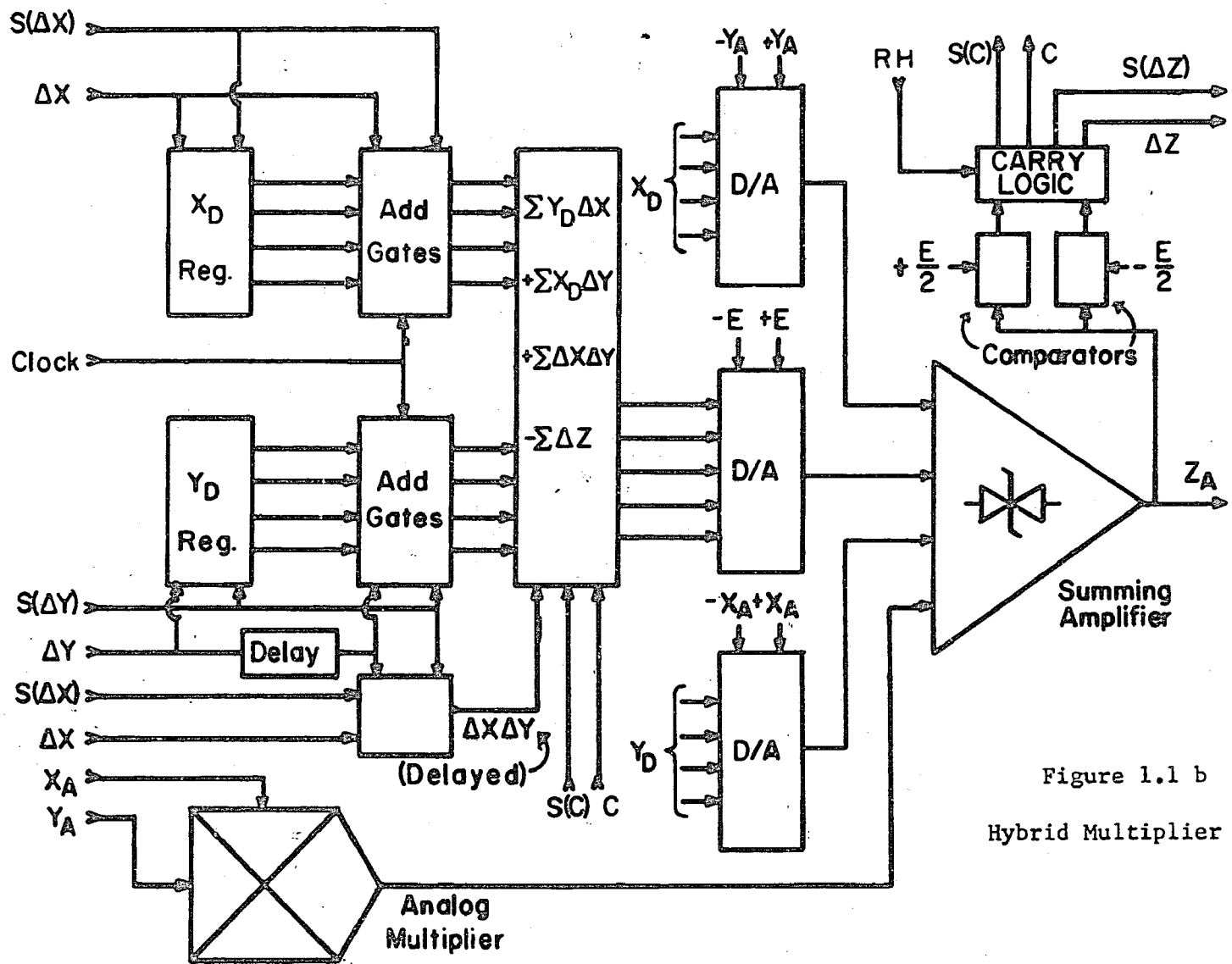


Figure 1.1 b  
Hybrid Multiplier

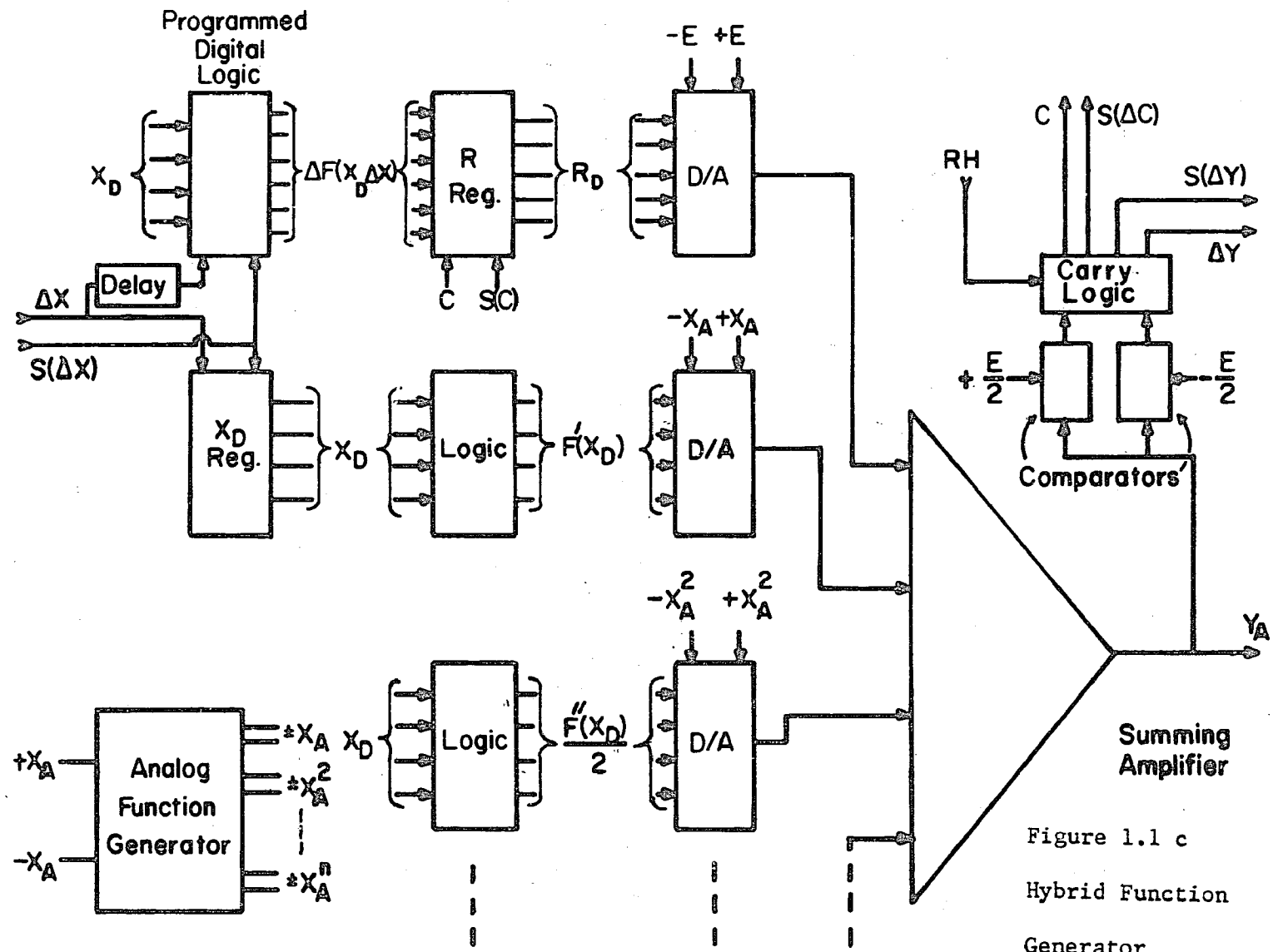


Figure 1.1 c  
Hybrid Function  
Generator

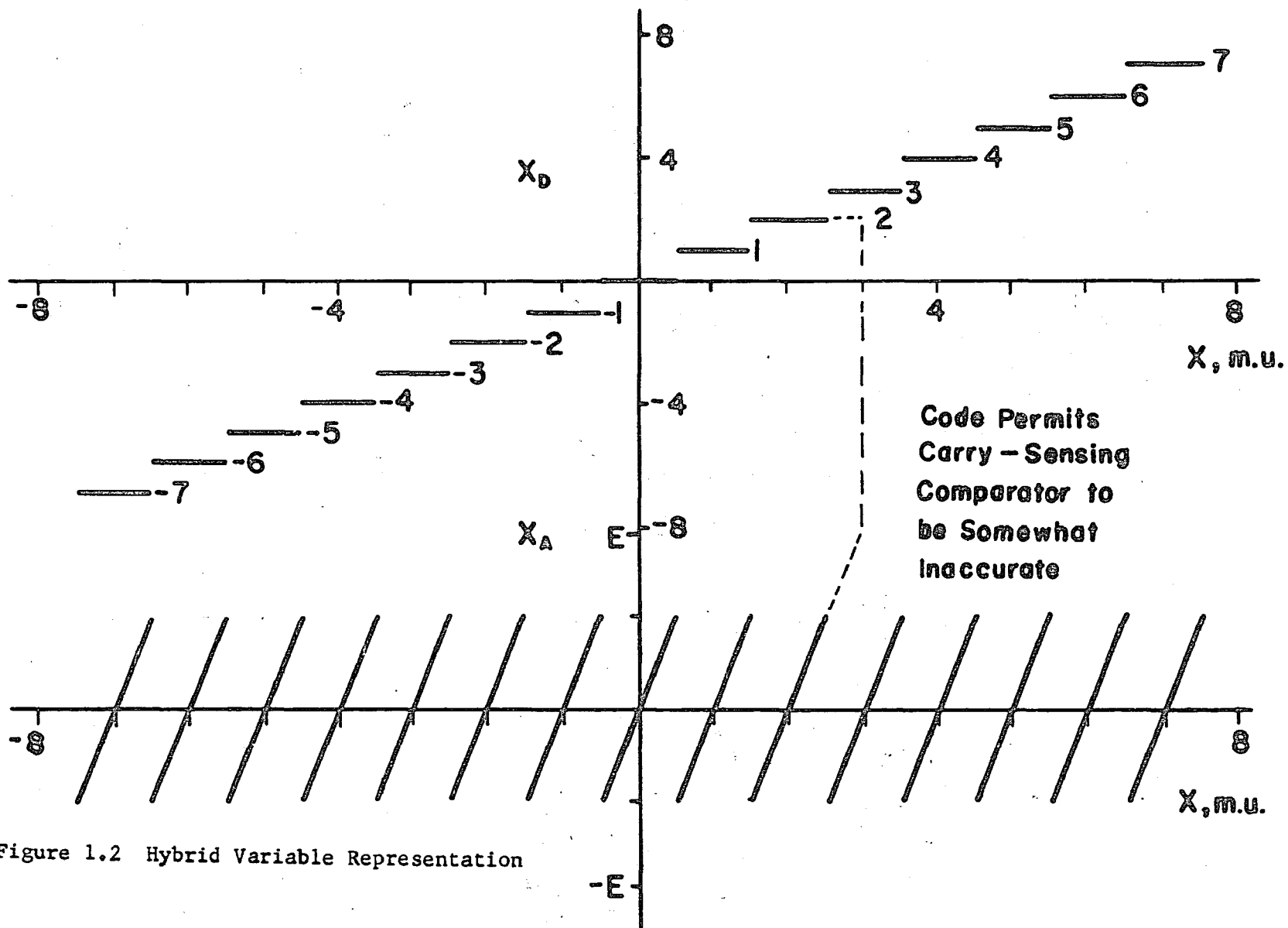


Figure 1.2 Hybrid Variable Representation

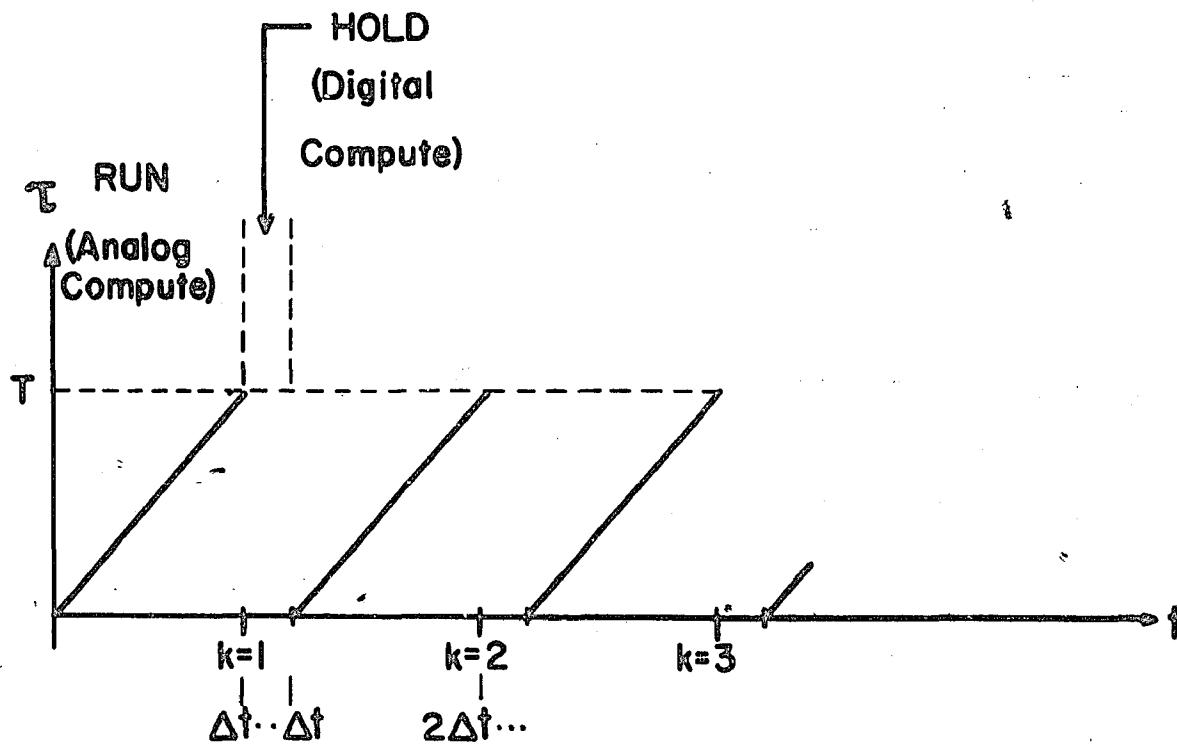


Figure 1.3 Representation of the Independent Variable

The independent variable  $t$  is represented by the integer  $k = 1, 2, \dots$  and the computer time during successive computing periods.

## Chapter 2

### HYBRID INTEGRATION PRINCIPLES

#### 2.1 INTRODUCTION

A hybrid integrator implements the operation

$$X(mT + \tau) = {}^mX_D + X_A(mT + \tau) = a \int_0^{mT + \tau} \dot{X}(\alpha_t t) d(\alpha_t t) + X_0$$

where :

T is the minimum time between possible resets of the analog channel.

The dynamic range of the analog channel is nominally  $\pm 1/2$  m.u.

The dynamic range of the digital channel is  $\pm (2^n - 1)$ , i.e.,  $\pm 7$  m.u.

The full-scale range of hybrid variable is  $\pm (2^n - 1/2)$  m.u.  $\approx \pm 7 \frac{1}{2}$  m.u.

By making T small enough, we ensure that the change in the integral of the digital portion of X is less than  $1/2$  m.u. in machine time  $\alpha_t \Delta t \approx T$  sec.

#### 2.2 SCALING

The magnitude and rate-of-change of  $X(t)$  and  $\dot{X}(t)$  are both assumed to be limited by appropriate scaling so that:

$$\left| \frac{dX}{d(\alpha_t t)} \right|, \left| \frac{d\dot{X}}{d(\alpha_t d_t)} \right| \leq \frac{1/2}{T} \text{ m.u./sec.} \approx \frac{1}{2T} \text{ m.u./sec.}$$

$$\begin{aligned} |X|, |\dot{X}| &\leq (2^n - 1/2) \leq 2^n \\ &\leq 7 \frac{1}{2} \leq 8 \end{aligned}$$

In a time interval  $T$  seconds long

$$\Delta X \leq a \int_0^T \dot{X} d(\alpha_t) \leq a T |\dot{X}|_{\max}$$

But

$$|\Delta X| \leq 1/2 \text{ m.u. in } T \text{ sec.}$$

Therefore

$$|\Delta X|_T \leq 1/2$$

And

$$a T |\dot{X}|_{\max} \leq 1/2$$

Or

$$a \leq \frac{1}{2^{n+1}T} \leq \frac{1}{16T} = \frac{1}{16\alpha_t \Delta t}$$

ensures meeting the scaling conditions above, regardless of  $|\dot{X}|$ .

As an example, with  $n = 3$  and  $T = 1.25 \text{ ms.}$ ,  $a \leq 50$  is satisfactory.

Consider a sample sine loop (Fig. 2.1) solving the differential equation

$$\ddot{X} = -a_1 a_2 X; \quad X(0) = 7.5, \quad \dot{X}(0) = 0$$

Here we have  $X = 7.5 \cos \omega t$

$$|\dot{X}| = \frac{7.5\omega}{a_2} \sin \omega t = \frac{7.5a_1}{\omega} \sin \omega t$$

Hence we have

$$\omega = \sqrt{a_1 a_2}, \quad \frac{\omega}{a_2} \leq 1, \quad \frac{a_1}{\omega} \leq 1$$

Thus  $\omega = a_1 = a_2 = 50$  is maximum for  $\omega$ ,  $a_1$  or  $a_2$  and thus we find the maximum value of  $\omega$  is 50 rad./sec. or  $f_{\text{HYBRID}} \simeq 8.0$  cps (similarly, for  $T = 500 \mu\text{sec}$ , we have 20 cps). The above calculations neglect the digital computing "dead-space" of 50  $\mu\text{sec}$ , which slows the real-time computation by 4 per cent below that calculated above; i.e.,  $f_{\text{HYBRID}} \simeq 7.65$  cps.

### 2.3 BASIC OPERATIONS

Now let us examine integration over a period of  $m$  computing intervals. For simplicity, we will assume  $\alpha_t = 1$ .

Then

$$X = a \int_0^{mT + \tau} \dot{X} dt + X_0$$

In General

$$\dot{X} = \dot{X}_D + \dot{X}_A$$

$$X = X_D + X_A$$

$$X_0 = X_{D0} + X_{A0}$$

Initially

$$k = 1$$

$$\dot{X}_D = \dot{X}_D^1$$

$$X_D = X_D^1$$

Thus

$$\begin{aligned} X &= X_{D0} + a \int_0^{mT + \tau} \dot{X}_D dt + a \int_0^{mT + \tau} \dot{X}_A dt + X_{A0} \\ &= X_{D0} + a \sum_{k=1}^{m-1} \dot{X}_D^k T + a \int_0^{\tau} \dot{X}_D^m dt + a \int_0^{mT + \tau} \dot{X}_A dt + X_{A0} \end{aligned}$$

or

$$X = \left\{ X_{D0} + aT \sum_{k=1}^{m-1} \dot{X}_D^k \right\} + \left\{ a \dot{X}_D^m \tau + a \int_0^{mT + \tau} \dot{X}_A dt + X_{A0} \right\}$$

The first bracketed expression is a digital operation with "Digital Value" if  $aT = (1/2)^1$ . (Note that it contains both an integral and fractional part.) The second bracketed expression is an analog quantity for general  $\tau$ , and in general may be greater than 1 m.u.

Figure 2.2 (based on Ref. 31) shows an arbitrary function  $\dot{X}(t)$ , and illustrates the areas associated with integration.

The terms  $X_{D0}$  and  $aT \sum \dot{X}_D$  are digital in nature if  $aT = (1/2)^1$ , and thus can be generated and stored with digital precision.<sup>1</sup> Area 1 in Figure 2.2 represents  $\sum \dot{X}_D$ .

The term  $a \sum \dot{X}_D \tau$  represents the linear interpolation term due to portion of the total integral arising from  $\dot{X}_D$ ; it must be included to obtain a correct value for  $X$  during a computer run. At the end of an integral number of runs, this term is "reset" to zero and  $\dot{X}_D$  is absorbed in  $\sum \dot{X}_D$ . Area 2 in Figure 2.2 illustrates the contribution due to this term.

The term  $a \int \dot{X}_A dt + X_{A0}$  represents the contribution to the total integral from the analog portion of the input (Area 3). So long as this term remains less than one-half machine unit, it can be represented as a part of the analog output  $X_A$ . However, it is possible for the magnitude of this term to exceed 1/2 m.u.

---

<sup>1</sup>The assumed limitation of  $aT = (1/2)^1$  is no minor consideration in designing a general-purpose computer. The problem of coefficient changing will be considered later.

## 2.4 METHOD OF APPROACH: "DDA-PLUS-INTERPOLATION"

The following method which follows closely that of Skramstad (Ref. 31) will first be presented without reference to errors or other limitations; therefore the equations will be exact, and problems of errors, scaling overload, etc., will have to be considered separately. Figure 1.1(a) shows the system with incremental digital data transfer.

Symbolically, we note that at any instant of time during a computer run, we wish the sum of  $X_D$  and  $X_A$  to form the correct value of  $X$ . During holding intervals between runs, appropriate carry generating and digital updating operations are performed, but it is important to note that the total value of  $X$  before and after the holding interval should remain the same.

In general, three operations must take place during the holding interval following the  $k$ -th computer run:

1. The value of  $aT \overset{k}{X}_D$  is added to the lower orders of the  $R$  register.
2. The analog input representing a  $\overset{k}{X}_D \tau$  at  $\tau = T$  is set to zero to compensate for the addition performed in the above step. These two operations produce no net change in  $X_A$ .
3. If  $X_A$  exceeds  $1/2$  m.u. in magnitude, it must be adjusted by removing or adding 1 m.u. from  $X_A$  and

correspondingly correcting R. Simultaneously, a  $\pm 1$  m.u. increment is sent to the next computing element (carry operation).

Note that  $X_A$  is composed of four terms,  $V_1$ ,  $V_2$ ,  $V_3$ , and  $V_4$ , where

$V_1 = -E R_D$ , where  $R_D$  is the term  $\Sigma X_D - N$ ,  $R_D \leq 2$  m.u.;

$V_2 = -a X_D \tau$  - this is the interpolation voltage. During a computer run, it increases linearly with  $\tau$ , and is reset to zero during the holding period (operation 2 above).

$$|V_2| \leq E/2;$$

$V_3 = \frac{aT}{2^{n+1}} \int \dot{X}_A dt$ ,  $|V_3| \leq E/2$ ;

$V_4 = X_{A0}$ , - this is the analog portion of the initial condition on X.

$V_1$ ,  $V_2$ ,  $V_3$ , and  $V_4$  are components of the voltage  $X_A$ , but they do not necessarily exist as individual voltages in the system. This point will be clarified later.

## 2.5 THE DIGITAL SYSTEM

Figure 2.3 shows the digital portion of the hybrid integrator. All digital variables are represented in a 2's complement code (see Appendix A). The input register ( $X$ ) contains  $X_D$ , the actual digital portion of the input variable in this code. The initial value of  $X$  is selected manually; as the computation proceeds, incremental changes in  $X_D$  are made upon receipt of  $S$  ( $\Delta X$ ) and  $\Delta X$  signals from the preceding computing element. During the digital updating period,  $X_D$  is serially added to the R-register; simultaneously, the R-register is corrected by  $\pm 1$  m.u., if there

is to be a carry. The implementation of these operations is explained more fully in Section 2.9 and Appendix A. Figure 2.5 shows the timing sequence. Note that an increment  $\Delta X$ ,  $S(\Delta X)$ , is transmitted to the next computing element only if a carry is made (the carry transmitting system is also inhibited by the read-out command signal  $R0$ , to prevent incorrect read-out of the digital variables).

The rest of the digital logic is used to generate the proper signals for initiating a carry in the digital section. Flip-flops A and B store the states of the Comparators (see Section 2.6) upon receipt of the positive-going transition of signal RH.

## 2.6 THE ANALOG SYSTEM

Figure 2.4 shows the analog system which accompanies the above digital system; some of the circuit details are discussed more fully in Appendix B. Since there is no polarity inversion in the term  $\int X_A$ , the D/A converters must operate to yield a non-inverted component at the output of the  $X_A$  summing amplifier (see Appendix B). An optional inverting amplifier provides  $-X_A$ , if desired. Transistors  $Q_1$  and  $Q_2$  are used to put the integrator into HOLD, the six-diode bridge is shorted during the RESET period, to insure that the proper initial condition on  $X_A$ . Comparator A detects the condition  $X_A > 5$  volts, Comparator B detects  $X_A < -5$  volts. D/A Converter X is a four-bit bipolar unit which provides the interpolation component  $X_D$ . D/A Converter R provides the component

E.R<sub>D</sub>. Appendix B and References 34 and 36 provide a more detailed description of the Comparator and D/A Converter circuits.

The resistance values in the integrator, summer and D/A converters are chosen so that proper scaling is maintained as follows:

- a. The voltage at the output of the analog integrator should always be less than 1 m.u. in magnitude. In the experimental system, it was found that making the integrator gain equal to "a" was satisfactory ( $K=1$ ). This is usually sufficient, since in most problems, the average value of  $X_A$  over several interpolating runs is close to zero.<sup>1</sup>
- b. The component of  $X_A$  due to a  $X_D\tau$  must be less than  $\pm 1$  m.u.
- c. The component of  $X_A$  due to  $E \cdot R_D$  must be less than  $\pm 2$  m.u.

The entire analog system was designed for an overall accuracy of 1 per cent of half-scale. Calibration tests indicated that the errors in the various components of  $X_A$  were typically less than 50 mv., i.e., 1/2 per cent of half-scale.

---

<sup>1</sup>Note, however, that in certain pathological cases, e.g., integration of a small constant, the integrator gain might have to be made much smaller. In the worst case, one might have to use  $K=2(n+2)=32$ . This would correspond to a poorly scaled problem.  $K=1$  was found to be satisfactory for all experimental problems, however,  $K=2$  or 4 might provide a margin of safety for assuring that the analog integrator would not overload in general use.

## 2.7 POLARITY INVERSION

The integrator may be operated in the inverted mode by using an analog unity-gain inverter to invert  $X_A$  and logically inverting  $S(\Delta X)$ ; this method is, of course, much simpler than using a separate inverting component.

## 2.8 INTERPOLATING WAVEFORMS

The ramp interpolating waveforms used to drive the "X" D/A Converter are supplied by a separate waveform generator, which can thus service a number of integrators simultaneously. This unit is described in Ref. 35. It provides 1 per cent-linear positive and negative 10 v ramps which are reset to zero in about 20 microseconds after the HOLD interval  $T_H$  begins.

## 2.9 TIMING SEQUENCE

Figure 2.5 shows the timing sequence of the integrator. All of the operations during the HOLD interval are controlled by a subroutine clock which can drive a number of integrators simultaneously. The sequence of operations is as follows:

- a. At the beginning of the HOLD interval, the analog integrators are put into hold. This is accomplished by the RH signal, which turns on  $Q_1$ ,  $Q_2$ .
- b. Simultaneously, the positive-going transition of RH sets Flip-flops A and B to store the states of the Comparators. This immediately determines whether or

not a carry is to be made, as indicated by signals C and  $S_C$ .

- c. Shortly after the HOLD interval begins, timing pulses  $T_1$ - $T_4$  perform a serial addition of  $X_D$  into the lower four states of the R-register.
- d. Pulses  $T_5$  and  $T_6$  simultaneously correct the upper two states of the R-register to complete the addition of  $X_D$  and to add or subtract 1, if a carry is required (see Appendix A).
- e. Pulse  $T_7$  transmits a  $\Delta X$  signal to the next computing element, if a carry is required. The state of the  $S(\Delta X)$  line tells the next element the polarity of the carry.

The timing pulses  $T_1$  to  $T_7$  are 5 microseconds apart; thus the entire digital operation is completed in about 35 microseconds. During this time, the sawtooth waveforms, which are supplied to the "X" D/A Converter are resetting to zero (this requires about 20  $\mu$ sec); thus the operations of updating the R-register, resetting the interpolation signal, and when required, initiating the carry are all completed in about 35 microseconds. Allowing another 15  $\mu$ sec for analog transients to settle, it is possible to use a HOLD interval  $T_H$  of 50 microseconds, or about 4 per cent of the computing interval,  $T$ . Additional hold time is required for other digital operations associated with such elements as summers and multipliers, since they must wait for the receipt of carries from integrators prior to the

initiation of their digital operations. Therefore, the prototype system includes an optional 10 per cent (125  $\mu$ sec) holding interval, to permit the inclusion of these extra sequences at a later date.

#### 2.10 CONTROL CLOCKS (See also Ref. 18)

The computing sequence is controlled by a digital clock system which provides the waveforms shown in Figure 2.5. This system uses an 80 kc crystal and 1 mc transistorized digital logic modules to provide precise timing of the analog computing interval. It provides a pre-selectable read-out time,  $T_T$ , which may be set from 0.01 to 999.99 T. This read-out signal, R0, places the entire computing system into a terminal HOLD and also commands read-out of the analog problem variable (see Section 3.11). Table 2.1 summarizes the basic clock parameters.

#### 2.11 COMPUTER READ-OUT (See also Ref. 37)

In order to provide a useful read-out of machine variables, the hybrid differential analyzer is equipped with a precisely-timed read-out system, which performs the following functions:

- a. Upon receipt of a read-out signal (positive-going transition of the R0 signal from the master control clock), a fast analog sample-hold circuit stores the value of the desired analog machine variable. Simultaneously, a digital voltmeter is commanded to convert this voltage to a digitally displayed value for  $X_A(T_T)$ .

- b. During the computer run, the digital portion of the read-out system has been receiving incremental information about the value of a particular digital variable,  $X_D$ . When read-out is commanded, the digital system will retain the value of  $X_D$ , and display it (in this system as a 2's complement binary number).
- c. For diagnostic and display purposes, a full-analog read-out of the sum of both  $X_D$  and  $X_A$  versus either real time or machine time is provided. Also  $X_D$  and  $X_A$  may be displayed separately.

The read-out system used in the prototype system is described in Ref. 37. It has accuracy of better than 25 mv on read-out of  $X_A$  (i.e., 0.03 per cent of half-scale of the hybrid variable). Full analog read-out of  $(X_D + X_A)$  can be made with an accuracy of 1 per cent of full-scale. Read-out timing accuracy is  $\pm 1/4$  micro-second, which provides a negligible effect on the overall accuracy.

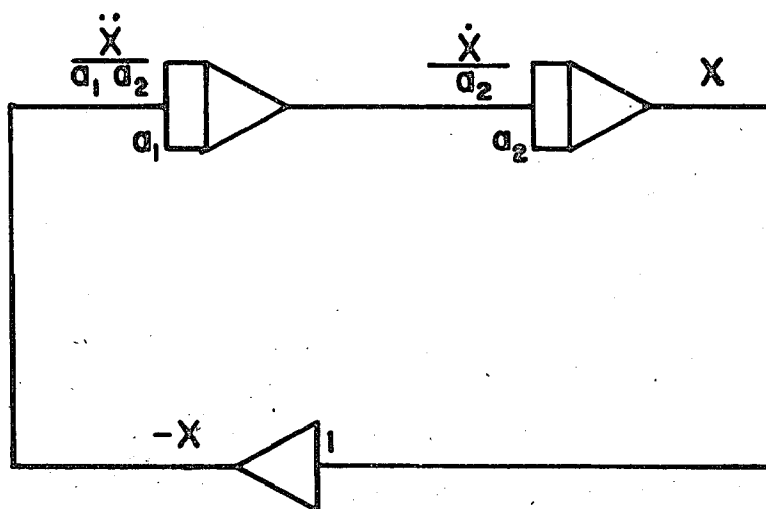
TABLE 2.1

CONTROL CLOCK PARAMETERS  
(See Also Ref. 18)

---

Analog Computing Interval, $T$ :	1250, 12,500, or 125,000 microseconds
Analog Hold Interval, $T_H$ :	25, 50, 125, or 250 microseconds
Total Computing Time, $T_T$ :	$T_T$ 0.01 $T$ to 999.99 $T$ , Selectable to nearest 0.01 $T$
Time Accuracy	
$T$	0.01 Per Cent of Nominal
$T_T$	0.01 Per Cent of Nominal

---



$$a=1; \quad n=3; \quad a \leq 50; \quad T=1.25 \text{ ms}$$

$$|X| \leq 7.5 \angle 8$$

Figure 2.1 Sine Loop Configuration

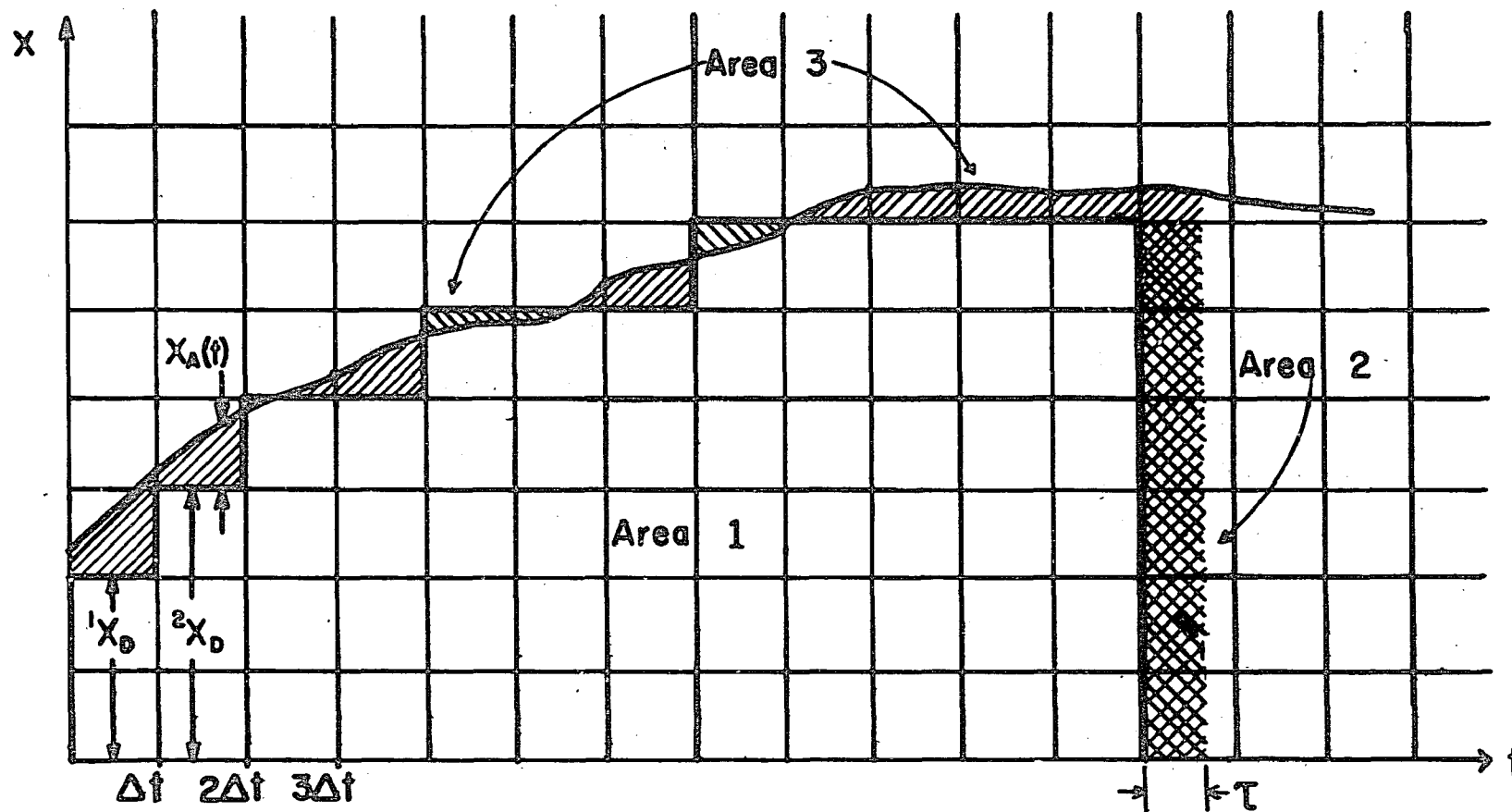
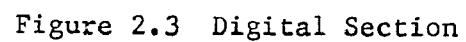


Figure 2.2 Graphical Representation  
of Hybrid Integration (from Skramstad, Ref. 31)



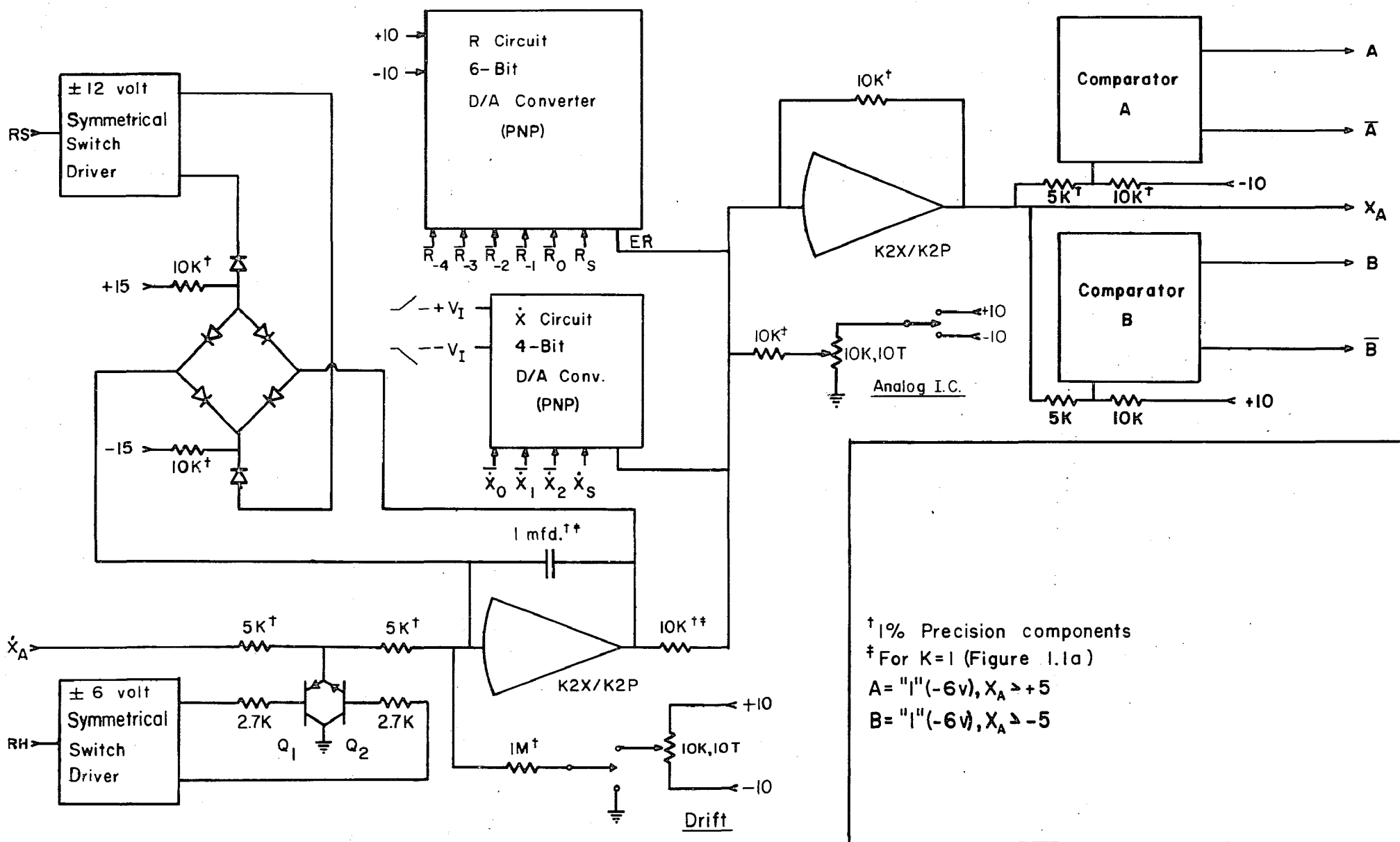


Figure 2.4 Analog Section

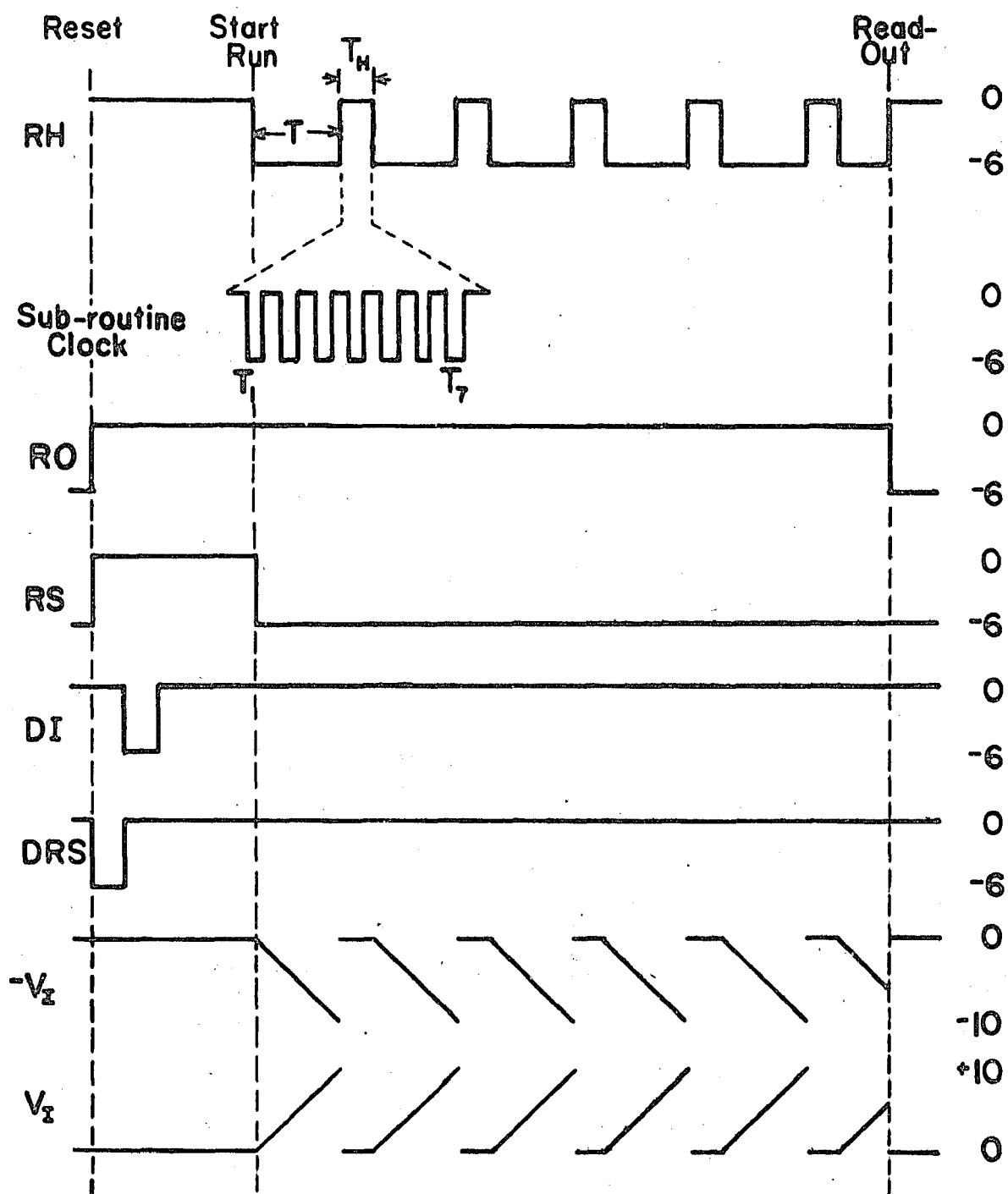


Figure 2.5 Timing Sequence

## Chapter 3

### OTHER HYBRID COMPONENTS

#### 3.1 ZERO-MEMORY ELEMENTS

This chapter describes the general design considerations for the zero-memory hybrid operational elements; these include components for summing, coefficient changing, multiplying, and function generation. These elements perform their digital operations subsequent to the updating of all hybrid integrators. They are all considered to be zero-memory elements, i.e., at any instant, the value of the output variable is related to the instantaneous value of the input variable(s).<sup>1</sup> If several such elements are connected in cascade, the digital and carry generating operations should proceed forward from the first element following an integrator, until all zero-memory elements in a cascade chain have been updated successively.

#### 3.2 SUMMING

Figure 3.1 shows a block diagram of a summing component to

form

$$Z(t) = Z_D + Z_A = \frac{(X_D + X_A) + (Y_D + Y_A)}{2}$$

---

<sup>1</sup>Note that this is not quite a simple one-to-one mapping, since each value of a hybrid variable may be represented in two different forms, e.g.,  $X = 6.3$  can be represented as  $X_D = 6$ ,  $X_A = 0.3$  or  $X_D = 7$ ,  $X_A = -0.7$ .

In Appendix C, it is shown that the essential digital operation is the formation of the digital quantity

$$R = \frac{X_{D0} + Y_{D0}}{2} - Z_{D0} - N + 1/2 \sum (\Delta X_D + \Delta Y_D)$$

Note that R is a three-digit number, with a value ranging from  $-1 \frac{1}{2}$  to  $+1 \frac{1}{2}$  m.u. It represents the net effect of the incoming increments,  $\pm \Delta X_D$  and  $\pm \Delta Y_D$ , and the outgoing carries  $\pm \Delta Z_D$ . The term  $(X_{D0} + Y_{D0})/2 - Z_{D0}$  is the fractional part of the initial sum  $(X_{D0} + Y_{D0})/2$ .

Subsequent to receipt of digital increments the state of the R-register, together with  $X_A$  and  $Y_A$ , may temporarily attempt to cause the output of the summing amplifier to exceed 1 m.u. A double-anode Zener diode will prevent amplifier overload until subsequent correction of the R-register by a 1 m.u. carry brings the summing amplifier amplitude below  $1/2$  m.u. Proper scaling insures that no overload occurs during an analog computing run. The determination of a carry is made at a time when all analog variables are constant; thus no digital storage elements are required following the comparators. Timing pulse  $T_0$  initiates the addition or subtraction of 1 m.u. from the R-register (a simple three-digit up-down counter). Note that this assumes  $\Delta X$  and  $\Delta Y$  are received from integrators. If these digital increments are from some intervening zero-memory element, then the timing of this carry-initiating pulse will follow the digital operations in the preceding element. Using the present

system components, the total time required for performing the carry operation in one summer would be about 20 microseconds.

### 3.2.1 SUMMING AT INTEGRATOR INPUTS

Note that digital summation may also be performed at the input to an integrator by providing separate inputs for each input variable. This requires separate carry-transmitting pulses for each incremental digital input. Proper scaling insures that there is no need for additional carry operations within the integrator, since it is the integral of the input, rather than the value of the input that affects the integrator output magnitude.

### 3.3 MULTIPLICATION (See Ref. 17, 29, 31)

The operation involved in hybrid multiplication and coefficient changing are essentially the same, except that in the latter case, the coefficient is a fixed hybrid constant  $C = C_D + C_A$ . In the following discussion, the general problem of hybrid multiplication will be treated with the implication that if the input  $Y = Y_D + Y_A$  is a simple constant, equipment simplifications then result.<sup>1</sup>

Consider the operation

$$\begin{aligned} Z &= Z_D + Z_A = \frac{(X_D + X_A)(Y_D + Y_A)}{2^n} = 1/8(X_D + X_A)(Y_D + Y_A) \\ &= 1/8 [X_D Y_D + X_D Y_A + X_A Y_D + X_A Y_A] \end{aligned}$$

---

<sup>1</sup>Note that the complexity of coefficient setting is no less of a problem in all-digital differential analyzers.

Appendix C presents a derivation of the digital operations required to implement multiplication, using incremental digital transfer. Figure 1.1(b) is a general block diagram of a hybrid multiplier; and it illustrates the basic operations required.

As in all other hybrid components, there is an R-register, which forms a digital remainder to be fed to the analog summing amplifier for determination of carries. The R-register forms

$$R = 1/2^n \left[ Y_D \Delta X_D + \Sigma X_D \Delta Y_D - \Sigma \Delta X_D \Delta Y_D - N \right]$$

Note that again  $|R| \leq 2$  m.u. Various digital schemes could be used to implement the required operations, depending upon desired operating time, number of bits, etc. A detailed discussion of this problem is not included in this work.

Using simple counter-registers with serial information drop-in, as in the previously discussed systems, the total updating and carry generating time for a four-bit hybrid multiplier would be about 100 microseconds (using the same parameters as before).

The hybrid multiplier requires a fast analog multiplier to form the term  $X_A Y_A$ . Note that its accuracy would not have to be high in a hybrid system with a large number of digital bits; indeed, the analog multiplier might even be dispensed with. However, in a four-bit hybrid system, the analog multiplier is required, and it should have an accuracy of better than 8 per cent, to maintain consistent accuracy with the other hybrid components thus far discussed.

### 3.4 COEFFICIENT CHANGING

The requirements for a hybrid coefficient-changing component follow directly from the above. Some simplifications are now possible (see Figure 3.2). Specifically, the analog multiplier is replaced by a potentiometer, the  $Y_D$  register is replaced by a manually-settable group of digital lines and the D/A converter for  $X_D C_A$  now receives only constant analog inputs  $C_A$  and  $-C_A$ . The R-register is also simpler, since it is now used to form

$$R = 1/2^N \sum C_D \Delta X_D^{-N}$$

since  $\Delta Y_D = \Delta C_D$  is now zero.

### 3.5 GENERATING FUNCTION OF ONE VARIABLE

In Appendix D, some of the requirements for a hybrid function generator are discussed, and a brief error analysis is made. Considerable prior work has been done in this area, with the object of developing hybrid function generators for use with conventional analog computing systems (Ref. 1, 4, 17, 28, 29, 30). For use in a true hybrid system, using incremental digital transfer, the system organization is somewhat different. Figure 1.1(c) shows a typical block diagram for a hybrid function generator. Unlike all previously discussed components, the choice of the number of digital bits determines the complexity of operations. The function will in general be generated by performing interpolations about digitally located values of the independent variable, using a

Taylor's series approximation. If fewer digital bits are used, then higher-ordered terms in the approximation may be required to achieve a given accuracy. To maintain an accuracy consistent with the other computing elements in a four-bit system, second-order interpolation terms might be required for some functions.

Another added complication in function generators is that digital functions must be formed with sufficient precision to match the full accuracy of the analog system. For example, a system with four bits and 1 per cent analog components requires that  $\Delta F(X_D)$  be generated with a precision of 7-8 bits; these lower-order bits are converted to an analog signal for the summing amplifier. Note that it is also not possible to improve the accuracy of function approximation by using unequal spacing of digital points, since they must be equally spaced in order to agree with the input variable coding shown in Figure 1.2.

Subject to the above considerations, it should be possible to develop a hybrid function generator to be used with the type of system discussed in this work; considerably more work should be done to determine an optimum method for implementing the required operations.

It is readily apparent that all of the above systems are organized on a similar basis. They contain an R-register for forming digital increments of the output variable, which normally

contains digital fractional parts of a machine unit and has magnitude strictly less than two. There are also D/A converters or multipliers and conventional analog elements. Each component contains two comparators and associated logic to perform carries by correcting the R-register and transmitting an increment to the next computing element. This general organization can be used to develop a broad class of hybrid operational elements. Of particular interest is a hybrid resolver, which would belong to a class of components for generating a factorable function of two variables, i.e.,

$$Z = F(X) G(Y)$$

In the case of the resolver,  $F(X) = R$ ,  $G(Y) = \cos \theta$  or  $\sin \theta$ . It would probably prove desirable to build a single resolver component than to implement the function using a multiplier and three separate function generators.

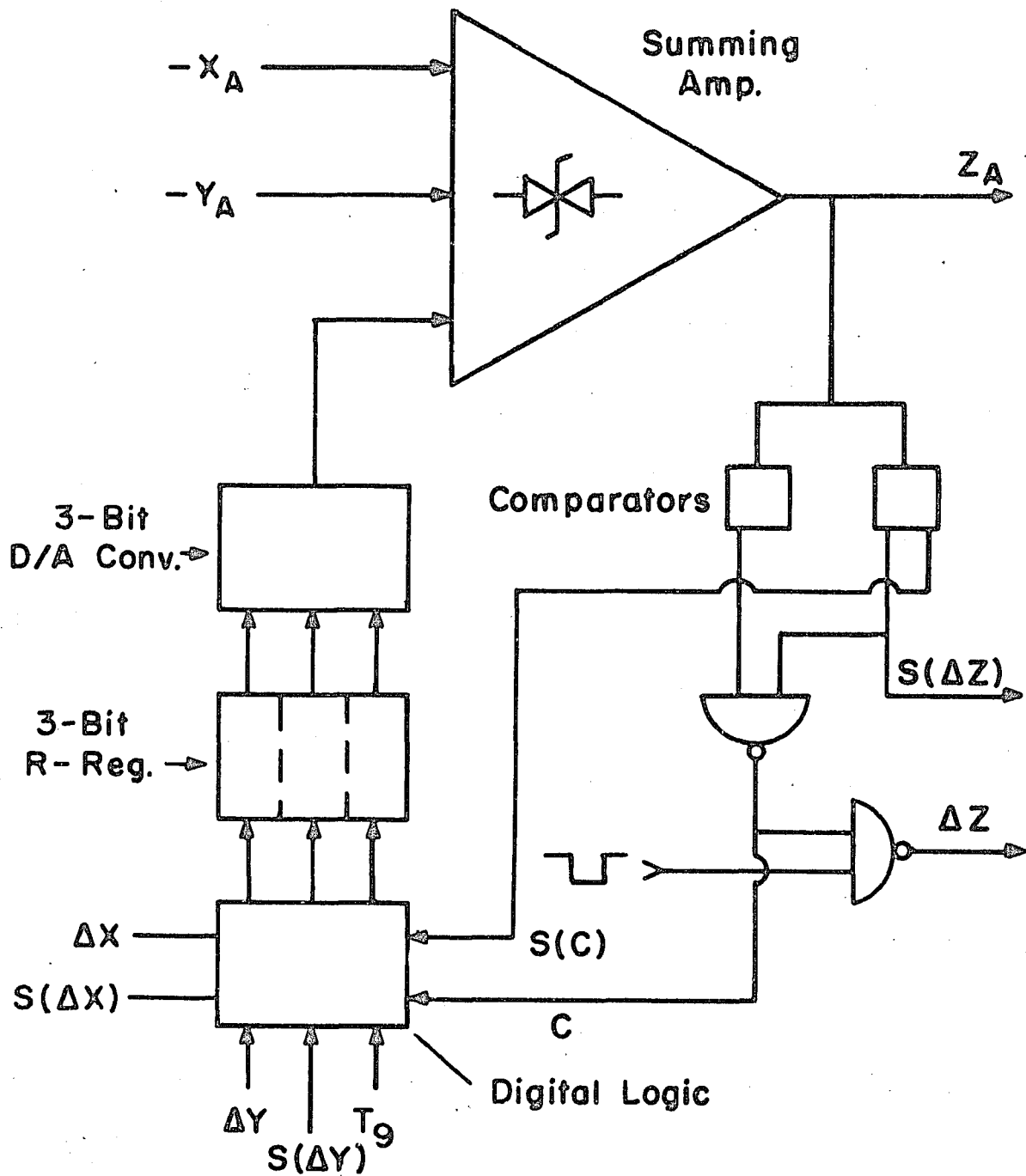
### 3.6 HYBRID DIVISION

Hybrid division of a constant may be accomplished by using a special function generating component to generate:

$$(Y_D + Y_A) = \frac{C}{(X_D + X_A)}$$

This would permit dividing a constant by the machine variable  $X$ . To divide one machine variable by another, one could cascade the above operational unit and a hybrid multiplier to form:

$$Z_D + Z_A = \frac{C}{(X_D + X_A)} \cdot (Y_D + Y_A) = \frac{C(Y_D + Y_A)}{(X_D + X_A)}$$



Note:  $\Delta X$  and  $\Delta Y$  pulses occur at different times, e.g.  $T_7$  and  $T_8$

Figure 3.1 Summing Component

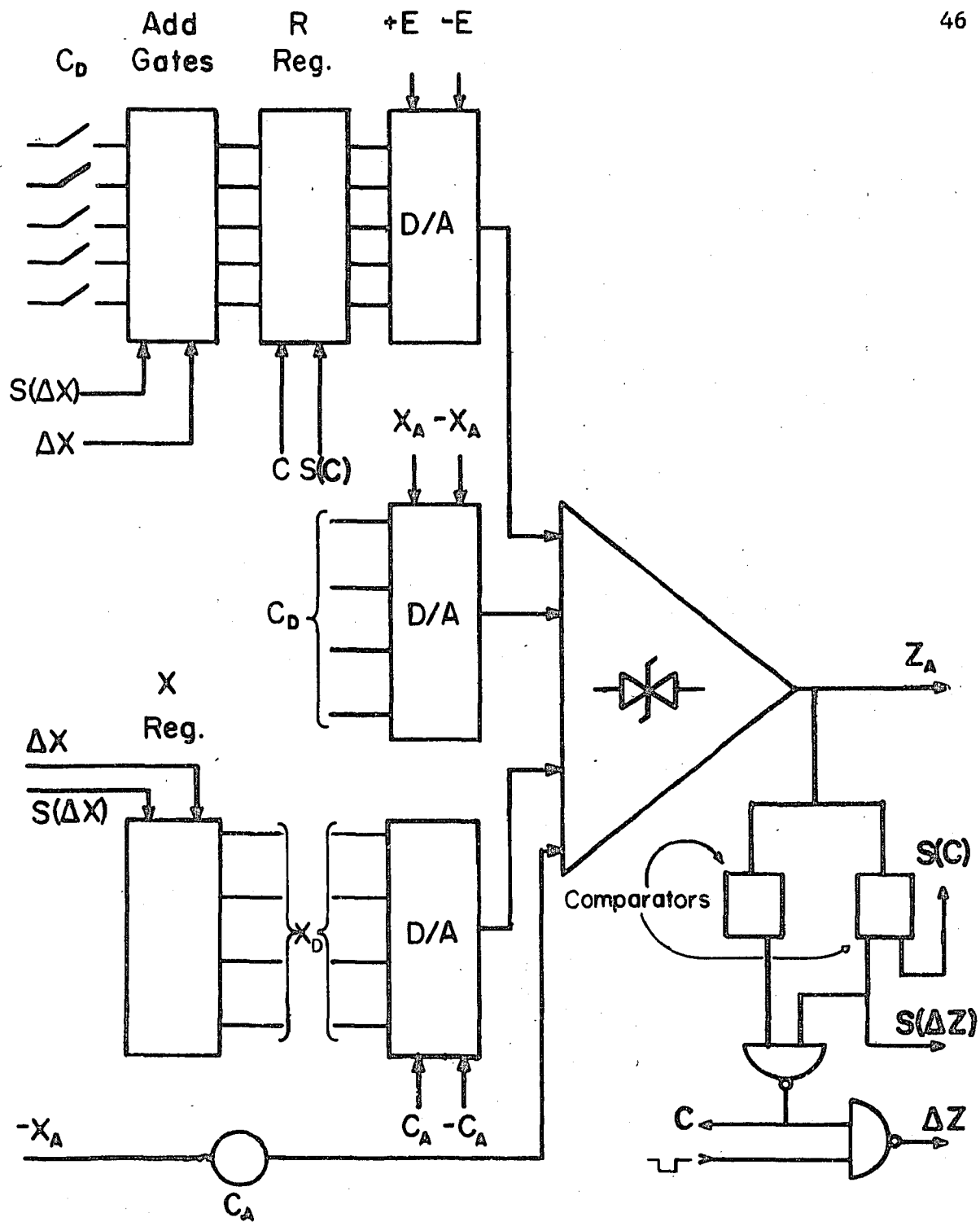


Figure 3.2 Coefficient Changer

## Chapter 4

### EXPERIMENTAL RESULTS

#### 4.1 EXPERIMENTAL PROCEDURE

Two hybrid integrators, with associated control and read-out equipment were used in three simple computing configurations:

- a. Free-Fall Parabolic Trajectory (second integral of a constant).
- b. Decaying exponential (first-order linear differential equation).
- c. Undamped sinusoid (second-order system without damping).

Each problem was scaled to approach the maximum amplitude and rate limitations of the system.

Figure 4.1 shows an analog display of the three problem solutions; also shown are the two solution components,  $X_D$  and  $X_A$ . Figure 4.2 shows  $X_A$  only, on an enlarged scale.

The equipment was carefully calibrated prior to the experimental computing runs; however, during the experiments, no detailed recalibration was performed, with the exception of periodic checks of the sample-hold circuit and digital voltmeter in the

read-out chassis, and an occasional check of the sawtooth generator amplitude. Table 4.1 summarizes the results of the initial calibration tests.

TABLE 4.1

Maximum Error in any D/A Converter:	60 mv
Typical Average Error in a D/A Converter:	20 mv
Maximum Error in Sawtooth Linearity:	40 mv
Integrator Drift (with drift-correction adjustment):	10 mv/sec
Error in Length of Analog Computing Interval, T:	0.01 o/o

To test the sensitivity of the system to errors, various artificial errors were introduced. Table 4.2 and Figures 4.3 to 4.5 summarize the results of these tests; Figures 4.6 to 4.9 provided additional insight into the performance of the system.

#### 4.2 FREE-FALL PARABOLIC TRAJECTORY

##### 4.2.1 NOMINAL MACHINE SOLUTION

The first computer problem was the solution of the differential equation:

$$\ddot{x}(t) = -32.2 \text{ ft/sec}^2; \quad x(0) = 0 \text{ ft}; \quad \dot{x}(0) = 600 \text{ ft/sec}$$

with solution

$$x(t) = 600 t - 16.1 t^2 \text{ ft.}$$

Maximum Height: 5590.092 ft.

Time to Impact (zero crossing): 37.267 sec.

This equation, of course, simulates an elementary flat-earth vacuum trajectory problem, and is of interest because it involves an open-loop computation, where cumulative errors should become readily apparent.

With X measured in volts and time in units equal to one analog computing period, the actual machine equation used was:

$$\ddot{X} = -32.2 \text{ volts; } X(0) = 0; \dot{X}(0) = 60 \text{ volts} = 6 \text{ digital m.u.}$$

with theoretical solution

$$X = 3.75 k - \frac{322}{512} k^2 \text{ volts}$$

Maximum Height: 55.9092 volts = 5.591 m.u.

Time to Impact: 59.627 intervals = 0.074534 sec. of machine time

One volt = 100 ft.

One Computing Interval = 1.25 ms = 0.625 sec. of problem time

$$\alpha_t = 500$$

Figure 4.3a shows a plot of solution error versus "time" in computing intervals. Some of the more pertinent results were:

a. Error in impact time: 0.02 per cent

- b. Error in maximum height: 2 ft. (0.04 per cent of theoretical value).
- c. Maximum measured machine error: 70 mv (0.09 per cent of half-scale machine range).

#### 4.2.2 ERRORS DUE TO REMOVAL OF SAWTOOTH INTERPOLATION AND ANALOG INTEGRATOR

Figures 4.4a and 4.5a shows the effect on the computer solution caused by removing the sawtooth interpolation and/or the analog integrators. When both are removed, a necessarily crude digital solution results. The case where the sawtooth only is omitted yields a smaller solution error than when the sawtooth is included, but the analog integrator is not. This would be expected, since when the analog integrators are absent, the analog portion of the input constant cannot be introduced into the first integration.

#### 4.2.3 EFFECTS OF ARTIFICIAL ERRORS

Figures 4.5b and 4.5c show the solution errors generated by various small artificial errors in the analog channel. Note that the five different error sources had about the same order of magnitude of effect. However, a number of conclusions about open-loop computation are suggested:

- a. As expected, a relatively large solution error arises if both analog integrators are reduced in gain; however, most of the error is generated in the first integration. Additional experiments confirmed the fact that out of a total solution error at impact of about 1.7 volts, only

about 0.5 volts error was caused by the loss of gain in the second integrator.

- b. An error in the gain of the summing amplifiers tends to produce slightly more error than an equivalent loss in sawtooth amplitude.
- c. The error due to the substitution of the finite one-second integrator time constant was relatively small, in this case, never exceeding 0.57 volt.
- d. The error due to degraded high-frequency response in the summing amplifiers appears to be greatest at times when the rate-of-change of the problem variable is greatest, but the overall error does not seem to be cumulative.

#### 4.3 DECAYING EXPONENTIAL

##### 4.3.1 NOMINAL MACHINE SOLUTION

The first closed-loop problem was the simple first-order linear differential equation

$$\dot{X} = -50 X; \quad X(0) = 60 \text{ volts}$$

with solution

$$X = 60 e^{-50t}$$

Stated in terms of computing intervals, the problem becomes

$$\dot{X} = -\frac{X}{16}; \quad X(0) = 60 \text{ volts (6 digital m.u.)}$$

with solution

$$X = 60 e^{-k/16}$$

One Computing Run = 1.25 ms of machine time

One Time Constant = 16 computing intervals

Figure 4.3b shows a plot of the nominal solution error; it indicates that the maximum error in the solution was 0.05 volt (0.07 per cent of half-scale).

#### 4.3.2 COMPARISON WITH STRAIGHT DIGITAL SOLUTION

Figures 4.4b and 4.5d show the effect on the solution caused by removing the sawtooth interpolation and/or the analog integrators. Note that the final value of the digital solution is different from zero. Maximum solution error was less than 3.34 volts, which corresponds to about 1/3 of a digital m.u., and agrees fairly well with what one might estimate on the basis of Section 5.3. Addition of the analog integrator reduced the maximum error to 1.9 volts. Removing the analog integrator and introducing the sawtooth interpolation produced a straight-line-segment approximate solution with a maximum error of 4.53 volts, actually slightly more than the straight digital solution.

#### 4.3.3 EFFECTS OF ARTIFICIAL ERRORS

Figure 4.5e shows the effect of linear gain errors in the computing system. The reduction of the summing amplifier gain causes

considerably more error than reducing either the analog integrator gain or the sawtooth amplitude. Note, however, that none of the error curves indicate a net steady-state offset, as produced in a straight digital solution, i.e., the solution still eventually decays to zero. The error in the summing amplifier gain tended to produce an oscillating solution error that alternates in magnitude and polarity; a maximum error of 450 mv (0.6 per cent of half-scale) was observed. A 10 per cent error in either the analog integrator gain or sawtooth amplitude produced solution errors of similar size; in this case, never more than 160 mv (0.21 per cent of half-scale).

Figure 4.5f shows the effects of poor amplifier and integrator response. As expected, poor high-frequency response in the summing amplifier introduces a fairly large error (as high as 0.63 volt) during the initial part of the solution, when the rate-of-change of  $X_A$  is high. Interestingly, the effect of poor low-frequency response in the analog integrator is not great. The use of a one-second integrator time constant produced a maximum error of 0.07 volts, which is less than the overall design accuracy of the system. It appears from this test (and also from later results) that it is essential to maintain good high-frequency response in the system. However, good low-frequency response in the analog integrator is not essential, which suggests that a passive RC network could be used in place of the active one used in these experiments, if less analog accuracy is acceptable.

Figure 4.6 shows the error resulting from a 10 micro-amp. offset current in the analog integrator. The dotted line is the

theoretically calculated error curve that would result from the same offset in an analog integrator. Except for normal random variations in solution values due to other effects in the system, the experimental error curve is qualitatively the same and indicates a steady-state final value of 0.2 v, as predicted.

#### 4.4 UNDAMPED SINUSOID (Circle or Sine-Loop Test)

##### 4.4.1 NOMINAL MACHINE SOLUTION

Two hybrid integrators and an inverter were used to study the simple undamped second-order differential equation:

$$\ddot{X} = -2500 \cdot X$$

with solution

$$X = X(0) \cos 50 t + \dot{X}(0) \sin 50 t$$

Stated in terms of computing intervals:

$$\ddot{X} = \frac{-X}{256}$$

with solution

$$X = X(0) \cos 1/16 k + \dot{X}(0) \sin 1/16 k$$

$$\text{One Period} = 100.26 \text{ computing intervals}$$

$$\text{One Period} = 0.1253 \text{ sec. of machine time}$$

$$\text{One Period} = 0.1303 \text{ sec. of real time (7.67 cps)}$$

Most of the experiments were done with either  $X(0) = 60$ ,  $\dot{X}(0) = 0$ , or  $X(0) = 0$ ,  $\frac{\dot{X}(0)}{50} = 60$ ; i.e.,  $X = 60 \cos 50 t$  or  $60 \sin 50 t$  volts.

Figure 4.3c shows that the maximum solution error was 0.13 v (0.17 per cent of half-scale), but that the errors were typically much smaller. The average error over one cycle is 0.017 volt, with an rms error of 0.061 volt.

The nominal solution exhibited a slight exponential build-up, which could be measured by observing the solution over several cycles. Figure 4.7 shows a plot of the extrema of the solution measured at the end of every half-cycle of the cosine waveform for nine full cycles. The amplitude of the waveform increased approximately 0.06 volts/cycle. From the equations derived in Appendix F, it can be shown that  $\alpha \simeq 0.0075$ , or equivalently,  $\alpha/\omega = 1.5 \times 10^{-4}$ . The zero-crossing time at the end of five complete cycles of the sine wave occurred at 502.53 computing intervals (theoretically 502.65 intervals), indicating that the natural frequency was low by 0.02 per cent ( $2 \times 10^{-4}$ ). Thus the overall accuracy of location of the system poles in the complex plane was about 0.025 per cent. This is considerably better than was originally expected. Note that the effect of the error in system pole location causes the absolute error in the solution to increase with time, as is the case with analog and digital solutions of the same differential equation.

#### 4.4.2 COMPARISON TO STRAIGHT DIGITAL SOLUTION

Removal of the sawtooth interpolation and the analog integration sections of the system yields a solution with a rapid exponential build-up, due to the truncation error in the resulting

rectangular integration process. Figure 4.4c compares the digital solution to the full-hybrid solution; the amplitude of the oscillation increases approximately 20 volts/cycle. Also shown is the improvement provided by adding the analog integrator (still no sawtooth); in this case, the solution still grows about 3 volts/cycle. Adding the sawtooth interpolation without the analog integrator yields a worse solution, which builds up about 7.5 volts/cycle (not shown).

#### 4.4.3 EFFECTS OF ARTIFICIAL ERRORS (Root Perturbation)

In this computing configuration, the effect of various artificially-introduced system errors can be readily studied by noting the resultant displacement of the characteristic roots of the system (as indicated by changes in the natural frequency and exponential build-up of the solution). Using the equations of Appendix F, it is possible to estimate the error in root location by measuring the solution errors at the extrema and at the zero-crossings. As a double check, the natural frequency errors were studied by two methods: viz., measurement of solution error at the theoretical zero-crossing times and measuring the actual zero-crossing times of the computer solution. Both methods gave almost identical results.

The data in Table 4.3 resulted from tests made after five cycles of computation, where the accumulative errors due to root perturbations are sufficiently large to overcome any short-term random variations. The real part of the system roots were

measured by examining the build-up (or decay) of the amplitude of the solution, whereas the natural frequency of the solution was examined by the two methods listed above.

Table 4.4 shows the sensitivity of the root location to changes in gain of the summing amplifier and analog integrator and changes in the sawtooth amplitude. Note that the natural frequency of the computing configuration is relatively insensitive to errors in the analog system. The system damping, on the other hand, is considerably more sensitive to analog errors. This suggests that the various analog errors tend to introduce an equivalent phase shift into the hybrid integrator, but do not appreciably affect the gain. This is further borne out by additional experiments which indicated that increases in the summing-amplifier or integrator gains or in the sawtooth amplitude all produced a build-up of the solution (positive  $\alpha$ ), whereas, corresponding decreases produce a decay of the solution (negative  $\alpha$ ). Figure 4.8 shows the effect of  $\pm 10$  per cent changes in the summing amplifier feedback resistor, and illustrates the effect of these changes on the exponential build-up or decay of the solution.

The effect of DC offsets and drifts on the computer solution were again similar to those to be expected in a conventional analog system. One additional observation is that drifts in the analog integrators are not cancelled by the computing loop, as they would be in a pure analog system, so that

they can cause an internal overload of the analog integrators during a long computing run. Increasing the factor  $K$  (Figure 2.4) would provide a means for preventing this effect, if required.

#### 4.5 EFFECT OF THE HOLD INTERVAL

Thus far, nothing has been said about the effect of the duration of the HOLD interval on the accuracy of the computer. In addition to the normal HOLD interval,  $T_H$ , of 50 microseconds, intervals of 125 and 250 microseconds were also used in several experiments. No noticeable effect on the computer performance was observed. Moreover, a smaller value of 25 microseconds caused little effect, except a slight disturbance in the read-out system due to the fact that the switching transients resulting from the digital updating operations occasionally propagate into the beginning of succeeding analog computing intervals. Figure 4.9 shows an expanded view of a typical waveform for  $X_A$ , and illustrates the appearance of the holding and carry resetting operations.

#### 4.6 COMPARATOR ACCURACY

Figures 4.2 and 4.9 show that the accuracy of the analog comparator is not critical. In fact, since most operational amplifiers will not overload until their nominal computing range is exceeded by perhaps 10 per cent, a comparator accuracy of  $\pm 1/2$  volts would suffice.

TABLE 4.2

## EFFECT OF ARTIFICIAL ERRORS ON COMPUTER SOLUTION: A SUMMARY

(See Also Figures 4.3 to 4.8 and Tables 4.3 and 4.4)

Type of Artificial Error	Problem Error			Remarks	
	Falling Body (Parabola)	Decaying Exponential	Sinusoid		
None	0.07 v max.	0.05 v max.	0.13 v max. 0.06 v RMS	Normal errors typically 0.1 v	
Straight Digital	28.7 v error at impact	3.3 v error in final value	20 v/cycle build-up	Computer errors generally larger when analog integrator is absent, than when sawtooth is absent	
Sawtooth, No Integrator	19.9 v	4.53 v	7.5 v/cycle		
Integrator, No Sawtooth	7.3 v	Small Final, Large Initial Errors	3 v/cycle		
Analog Gain Errors			$\frac{o/o}{\Delta\alpha/\omega}$ $\frac{o/o}{\Delta\omega/\omega}$		
90 o/o Summing Amp.	0.8 v max.	0.45 v max.	0.44	0.05	Sawtooth accuracy generally less important than amplifier and integrator gain
90 o/o Integrator Gain	1.7 v max.	0.16 v max.	0.30	0.065	
90 o/o Sawtooth	0.6 v max.	0.17 v max.	0.30	0.03	
Poor High-Freq. Resp.	0.79 v max.	0.34 v max.	0.07	0.02	Good High-Frequency Response more important
Finite Integrator Time Const.	0.57 v max.	0.09 v max	0.01	0.055	

TABLE 4.3

SOLUTION ERRORS CAUSED BY ERRORS IN ANALOG CHANNEL:  
SINE-WAVE PROBLEM

Parameter Change	Damping Effect $\alpha/\omega$ , Per Cent	Frequency Error <sup>*</sup> $\Delta\omega/\omega$ , Per Cent	Remarks
10 Per Cent Reduction in Summer Gain,			
one:	0.19	-0.01	Average
both:	0.44	-0.05	
10 Per Cent Reduction in Int. Gain,			
one:	0.15	-0.007	Average
both:	0.30	-0.065	
10 Per Cent Reduction in Sawtooth	0.30	-0.03	
DC Offset	Negligible	Negligible	
Slow Amplifiers	0.07	-.02	5° Phase Shift at 60 cps
Lossy Integrators,			
1 sec. T. C.	0.01	-0.055	
0.1 sec. T. C.	0.05		

<sup>\*</sup>Frequency Error Measure Two Ways, Average Listed.

TABLE 4.4

ROOT SENSITIVITY TO ANALOG CHANNEL ERRORS:  
SINE-WAVE PROBLEM

Parameter Change	Damping Sensitivity* Per Cent/Per Cent	Frequency Sensitivity* Per Cent/Per Cent
Summer Gain		
one:	0.019	-0.002
both:	0.044	-0.005
Integrator Gain		
one:	0.015	-0.0007
both:	0.030	-0.0065
Sawtooth Amplitude	0.03	-0.003

\*Sensitivity Figures are given as a ratio of the per cent change in the root location to the per cent change in the nominal value of the parameter.

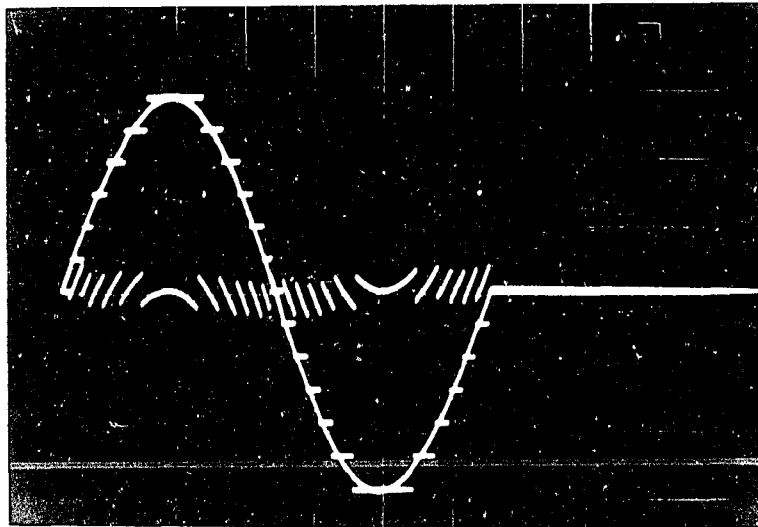
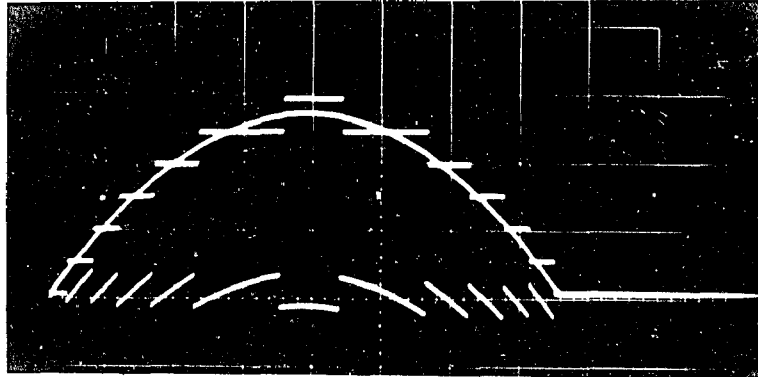


Figure 4.1'  $X_D$ ,  $X_A$ , and Their Sum

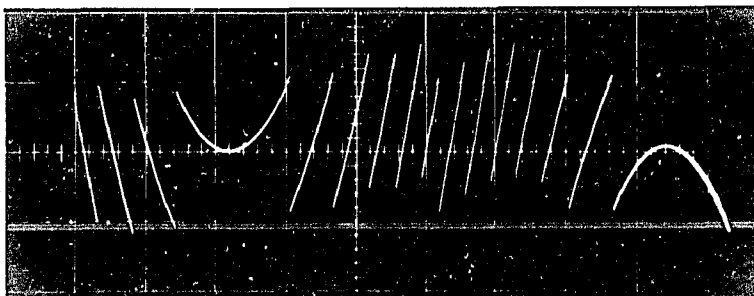
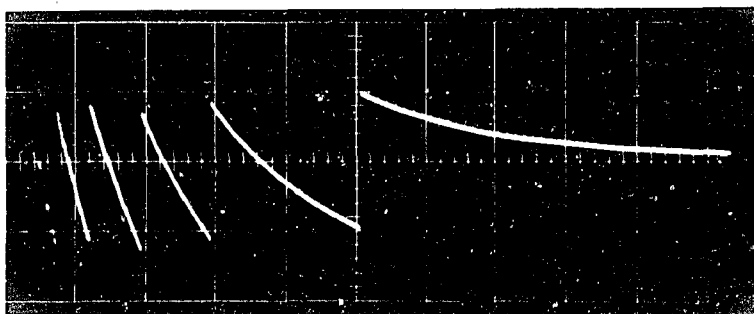
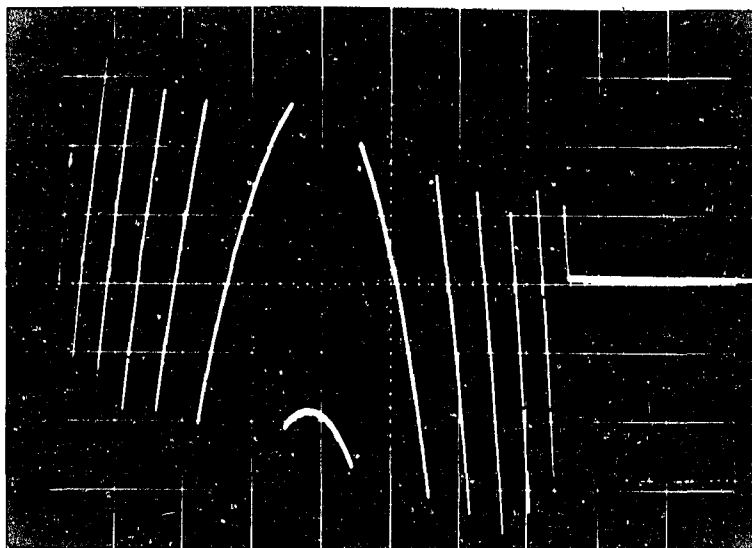


Figure 4.2  $X_A$  Only

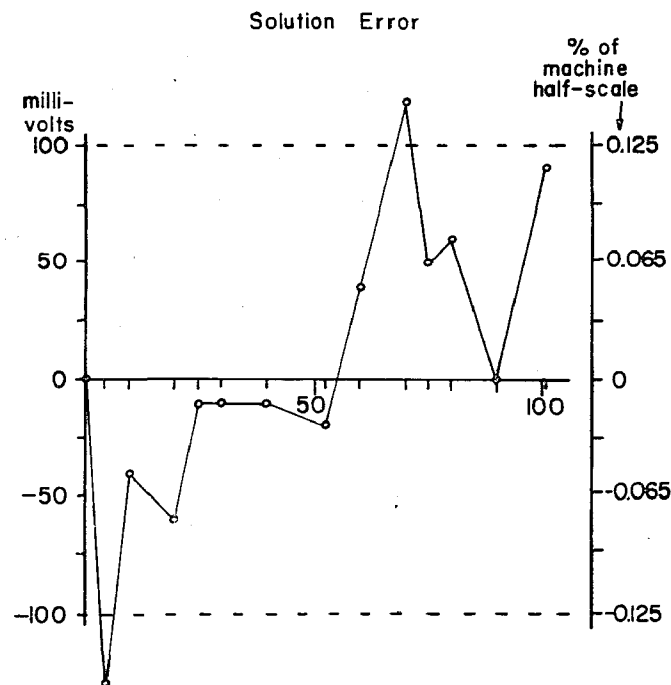
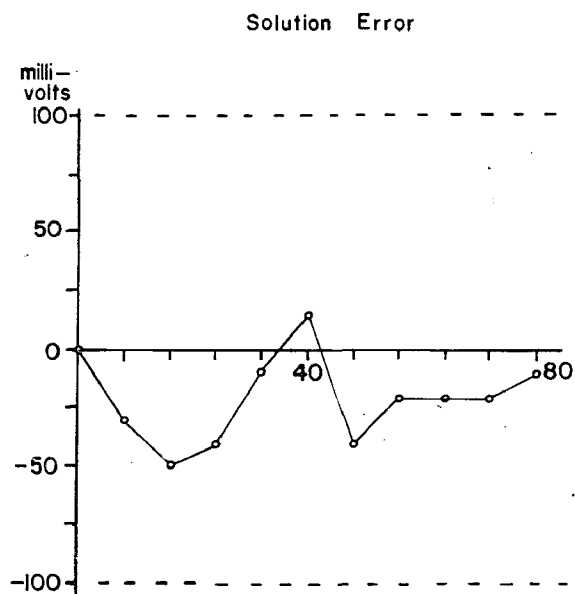
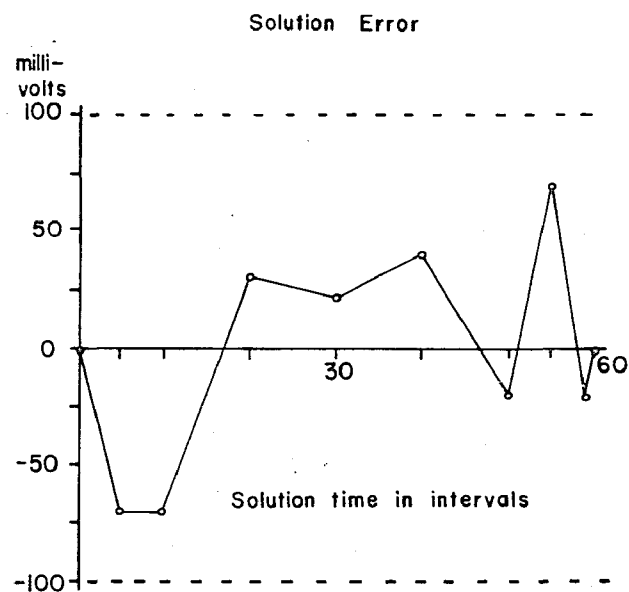
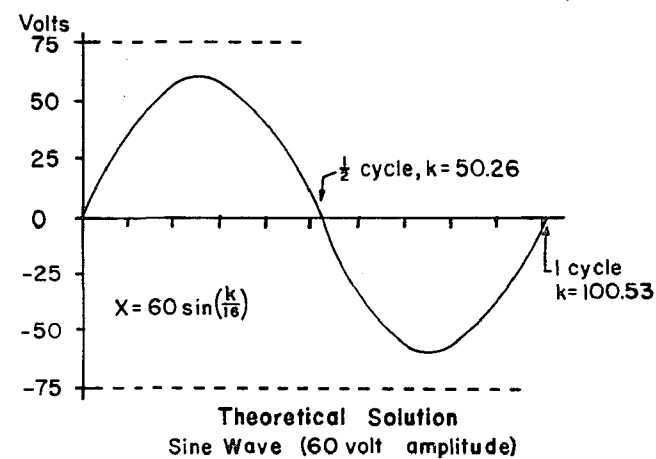
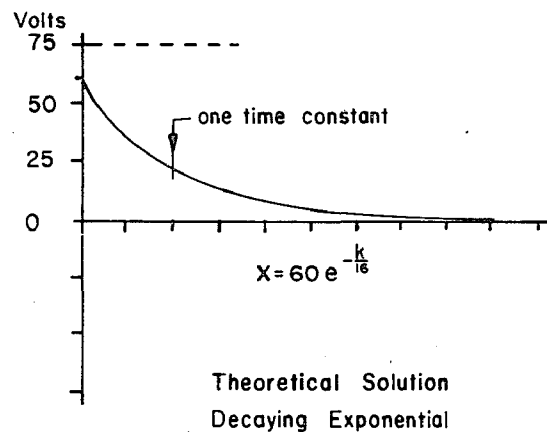
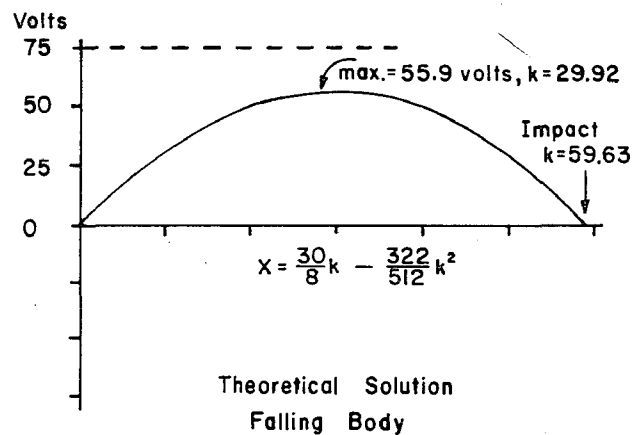
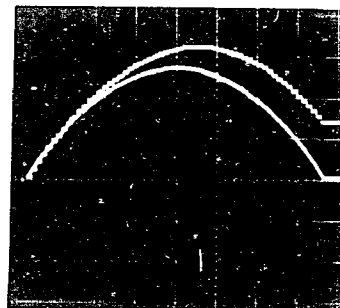
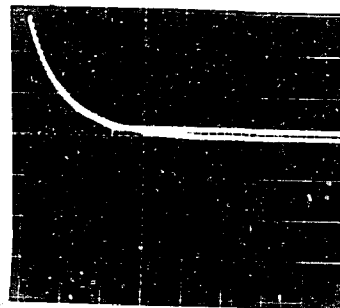


Figure 4.3 Typical Solution Errors

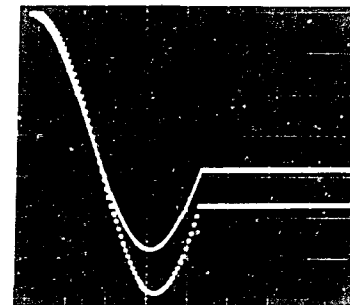
Both  
Removed



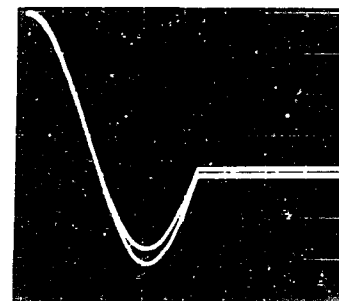
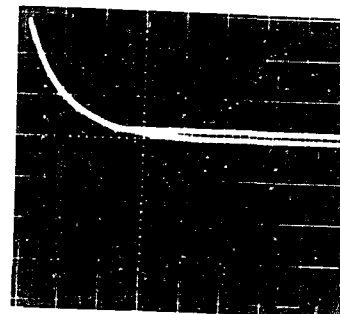
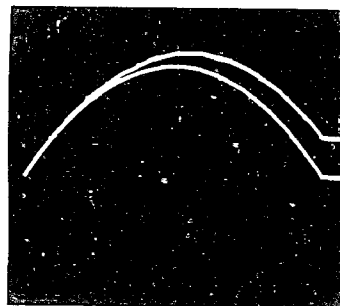
(b)



(c)



Sawtooth,  
No Integrator



Integrator,  
No Sawtooth

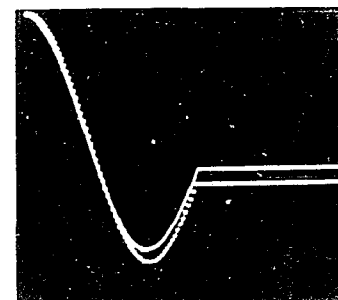
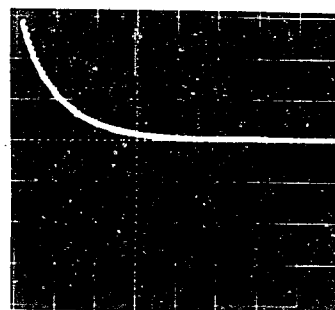
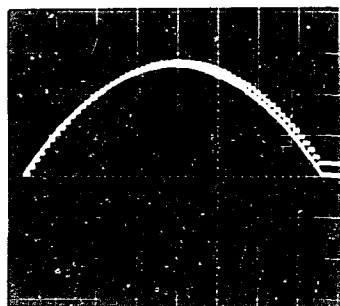


Figure 4.4 Effect of Removing Sawtooth And/or Analog Integrator

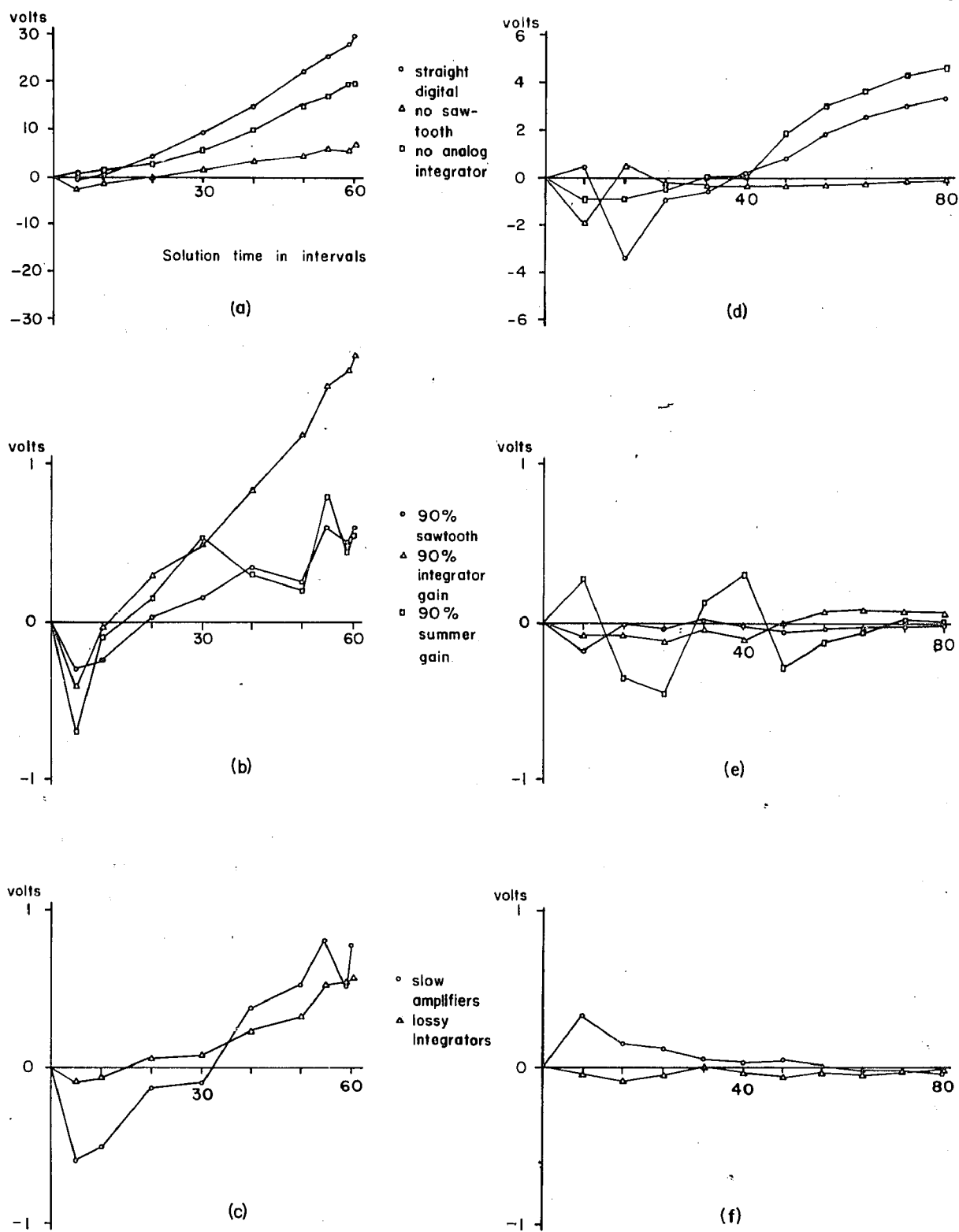
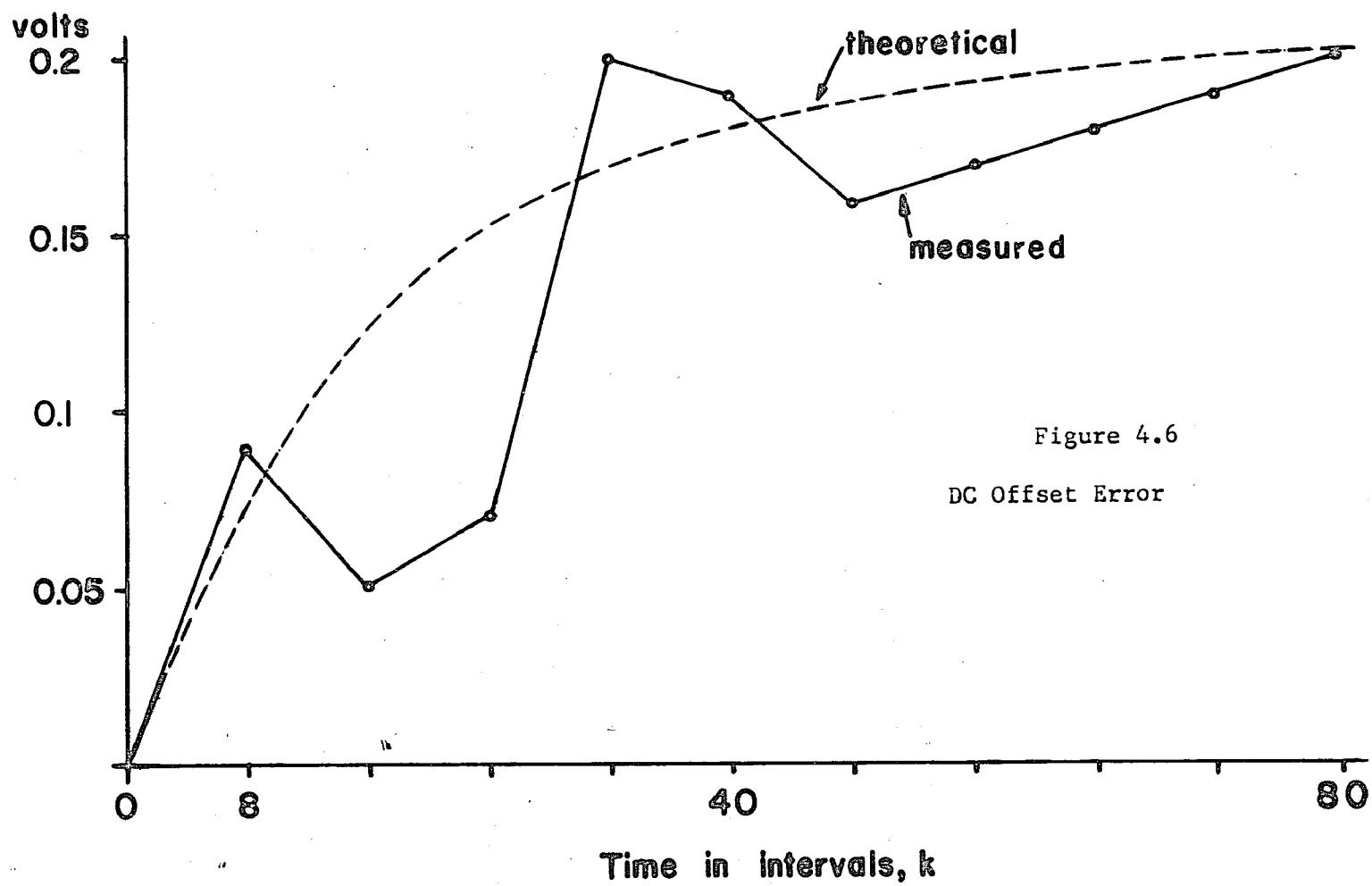
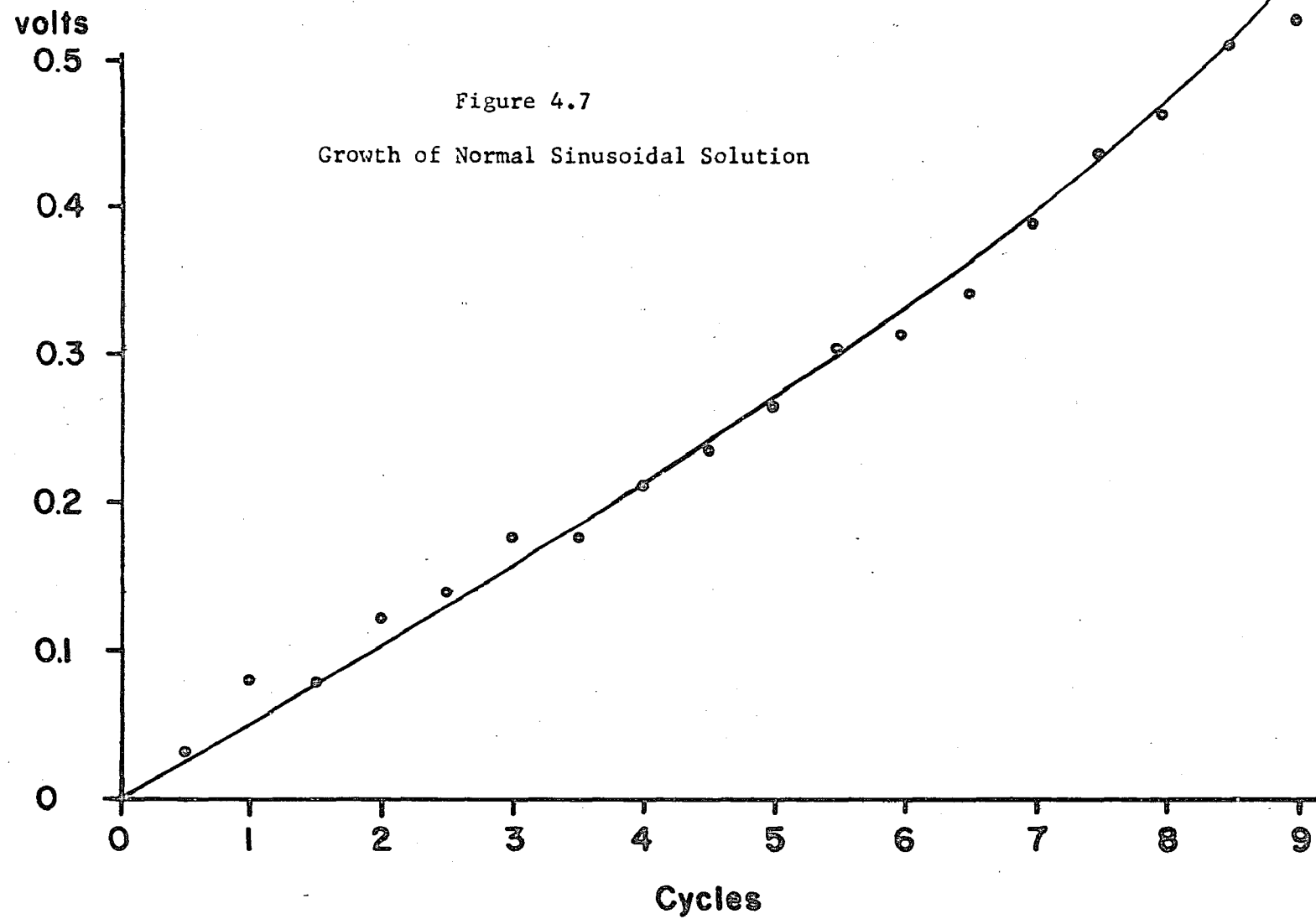


Figure 4.5 Solution Error Due to Artificial Analog Errors

Error



## Solution Growth



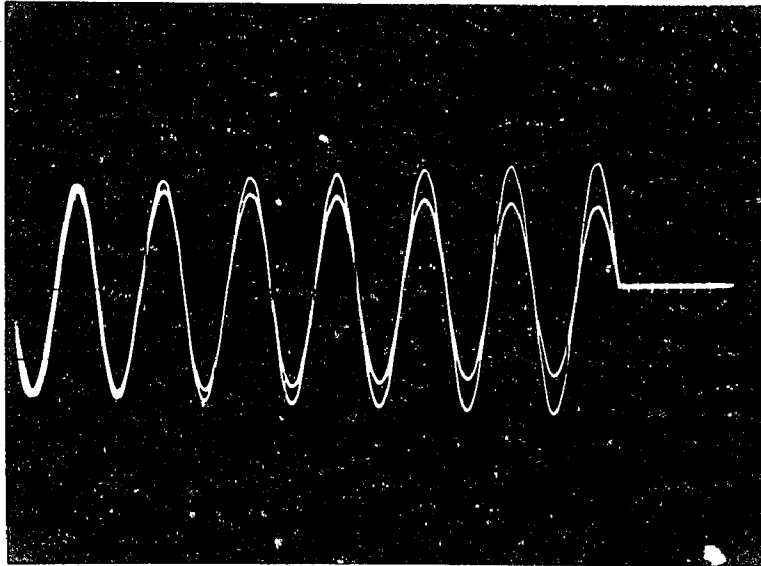


Figure 4.8  
Effect of Summing Amplifier Gain



Figure 4.9  
Variation of  $T_H$

(a)  $T_H = 50 \mu \text{ sec.}$



(b)  $T_H = 250 \mu \text{ sec.}$

## Chapter 5

### COMPARISON OF DIFFERENTIAL ANALYZER SYSTEMS

#### 5.1 INTRODUCTION

The inherent differences among analog, digital and hybrid differential analyzers make them difficult to compare. This chapter presents an attempt at such a comparison in terms of a gross accuracy-bandwidth figure-of-merit based on a per cent of half-scale accuracy.

Before going further, it is important to point out some of the other considerations in a fair comparison of different systems. It should be recognized that analog differential analyzers have limited accuracy, but can provide relatively rapid solutions without truncation errors. Digital and hybrid systems are potentially capable of much higher accuracy, normally at the price of speed and at present, complexity and cost. (Digital components will, however, become less and less expensive as integrated circuits develop.)

Moreover, no generally applicable set of techniques for predicting differential analyzer errors are available. Useful estimates can be made for the class of linear problems, i.e., cases where the computer configuration contains only integrators, summers

and coefficient changing elements, (and machine time is the independent variable of integration). With the realization that the following comparisons cannot be directly extended to the broader class of nonlinear problems, the gross characterization of the various systems does offer some insight into their relative capabilities.

The accuracy-bandwidth figures are based upon the assumption that the total available range of machine variables is utilized, regardless of the type of computer system being discussed. The term "bandwidth," as applied to the speed of computers is related to the maximum slewing rate, or rate-of-change of variables attainable by the computing system.

Consider a system with restrictions:

$$|X(t)| \leq A; \quad |dX(t)/dt| \leq R$$

Let

$$X(t) = A \sin \omega t = A \sin 2\pi f t$$

Then

$$|dX/dt| = |A \omega \cos \omega t| \leq A\omega = 2\pi A f$$

We can thus think of a maximum full-scale frequency of operation

$$f_{\max} = R/2\pi A$$

$$= \frac{|dX/dt|}{2\pi A}$$

This frequency will be the upper limit of the bandwidth of operation for full-scale sinusoids.

## 5.2 ANALOG DIFFERENTIAL ANALYZERS

Assuming dc drift and offset errors can be neglected in high-speed analog computers with chopper-stabilized amplifiers, then the principal types of errors in linear computing elements can be classified as follows:

- a. Static errors due to errors in the dc or static values of computing elements, imprecise settings of initial condition and coefficient potentiometers, noninfinite amplifier gain, and other factors affecting the static gain constants.
- b. Dynamic errors in computing elements; these include any linear frequency sensitive effects in the system. Principal sources in linear computations include:
  - (1) Dissipation in integrator capacitors.
  - (2) Amplifier frequency response, which includes effects of stray capacity in patchboard connections, etc., and which manifests itself principally as a phase shift error in both integrators and summing amplifiers.
  - (3) Frequency-dependent terms in the transfer impedance of other computing elements, viz., resistors and potentiometers, which again produce phase shift and introduce extra poles and zeros into transfer functions of the computer elements.

- c. Timing errors in the READ-OUT and RUN-HOLD systems, which prevent precise measurement of the value of a machine variable. In high-speed computation, the effect of read-out timing errors can be a limitation on useful accuracy.

The effects of the static and dynamic error sources are twofold. First, they cause the characteristic roots of the computer system to differ from those of the system or differential equation being simulated (root perturbations). Secondly, they cause the creation of spurious roots in the computer system, which make the form of the computer solution differ from the desired solution. These two types of error effects are rather similar to the difference-equation-approximation truncation errors arising in DDA's. The effects of timing errors are, of course, equally as important, but in a sense, these errors are not internally generated by the operational computing elements, and cannot be related to the actual capabilities of the computing elements themselves.

Figure 5.1 shows a typical estimate of the accuracy-bandwidth capability of conventional analog equipment (see also Ref. 15 Chapter 4 and Ref. 17). To achieve the higher range of accuracies, precision components must be used in a controlled environment. The curve for repetitive analog systems is also shown, since estimates of ultimate hybrid system capabilities should be made assuming the analog elements are similar to those found in

repetitive systems. The accuracies shown are grossly related to an absolute half-scale accuracy of the computer.

Additional information about the performance of analog differential analyzers can be obtained by observing the attainable accuracy of location of the characteristic roots of a linear system. The sine loop (circle test) used in Chapter 4 has been used previously to study analog systems (Ref. 15). Obviously, the frequency accuracy of a sine loop will essentially depend upon the static accuracy of the computing resistors and capacitors, and will typically be from 0.1 to 1 per cent. The build-up or decay of the solution will depend upon numerous factors, and will vary with frequency. Figure 5.2 shows a typical range of damping characteristics for analog computers.

### 5.3 DDA CAPABILITIES

Considerable work has been done on the analysis of errors in digital differential analyzers.<sup>1</sup> Various assumptions have been made which lead to somewhat different results. One point that is easily overlooked is that parallel-organized DDA's almost invariably use incremental data transfer. Moreover, the integrators must operate on "stale" values of the machine variables. Conventional DDA's usually use an open (extrapolative) trapezoidal

---

<sup>1</sup>For example, see Ref. 8, 9, 11-13, 20-22, and 25. Ref. 13 is especially comprehensive.

integration rule. Assuming this type of machine organization, an estimate of DDA capabilities can be made (in the case of linear computing configurations) which may be used for comparison of the DDA to other types of differential analyzers.

### 5.3.1 DDA's WITH INCREMENTAL TRANSFER

Most modern commercial DDA's use open trapezoidal integration and incremental data transfer. The results of the analysis of truncation and round-off errors presented in Appendix E shows that in precise computations, the round-off errors will predominate in this type of system.<sup>1</sup> Thus the accuracy-bandwidth capability of typical DDA's can be characterized by the expression:

$$\frac{\text{error}}{\text{frequency}} \simeq 200\pi T_I \text{ o/o/cps; } (T_I \text{ small})$$

where  $T_I$  is the iteration time. Equivalently, DDA's of this type have a capability of computing approximately  $1/T_I$  distinguishable increments-per-second.<sup>2</sup>

Figure 5.1 shows the gross accuracy-bandwidth capability for a parallel-organized incremental DDA, using trapezoidal integration. The machine is assumed to have an iteration time of

---

<sup>1</sup>This conclusion does not hold for general-purpose digital computers, where total transfer is used. Truncation errors may predominate when general-purpose machines are used as differential analyzers, depending upon the type of integration rule used (see Section 5.3.2).

<sup>2</sup>This simple result has been pointed out previously by Korn (Ref. 17).

$T_1$  seconds. This figure illustrates how the performance ranges of analog and digital systems overlap, and also indicates roughly the regions where one or the other might be considered distinctly superior. Errors due to improper initial conditions and inaccurate read-out are overlooked in the above discussions of both the analog and digital systems. However, it is felt that the comparison of the gross capabilities of the two types of system is generally valid.

### 5.3.2 TRUNCATION ERRORS IN TOTAL TRANSFER DDA's

It is possible to use a general-purpose digital computer as a total or full-word transfer DDA, although one is usually forced to use serially-organized computation. i. e., one where each integration is performed sequentially, so that the effective iteration time increases in proportion to the number of integrations involved. It would be possible to construct a parallel-organized total transfer DDA, but such a system with a large number of integrators becomes considerably more complicated than an incremental machine. In some cases, solution speed can be increased by using a variable-increment machine which provides a capability for increasing speed and decreasing accuracy as needed.

Nevertheless, it is of interest to compare the capability of a total transfer DDA to the other types of differential analyzers thus far discussed. In such a system, it may be assumed that truncation errors would normally predominate, assuming a sufficient number of digital bits were used for variable representation.

The analysis in Appendix E deals with DDA truncation errors in the case of linear computing configurations. The results apply to closed-loop configurations, without multiple characteristic roots; it assumes no errors due to choice of starting formula. From the analysis, one can predict the truncation errors in terms of the perturbation of the characteristic roots of the computer system. Table 5.1 shows the results for the two cases of rectangular and open trapezoidal integration (see also Ref. 21, 22, and 25). In general, the magnitude of the (fractional) error in root location is  $(ST_I/2)$  for rectangular and  $(5S^2 T_I^2/12)$  for trapezoidal integration.

Figures 5.2 and 5.3 show the effect of truncation errors on the accuracy of location of the roots of a simple undamped sinusoid (harmonic oscillator). The effect will, of course, depend upon the iteration time  $T_I$ ; typical curves are shown for  $T_I = 1.25$  milliseconds, and 10 microseconds, (800, and 100,000 iterations-per-second). The slow rate corresponds to the "iteration" rate of the hybrid system; the fast rate corresponds to a fast DDA, such as TRICE.

It is re-emphasized that in a conventional general-purpose digital computer, the solution of differential equations must be carried out in a serial-organized fashion, so that in a problem involving  $n$  integrations, the iteration time will be at least  $n$  times the machine time required to perform the iteration or updating of one machine variable. For example, the implementation

of the trapezoidal integration rule shown in Table 5.1 requires a minimum of two shift and four add cycles per integration, per iteration.

#### 5.4 HYBRID DIFFERENTIAL ANALYZERS: THEORETICAL PERFORMANCE

The hybrid differential analyzer can be fitted into the above analysis quite easily.<sup>1</sup> Briefly, one can estimate the gross accuracy-bandwidth capability of a hybrid system as follows:

Given the analog computing interval  $T$ , estimate the full-scale bandwidth

$$B_{ANALOG} = 1/4\pi T$$

Pick the point on the estimated accuracy-bandwidth curve for the analog system corresponding to  $B_{ANALOG}$ . From this point draw a line downward along a slope corresponding to a constant accuracy-bandwidth product ( $45^\circ$  slope in Figure 5.1). The length of the line depends upon the number of digital bits. If the system uses  $n$  bits plus sign, the line should correspond to a reduction in bandwidth and an increase in half-scale accuracy of  $2^n$ .

In the actual hybrid system studied, only 1 per cent analog computing elements were used, and  $T$  was set at 1250 microseconds. This corresponds to a full-scale analog bandwidth,  $B_{ANALOG}$  (see Section 1-2-5), of about 64 cps, which is considered to be conservative, being considerably slower than the maximum speed of

---

<sup>1</sup>This approach follows closely that of Korn, Ref. 17.

fast repetitive analog systems of 1 per cent of half-scale accuracy. For a given  $T$ , the rate-scaling limitation precludes computation at higher speeds. In Figure 5.1, point  $A_1$  corresponds to a 1 per cent analog system, with  $T = 1250$  microseconds. Point  $H_1$  then shows the predicted accuracy of a four-bit hybrid system, which should be attainable at all computing frequencies below  $B_{HYBRID} = B_{ANALOG}/2^n$ . In this example,  $B_{HYBRID}$  is about 8 cps, and the estimated error is correspondingly about 1/8 per cent of half-scale (0.125 volts for  $E = 10$ ).

As pointed out in Chapter 1, the analog HOLD intervals,  $T_H$ , reduce the effective real-time hybrid computing speed by a factor  $T/(T + T_H)$ ; in this case, this factor is approximately 0.96, so that the effective real-time value of  $B_{HYBRID}$  is about 7.67 cps.

Point  $H_2$  on Figure 5.1 shows the estimated capability of a hybrid system using 9 bits ( $n = 8$ ) with a 50 microsecond computing interval and a 5 per cent-accurate analog channel. Such a system is technically feasible using modern wide-band transistorized amplifiers, and would provide an estimated accuracy of better than 0.02 per cent of half-scale at a maximum  $B_{HYBRID}$  of about 6.2 cps. Note that an equivalent incremental DDA would have to have to operate at a minimum iteration rate of about 300,000 per second.

Areas  $H_1$  and  $H_2$  on Figures 5.2 and 5.3 show the accuracy of root location (simple undamped sinusoid solution) that can be expected with the two hybrid systems discussed above.

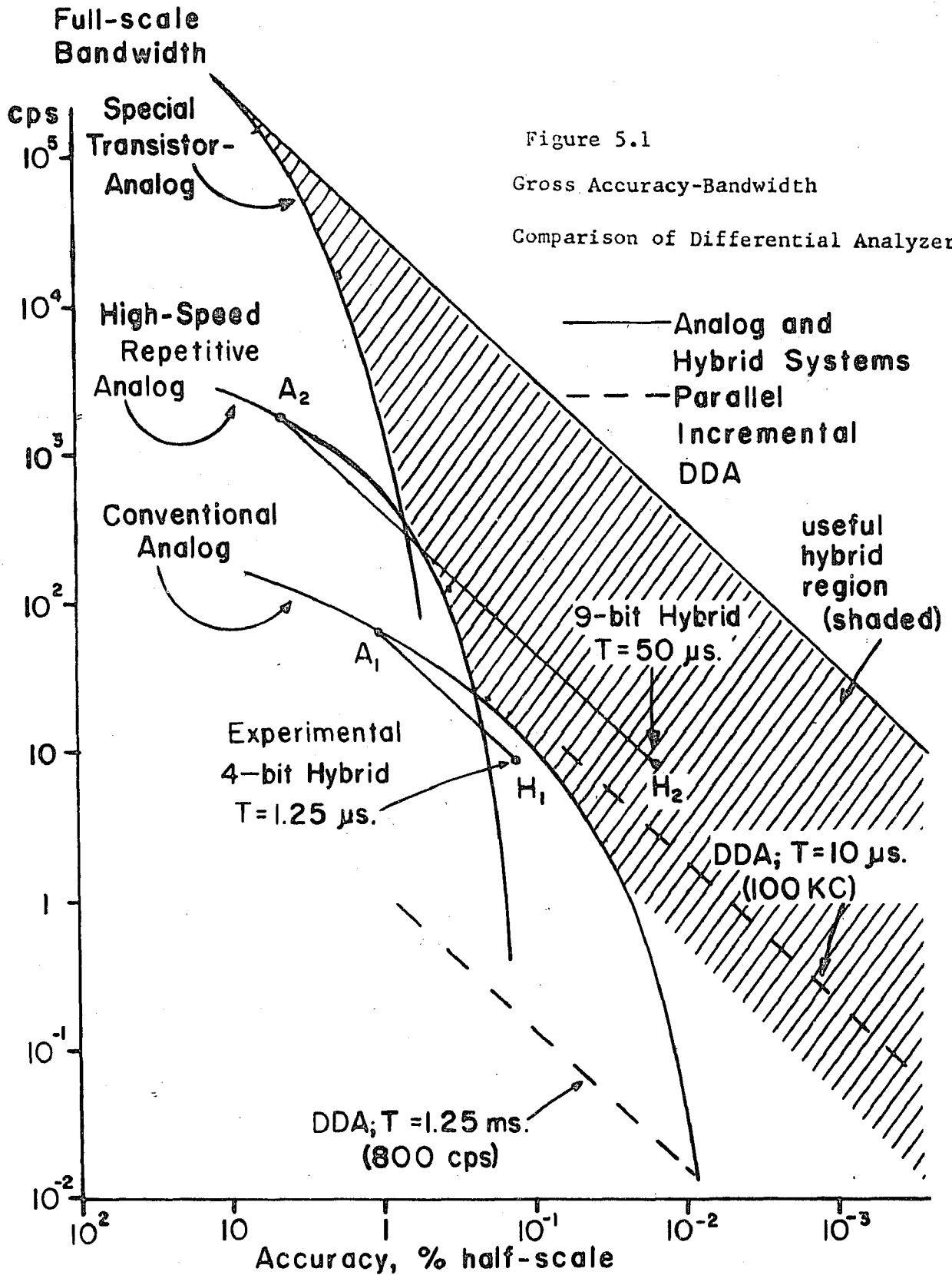
TABLE 5.1

ROOT PERTURBATION DUE TO TRUNCATION ERROR IN  
THE DIGITAL SOLUTION OF DIFFERENTIAL EQUATIONS

<u>Integration Rule</u>	<u>Root Perturbation</u>
Rectangular:	
$X_{n+1} = \dot{X}_n T_I + X_n$	$(1 + ST_I/2)$
Trapezoidal:	
$X_{n+1} = X_n + T_I (3 \dot{X}_n - \dot{X}_{n-1}/2)$	$(1 - 5/12 S^2 T_I^2)$

Note: S is the desired unperturbed root location

$T_I$  is the iteration time; for purposes of comparison to the notation of Chapter 1,  $T_I = T + T_H$ ; also  $T_I = \Delta t$  if  $\alpha_t = 1$ .



# DAMPING COEFFICIENT

$$\frac{\alpha}{\omega} \cdot 10^4$$

CONVERGENT ↑

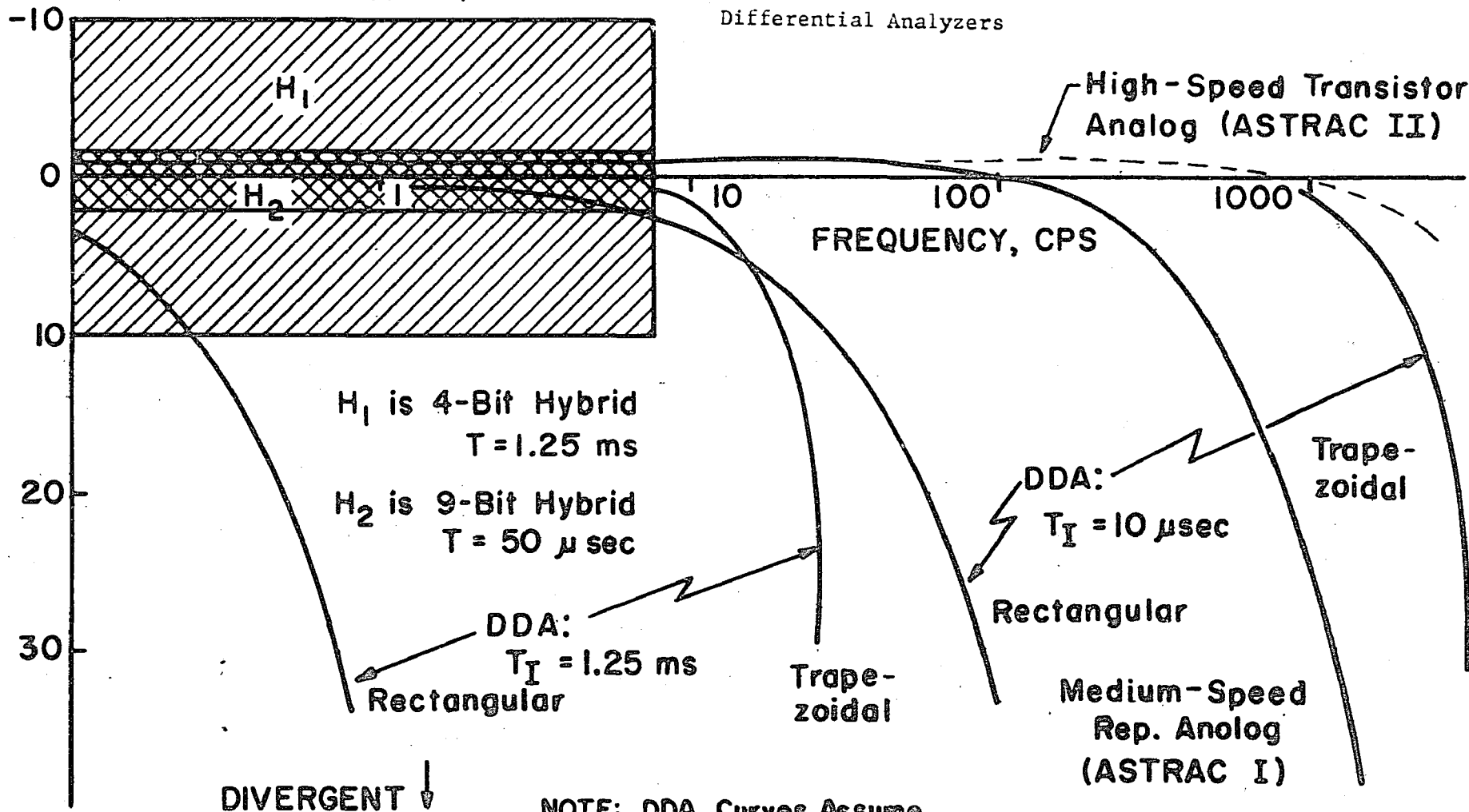
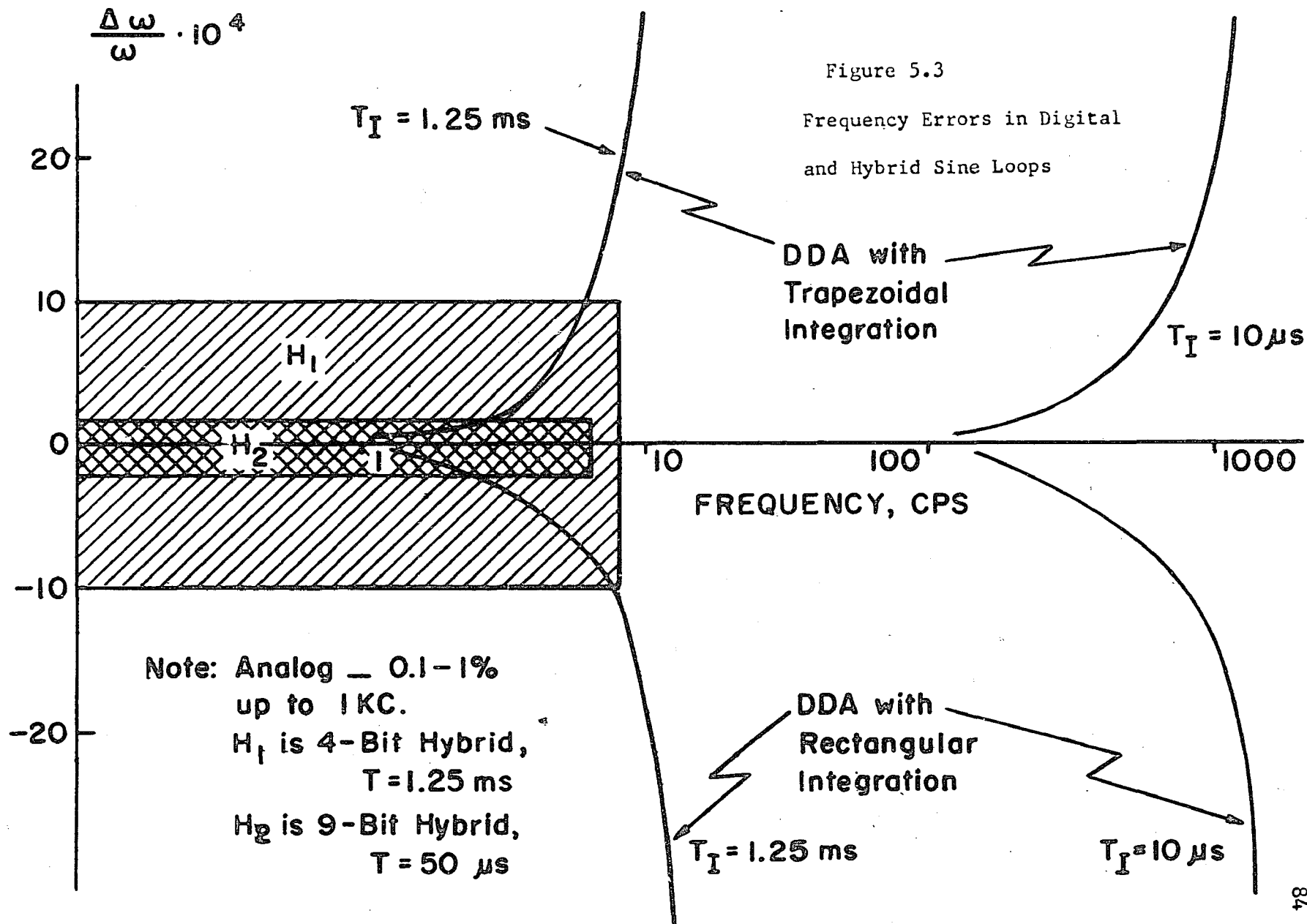


Figure 5.2

Damping Characteristics of  
 Differential Analyzers

NOTE: DDA Curves Assume  
 Total Transfer; Incre-  
 mental DDA'S Are Rate  
 Limited (Fig. 5.1)



## Chapter 6

### CONCLUSIONS: PROJECTED APPLICATIONS AND FUTURE STUDIES

#### 6.1 SUMMARY OF PERFORMANCE OF THE HYBRID SYSTEM

The experimental results verify the theoretically predicted accuracy of the prototype hybrid system, and demonstrate the feasibility of this type of computing technique (at least for linear systems). It is felt that the results justify the concept that the accuracy of the hybrid system can be directly improved by increasing the number of digital bits, as originally predicted. The experimental errors were consistent with the accuracy of the individual system components; no unexplainable deviations from the anticipated performance were observed. The system appears capable of maintaining an accuracy of 0.01 m.u., out of a half-scale range of 7.5 m.u., so long as proper rate-scaling restrictions are observed.

In linear computing configurations, it appears that the location of the system roots can be established with an accuracy of better than 0.1 per cent. The long-term accuracy of the integration process appears to be relatively insensitive to small errors in the analog system components. The digital portion of the system preserves the static gain of the integrators with

surprising precision, and the effect of errors in the analog elements normally appears as a phase shift rather than a gain error. The effects of random variations in the static values of the various computing elements (precision resistors and capacitors) thus would tend to cancel in a large system.

Based on the parameter-influence studies described in Chapter 4, some general conclusions about the different sections of the hybrid integrator are suggested:

- a. It is important that the summing amplifier gain be accurate; also, its bandwidth should be adequate to insure negligible phase shift at the maximum computing speed (e.g., less than one degree phase shift for sine waves of amplitude  $E$  and frequency  $B_{ANALOG}$ ).
- b. It is important that the gain and high-frequency response of the analog integrator also be accurate; however, the DC or low-frequency response is not critical. This suggests that a passive RC network using precision elements could be used in place of the active operational integrator, particularly for systems with a short analog computing interval and modest analog accuracy requirements. Note that this could save a complete d-c amplifier per integrator.

- c. The accuracy of the hybrid integrator is less sensitive to errors in the sawtooth interpolation channel than it is to errors in the summing amplifier and analog integrator. This suggests that a hybrid system utilizing a large number of digital bits (e.g.,  $n > 8$ ) would still operate well with only 5-6 bit D/A converters in the interpolation channel. On the other hand, the D/A converter for the R-register should have sufficient precision to utilize the full accuracy of the analog channel (e.g., at least 7 bits for  $n > 4$ ).
- d. Simple two- or three-transistor comparators could replace the seven-transistor units used in the prototype system. An accuracy of  $\pm 1/2$  v at the maximum analog speed of the system (in this case, 4000 v/sec.) would be adequate.
- e. Since the analog parts of the hybrid machine variables normally have an average value close to zero, long-term overloading of the analog integrators is not a serious problem, except in certain pathological problems. Integrator drift affects the computer solution in a manner quite similar to drift in analog systems but is divided by  $2^n$ .

- f. The use of a HOLD interval,  $T_H$ , successfully eliminates the problem of accommodating the analog transients which accompany the digital operations. The HOLD period may be made an appreciable fraction of the RUN time,  $T$ , without noticeably affecting the solution accuracy. This permits using relatively slow serial digital arithmetic schemes.

## 6.2 FUTURE AREAS OF DEVELOPMENT: PROJECTED CAPABILITIES

This work has been primarily a feasibility study, with experimental verification of the basic operating characteristics of the hybrid integrator. The comparative error analysis presented in Chapter 5 and Appendix E allows one to estimate the effects of changes in the basic system parameters ( $n$ ,  $T$ , and  $p$ ) on speed and accuracy.

On the basis of the experimental results thus far obtained, it appears that the development and testing of additional prototype summing, coefficient-setting, multiplying and function-generating elements would be useful in deriving further information about this type of system.

In particular, the writer feels that a useful hybrid function generator, operating on a principle similar to that discussed in Section 3.5 and Appendix D, can be developed along with auxiliary analog-hybrid and hybrid/analog conversion

equipment to permit the use of such a device with conventional analog computers. Considerable work in this area has already been done, notably by H. Schmid and others (Ref. 1, 4, 28, and 30). Besides having a potentially high accuracy, a hybrid function generator can be designed so as to be digitally programmed by a removable patchboard, punched cards, or tape. With hybrid techniques, it should be possible to construct an accurate resolver component producing  $R \sin \theta$ ,  $R \cos \theta$  with high bandwidth and accuracy (e.g., 0.01 to 0.05 per cent at 10-100 cps) low drift and high repeatability.

Through the use of high-speed transistorized operational amplifiers, the hybrid differential analyzer system described in this work could be given a substantially increased accuracy-bandwidth capability. It would be feasible to operate with an analog computing interval of perhaps 100 microseconds. At this rate, timing errors would become more of a problem. Nevertheless, an accuracy of perhaps five per cent of half-scale could easily be maintained in this type of application. This would permit adding more digital bits, while maintaining reasonable computing speed. For example, with  $n = 8$ ,  $T = 50$  microseconds,  $T_H = 5$  microseconds, and a five per cent analog accuracy, it would be possible to achieve a hybrid computing accuracy of perhaps 0.02 per cent of half-scale, while retaining a real-time computing bandwidth of over five cps. Note that such a system would utilize even cruder analog components than the present system, particularly in the

multiplier and function generator. Moreover, it could be expected to maintain its accuracy without any periodic adjustment or calibration, and without careful environmental control. A parallel-organized incremental DDA using trapezoidal integration would have to operate at an iteration rate of at least 300,000 per second to achieve a similar accuracy and speed. Due to the absence of truncation and round-off errors, a hybrid system should retain this accuracy in computing configurations involving coordinate transformation, division, implicit function generation, and other nonlinear operations. The ensuring of a similar accuracy in DDA solutions involving these operations has proved to be no simple matter (Ref. 6, and 32).

### 6.3 RELATIVE COST

This point will only be touched on lightly, since the relative costs of modern computing systems depend considerably on the quantity produced. Certainly the cost of digital components will decrease with time, so as to narrow the gap between analog and digital system costs. The hybrid differential analyzer appears to lie somewhere between full-analog- and full-digital systems in cost and complexity, probably closer to the DDA at present. A high-speed nine-bit hybrid system would probably have approximately the same commercial cost as a fast parallel DDA such as TRICE or SPEDAC (Ref. 2, and 20). As of this writing, the parts for a hybrid integrator of this type

would cost about \$1,000. Including assembly cost and mark-up, a commercial version of a hybrid integrator might sell for perhaps \$3,000; which is comparable to commercial DDA's (\$4,200 for TRICE). Thus cost would probably not be a major factor in selection of a hybrid system over a full-digital one; however, one would generally not select a hybrid system in cases where an analog system would suffice.

#### 6.4 APPLICATIONS

Obviously, a hybrid differential analyzer is a special-purpose computing system, and one may legitimately ask where useful applications for a hybrid system might arise.

One specific area could be the solution of differential equations associated with the trajectories, orbits, and impact "footprints" of space vehicles. Such calculations usually require nonlinear operations, including division, coordinate transformations, and the use of nonlinear functions to represent effects of gravitational corrections, atmospheric drag, etc. In many cases, a moderately accurate (0.01-0.05 per cent) computing system might be adequate, particularly if it had a computing speed much faster than real-time. Currently, computing installations for this type of work often include DDA's with 100 or more integrators, multipliers, and servos (Ref. 23, and 24), or else a large general-purpose digital computer, or both.

Quite often, these systems must be used with considerable programming skill to obtain fast computation (Ref. 6).<sup>1</sup>

The use of hybrid techniques might permit increased computing speed in such situations; more important, it should be considerably easier to program a hybrid machine to ensure a given degree of accuracy.

#### 6.5 CONCLUSIONS

As a final conclusion, the results of this study confirm that the predicted capabilities of this type of hybrid differential analyzer system can be achieved in practice, and such a system can be used in applications where moderately high-accuracy real-time and faster-than-real-time computing is required. A more conclusive judgment of the practical advantages of this system can be made after the capabilities of hybrid multipliers and function generators have been studied. The outlook is encouraging at this point.

---

<sup>1</sup>Reference 6 cites an example of a lunar orbit calculation requiring the use of both a fast DDA (TRICE) and an IBM 7090 to obtain an answer accurate to 160 km (0.05 per cent of the earth-moon distance), and requiring 5 minutes computing time for one orbit.

## Appendix A

### DESCRIPTION OF DIGITAL SYSTEM, COMPONENTS, CODING

#### A.1 DIGITAL COMPONENTS

The major portions of the digital systems were constructed using Computer Control Corp. 1 MC S-PAC modules. These use 0 and -6 v logic levels (-6 for logical "1"); signal rise and fall times are less than 0.1 microsecond. Most of the logical operations in the actual computing system were implemented using NAND gates and asynchronous AC pedestal-gated flip-flops.

#### A.2 DIGITAL VARIABLE REPRESENTATION - CODING

The proper choice of a digital code must, of course, depend upon careful consideration of all operations to be performed. In a typical differential analyzer, at least the following operations involving digital variables are required:

- a. Digital integration of one variable (summation and storage).
- b. D/A and A/D conversion.
- c. Digital summation of several variables (without memory).
- d. Digital multiplication.
- e. Digital function generation.

It is assumed that both positive and negative numbers must be accommodated. In the present case, the digital variable will range from -7 to +7, and thus can be represented by 4 binary digits. Several codes could be used including:

- a. Simple signed magnitude.
- b. Complementary representation of negative numbers (one's or two's complement).
- c. Cyclic (Gray code).

Other codes could, of course, be used but no others seem to have any particular advantages.

The signed magnitude code, is perhaps easiest to use in driving read-out display equipment and also the simplest to work with for multiplication and function generation. It is also not the best code for easy D/A conversion. The complemented codes are the best for summation and for digital integration: e.g., using the two's complement code and incremental data transfer, a digital integrator becomes basically an up-down counter. Using one's complement code, due to the dual representation of "zero", a slight amount of extra logic is required to handle + and - carries properly.

Since integration, summation, and D/A conversion are anticipated to be the most common operations in a hybrid differential analyzer, the two's complement code was chosen for the standard digital number representation in the system (see Table A.1).

### A.3 DIGITAL DATA TRANSMISSION

In principle, it is possible to use either parallel or incremental (serial) transmission of the digital variables in the system. Scaling considerations will ensure that the change in a digital variable will never exceed one machine unit at the end of a particular computing interval. This makes incremental transmission attractive, since only two signals are required to transmit digital information between operational elements,  $\Delta X$  and  $S(\Delta X)$ .<sup>1</sup> Reversible counters (registers) are used for generation of each full digital variable  $X_D$  from its increments,  $\Delta X_D$ .

It should be pointed out that a full-word representation of each digital variable always exists in some register, even if incremental transfer is used. From an organizational viewpoint, the basic difference between an incremental and a parallel transmission system is in the assignment of the interface between various computing elements. For example, in the discussion of the hybrid integrator in Chapter 2, it is shown that an input register is required to store the digital portion of the integrand,  $\dot{X}_D$ . In an incremental system, this register is physically located in the integrator itself, but in a parallel system, the  $\dot{X}_D$  register would be located in the preceding computing element.

---

<sup>1</sup>In this case, if incremental transfer is used, it must be of the ternary form (-1, 0, or + 1), rather than the binary form (-1 or + 1), which is sometimes used in DDA systems.

In the present system, incremental transfer is used with the following code:

$\Delta X$	= Existence of an Increment:	Negative (assertion) pulse
$S(\Delta X)$	= Sign of Increment:	"1" (-6 v) for negative
		"0" (0 v) for positive

#### A.4 THE DIGITAL INTEGRATOR SYSTEM (Incremental Transfer)

The digital integrator system shown in Figure 2.3 uses standard Computer Control Corporation 1 MC S-PAC modules to perform the following operations:

1. Accumulation of  $\Delta X_D$  increments in  $X_D$  register.
2. Digital addition of  $X_D/16$  to  $R_D$ .
3. Digital addition of carry ( $\pm 1$ ) to  $R_D$ .
4. Transmission of output increments  $\pm \Delta X$  to next unit.

The system uses a two's complement to represent negative numbers, both in the  $X_D$  and in R-registers. In a two's complement system, no end-around carries or borrowing are required; this greatly simplifies the timing and logic requirements of the system. For  $n = 3$ , the system requires 26 gates and 12 flip-flops (48 transistors total).

The sequence of operations is controlled by a series of seven timing pulses from the integrator subroutine clock and the RH (Run/Hold) signal (Figure 25); the sequence is as follows:

- a. The positive-going transition of the RH signal at the

beginning of the analog holding interval causes the carry storage flip-flops (A and B) to be set in accordance with the states of the comparators at that time: that is:

1. Flip-flop A is in the "1" or SET state if  $X_A > 5$  volts.
2. Flip-flop B is in the "1" or SET state if  $X_A < -5$  volts.

The auxiliary digital logic then produces the three carry-implementing signals C, S( $\Delta X$ ), and S(C) as follows:

1.  $C = (A + B)$ .
2.  $S(C) = A$ .
3.  $S(\Delta X) = \bar{A}$ .

The carry signal, C, is a "1" whenever there is a carry of either polarity. S(C) and S( $\Delta X$ ) are logical complements. S( $\Delta X$ ) is a logical "1" when a negative carry is to be transmitted to the next computing element; correspondingly, S(C) is a "0" to denote the concurrent addition of + 1 m.u. to the R-register.

- b. Clock pulses  $T_1$  to  $T_4$  cause the addition of  $X_D$  to the four lower positions of the R-register. This is accomplished by sequentially gating each bit of  $X_D$  into a parallel information drop-in input in the corresponding R-register flip-flop, and allowing carries to ripple through the register before the next

bit is transferred. By using a 200 KC pulse rate with 1 MC logic, more than adequate time for carry propagation is assured. Note that if  $\dot{X}_D$  has a "1" in the sign-bit position, there still remains the need for adding a "1" to the upper two orders of the R-register; this operation is done simultaneously with the addition of the carry.

- c. Information is added to the upper two orders of the R-register by the action of pulse  $T_5$  and  $T_6$ . The gates controlling this operation receive information about the sign of  $\dot{X}_D$  and the sign and existence of a carry, and adjust the R-register by the following rules:
1. Add 11 if  $\dot{X}_D$  is negative ( $\dot{X}_D$  sign bit is "1") and no internal carry OR if  $\dot{X}_D$  is positive and there is a negative internal carry ( $C = 1$ ,  $S(C) = 1$ ).
  2. Add 01 if  $\dot{X}_D$  is positive, and the internal carry is + 1 m.u.
  3. Add 00 if  $\dot{X}_D$  is positive and no carry OR if  $\dot{X}_D$  is negative and there is an internal carry of + 1 m.u. (00 is 0 or 4, modulo 2).
  4. Add 10 if  $\dot{X}_D$  is negative and there is an internal carry of -1 m.u. (10 is 6, modulo 2).

Figure 2.3 shows the required logic elements to perform the necessary operations in accordance with these rules. The lines  $P_0$  and  $P_S$  must implement:

$$P_0 = T_5 (\dot{\bar{X}}_S C + \dot{X}_S \bar{C})$$

$$P_S = T_6 (A\bar{B} + \bar{C}\dot{X}_S)$$

- d.  $T_7$  transmits a  $\Delta X$  pulse to the next computing element; the carry sign  $S(\Delta X)$  is transmitted as a dc level.
- e. Digital input increments are added or subtracted in  $\dot{X}_D$  whenever they are received. This is accomplished by making the  $\dot{X}_D$  register an up-down counter. The proper carry sign signal must occur at least 1 microsecond before the  $\Delta X$  pulse. This signal controls the gates on the  $\dot{X}_D$  register, making it either an up- or a down-counter; the  $\Delta X$  pulse is then correctly accumulated and stored for the next computer run. If the input increment comes from another integrator, it will occur at time  $T_7$ . It may also arrive later, if it comes from some other type of computing element. Input increments may actually be received from more than one source, if staggered timing is used.

#### A.5 CHANGING SCALE FACTORS AND NUMBER OF BITS

The system discussed above has a digital scale factor of 1/16th, i.e.,  $R = 1/16 \dot{X}_D - (\text{carries})$ . The scale factor

cannot be made larger, but it can be made smaller by any desired power of two simply by adding more bits to the R-register, and introducing  $X_D$  into the lower-order digits of R. The sign digit of  $X_D$  should be added into all higher-order locations of the R-register, the upper two digits of R being treated as in the above system, and any intermediate digits sequentially receiving a "1" if the sign of  $X_D$  is negative. For example, to cut the digital gain to  $1/64$ , two more digits are required in the R-register, and two more clock pulses are required.

A hybrid system with additional digital bits could be implemented in the same manner. Beyond about 8-10 bits, it might be desirable to consider using some form of true parallel scheme to conserve time. The equivalent amount of logic equipment includes  $4n + 14$  gates,  $n + 3$  dual input flip-flops, and  $n + 3$  single input flip-flops ( $8n + 22$  transistors and  $20n + 50$  diodes). For example, a nine-bit ( $n = 8$ ) system would require about 86 transistors and 210 diodes.

TABLE A.1

FOUR-BIT TWO'S COMPLEMENT CODE ( $n = 3$ )

Binary Number	Variable Value
0111	+7
0110	+6
0101	+5
0100	+4
0011	+3
0010	+2
0001	+1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	Unused

## Appendix B

### DESCRIPTION OF ANALOG SYSTEM COMPONENTS: COMPARATORS, DIGITAL-ANALOG CONVERTERS.

Figures 2.4 and B.1 show the analog portion of the hybrid integrator; it includes two comparators and two D/A converters of the type discussed in Ref. 34 and 36. An all-transistor version would require about  $(2n + 30)$  transistors and 6 diodes. The symmetrical switch drivers are discussed in Ref. 35. The purpose of this Appendix is primarily to describe the basic operations performed in this section of the hybrid integrator in sufficient detail to permit a similar system to be designed using different available components; where desired.

The effective gain of the integrator is 50, i.e.,

$$X = 50 \int_0^t \dot{X} dt + X_0$$

This gain constant is consistent with the gain of 1/16 in the digital section, and the analog computing interval of 1250 microseconds. For a different gain, one should use the relationship

$$a = \frac{2^{-(n+1)}}{T}$$

This is the gain of a hybrid integrator using the highest digital scale factor for a given  $n$ . Note that for a given  $T$ , only discrete values of  $a$  are available. Other values would have to be obtained

with a coefficient-multiplying component.  $T$  was chosen to be 1250 microseconds in the experimental system in order to yield a gain of 50, which agrees with the usual choice of operational gains as some decimal multiple of 1, 2, or 5. If, for example,  $T$  is made 1 millisecond, then the integrator gain would have been  $1000/16$ , a rather odd value for a computing element, although it could be used in a general computing system which contained coefficient-changing components.

The gain through each signal path to the summing amplifier must be established to give this same gain constant. In particular,

- a. The total gain from the  $\dot{X}_A$  input, through the analog integrator, to the output of the summing amplifier should be  $a$ ; to prevent overload of the integrator itself, the gain from  $\dot{X}_A$  to the integrator output may have to be made less than  $a$ ; hence the factor  $K$  in Figure 2.4. In the present system,  $K = 1$  was found to be satisfactory.
- b. The signal component due to the linear interpolation term  $a\dot{X}_D \tau$  is produced by a D/A converter driven by  $\pm 10$  v interpolating ramps,  $+V_I$  and  $-V_I$ . The gain through the  $\dot{X}$  converter to the summing amplifier output must yield a component:

$$+V_I (\dot{X}_D) \cdot 1/2^{n+2} \text{ volts; } V_I \text{ in volts, } \dot{X}_D \text{ in m.u.}$$

- c. Similarly, the gain through the "R" converter must yield a component

$$10 R_D \cdot 1/2^{n+2} \text{ volts}$$

where  $R_D$  is the value of the R-register in m.u. Note that the most significant digit of the R-register is a sign digit, the next, a units digit, the rest are fractional digits. (This statement still holds true for systems with more digital bits.)

#### D/A CONVERTERS (See Ref. 28)

Figure B.1 shows the two D/A converter networks, with an appropriate set of resistance values to achieve the required gains for a four-bit system. The complementary logic levels are directly obtained from the complementary outputs of the  $X_D$ - and R-register flip flops, where required.

#### COMPARATORS (See Ref. 27)

The comparators used to sense analog overflows are identical units, designed to have a threshold level of zero volts. The external voltage dividers provide accurate level translation of the output of the summing amplifier, so that comparator A senses when  $X_A$  is greater than +5 volts, Comparator B senses when  $X_A$  is less than -5 volts. The gate level controls on the flip-flops FFA and FFB are controlled by the comparator output signals, so that, upon receipt of a positive-going transition in the RH signal, they are set to

store the state of the comparators at the end of a computing interval. There is actually a timing error of about 1.5 microseconds in this interrogating process, which means that the overall error in the over-flow sensing operation might be as great as 6 mv, due to the timing error. However, the comparator itself is only accurate to about  $\pm 10$  mv, so that the overall error is not appreciably affected. Note that the comparator accuracy does not actually have to be high, since the rate scaling limitations prevent the analog channel from going more than 1 m.u. in two computing intervals. Normally the analog channels can exceed 1 m.u. by 0.05 to 0.1 m.u. without overloading, so that this much error in the comparator systems can be tolerated. Thus, the comparator circuit used in this experiment could be replaced by a simple 2-3 transistor Schmitt trigger circuit.

#### LINEAR COMPUTING ELEMENTS

A  $\pm 10$  volt analog signal range was selected, so that the majority of analog operations could be easily implemented with modern fast transistor operational amplifiers. The speed of the system was basically determined by an estimate of the maximum rate-of-change of the analog variables which could successfully be accommodated by the analog computing components and an estimate of the timing accuracy of the clock and read-out systems.

The use of conventional  $\pm 100$  v operational amplifiers is less desirable for many reasons. The analog switching circuits required would be somewhat more complex, require much more power

consumption and, worst of all, would be slower than the corresponding solid-state circuitry suitable for use with transistorized analog systems. Moreover, since all-solid-state digital-logic modules are used throughout, the problem of providing level-matching circuitry between the analog and digital sections is greatly reduced by using low-voltage circuitry in both systems.

It was estimated that the analog systems should be able to maintain 1 per cent of half-scale component accuracy if the variables have a maximum rate-of-change of 4000 v/sec. This corresponds to a full-scale sine wave frequency of about 63.3 cps, and was felt to be a conservative limit on computing speed using modern analog computing elements.

The actual prototype system was constructed before suitable  $\pm 10$  volt transistor operational amplifiers were locally available. The operational amplifiers actually used were a standard Philbrick K2-XA/K2-P vacuum tube amplifier. This substitute was found to simulate a transistor operational amplifier adequately, and worked nicely with low-impedance computing networks.

The computing networks were selected to be suitable for the speed and current capabilities of the amplifiers. Summing and feedback resistors can be in the range of 10-100K-ohms. Where required, 5,000 ohm coefficient potentiometers would be suitable, and integrating capacitors can range from 0.01 to 1 mfd.

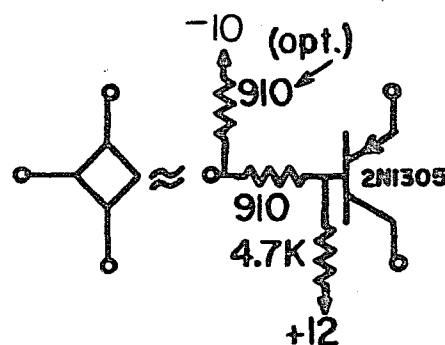
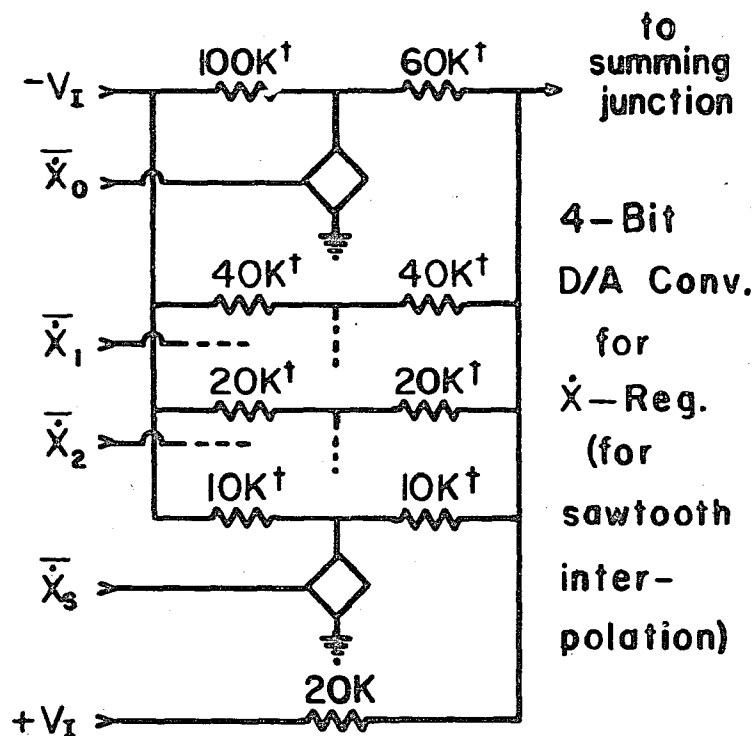
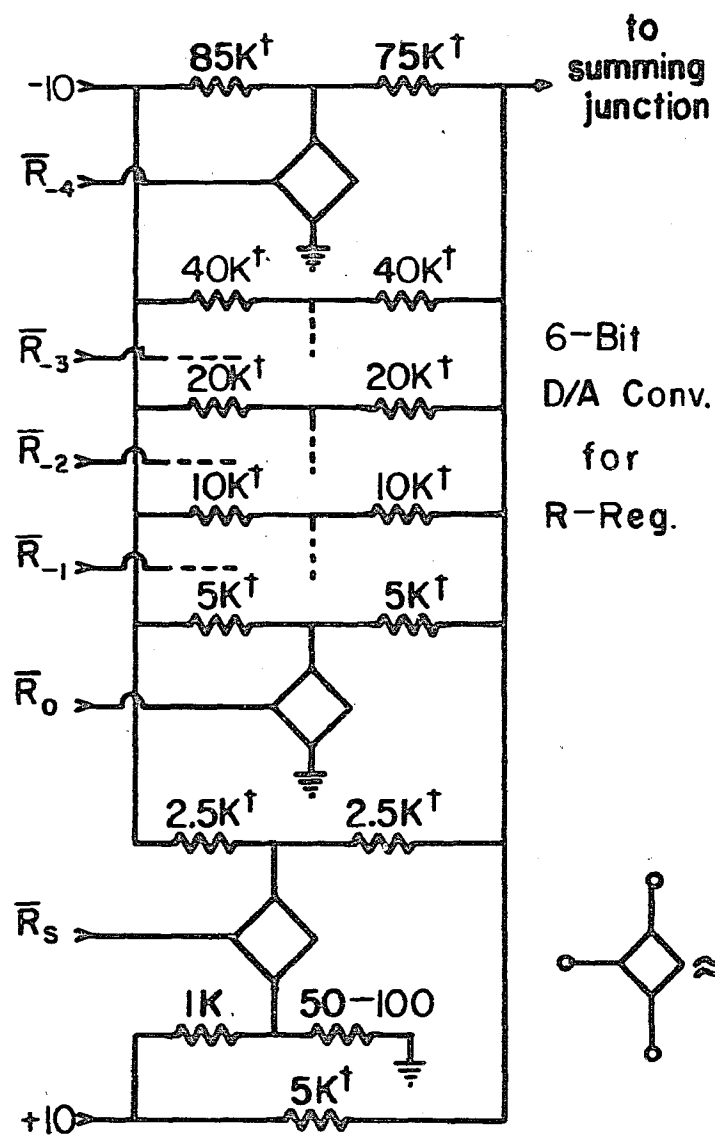


Figure B.1  
Digital-To-Analog Converters

## Appendix C

### DERIVATION OF OPERATIONS REQUIRED IN A HYBRID SUMMING COMPONENT

The hybrid summing component shown in Figure 3.1 performs the operation

$$Z_D + Z_A = \frac{1}{2} \left[ (X_D + X_A) + (Y_D + Y_A) \right]$$

The digital inputs and outputs are the incremental quantities  $\Delta X_D$ ,  $\Delta Y_D$ , and  $\Delta Z_D$ . To express the required operations mathematically,

Let  $X_{D0}$ ,  $Y_{D0}$ , and  $Z_{D0}$  be the initial values of the digital machine variables; note that  $Z_{D0}$  actually appears in the input register of the next succeeding computing element. During a computer run, we want

$$\begin{aligned} Z &= Z_D + Z_A = Z_{D0} + \Sigma \Delta Z_D + Z_A \\ &= \frac{1}{2} \left[ X_{D0} + Y_{D0} \right] + \frac{1}{2} \left[ \Sigma \Delta X_D + \Sigma \Delta Y_D \right] + \frac{1}{2} (X_A + Y_A) \end{aligned}$$

or

$$\begin{aligned} &Z_{D0} + \Sigma \Delta Z_D + Z_A \\ &= Z_{D0} + \left[ \frac{1}{2} (X_{D0} + Y_{D0}) - Z_{D0} \right] + \frac{1}{2} (\Sigma \Delta X_D + \Sigma \Delta Y_D) + \frac{1}{2} (X_A + Y_A) \end{aligned}$$

Let  $\Sigma \Delta Z_D = N =$  algebraic sum of all output carries

Thus  $Z_D = Z_{D0} + N$

Then

$$Z_D + Z_A = Z_{D0} + N + Z_A = Z_{D0} + N + 1/2 (X_A + Y_A) \\ + \left[ 1/2 (X_{D0} + Y_{D0}) - Z_{D0} - N + 1/2 (\Sigma \Delta X_D + \Sigma \Delta Y_D) \right]$$

Thus

$$Z_A = 1/2 (X_A + Y_A) + \left[ 1/2 (X_{D0} + Y_{D0}) - Z_{D0} - N + 1/2 (\Sigma \Delta X_D + \Sigma \Delta Y_D) \right]$$

Also

$$Z_A = 1/2 (X_A + Y_A) + R$$

or

$$R = (1/2 (X_{D0} + Y_{D0}) - Z_{D0}) - N + 1/2 (\Sigma \Delta X_D + \Sigma \Delta Y_D)$$

From this it can be seen that the R-register must be a three-bit register, which receives the following information:

- a. The initial value  $1/2 (X_{D0} + Y_{D0} - Z_{D0})$ , note that this number is either  $-1/2$ ,  $0$ , or  $+1/2$ .
- b.  $N = \Sigma \Delta Z_D$ , the algebraic sum of all output carries; this sum is subtracted from the R-register, one m.u. at a time, as the computer run progresses.
- c.  $1/2 (\Sigma \Delta X_D + \Sigma \Delta Y_D)$ , the algebraic sum of all input increments; this number is accumulated in the R-register to keep account of the running sum of the digital input variables.

Note that the value of the R-register can take on only the value  $-1 \frac{1}{2}$ ,  $-1$ ,  $-1/2$ ,  $0$ ,  $1/2$ , or  $1 \frac{1}{2}$ ; this is true regardless of the number of digital bits in a machine variable. Thus the same summing component may be used in a system with a different number of digital bits. Note also that the scale factor of the summing component can be divided by an integral power of two, without any significant change in the design, except the addition of more low-order digits in the R-register, and appropriate changes in the analog channel gain.

Three input variables can be summed in a similar fashion; of course, the summer gain must be changed accordingly. If  $K$  variables are to be summed in this manner, the summer gain constant must be  $2^{-k}$ , where  $k$  is an integer such that  $2^k \geq K$ .

## Appendix D

### MULTIPLYING AND FUNCTION GENERATING WITH HYBRID COMPONENTS: ERRORS IN HYBRID FUNCTION GENERATORS

#### D.1 HYBRID MULTIPLIERS

A hybrid multiplier performs the operation

$$\begin{aligned} Z &= Z_D + Z_A = \frac{1}{2^n} (X_D + X_A) (Y_D + Y_A) \\ &= \frac{1}{2^n} \left[ X_D Y_D + X_D Y_A + X_A Y_D + X_A Y_A \right] \end{aligned}$$

Where

$$Z_D = Z_{DO} + \Sigma \Delta Z_D$$

$$Y_D = Y_{DO} + \Sigma \Delta Y_D$$

$$X_D = X_{DO} + \Sigma \Delta X_D$$

Note that  $\Delta(X_D Y_D) = Y_D \Delta X_D + X_D \Delta Y_D + \Delta X \Delta Y$

so that  $X_D Y_D = X_{DO} Y_{DO} + \Sigma Y_D \Delta X_D + \Sigma X_D \Delta Y_D - \Sigma \Delta X \Delta Y$

Letting

$$N = \Sigma \Delta Z_D$$

$$\begin{aligned} Z &= Z_{DO} + \Sigma \Delta Z_D + Z_A = \frac{X_{DO} Y_{DO}}{2^n} + \frac{\Sigma Y_D \Delta X_D}{2^n} + \frac{\Sigma X_D \Delta Y_D}{2^n} - \frac{\Sigma \Delta X \Delta Y}{2^n} \\ &\quad - Z_{DO} - N + \frac{X_D Y_A + X_A Y_D}{2^n} + \frac{X_A Y_A}{2^n} + Z_{DO} + N \end{aligned}$$

Regrouping, we have

$$Z_{DO} + \Sigma \Delta Z_D + Z_A = Z_{DO} + N + \left[ \frac{X_{DO} Y_{DO} + \Sigma Y_D \Delta X_D + \Sigma X_D \Delta Y_D - \Sigma \Delta X \Delta Y}{2^n} - Z_{DO} - N + \frac{X_D Y_A + X_A Y_D + X_A Y_A}{2^n} \right]$$

Thus

$$Z_A = R + \frac{X_D Y_A + X_A Y_D + X_A Y_A}{2^n}$$

or the R-register must represent

$$\frac{X_{DO} Y_{DO} + \Sigma Y_D \Delta X_D + \Sigma X_D \Delta Y_D - \Sigma \Delta X \Delta Y}{2^n} - Z_{DO} - N$$

Again, as in the case of the hybrid summing component (Appendix C), the R-register performs three principal functions:

- a. Storage of the initial fraction

$$\frac{X_{DO} Y_{DO}}{2^n} - Z_{DO}$$

- b. Accumulation of the incremental values of the product of the input digital variables

$$1/2^n Y_D \Delta X_D + X_D \Delta Y_D - \Delta X \Delta Y$$

- c. Accumulation of  $-\Sigma \Delta Z_D$ , i.e., subtraction of the algebraic sum of all output increments.

In this case, the R-register is an  $(n + 2)$  - digit register, with magnitude strictly less than 2 m.u.

In the case where only coefficient changing is required, i.e.,  $Y = C = C_D + C_A$ , the organization of the system is the same, except that now, since  $\Delta Y_D = \Delta C_D = 0$ , the R-register is simpler, since the incremental inputs are:

$$1/2^n \left[ C_D \Delta X_D \right]$$

and thus

$$R = \left[ \frac{X_{D0} C_{D0}}{2^n} - Z_{D0} \right] + 1/2^n C_D \Sigma X_D - N$$

## D.2 HYBRID FUNCTION GENERATORS

A hybrid function generator forms the function

$$Z_D + Z_A = F(X_D + X_A)$$

$$= F(X_D) + \left. \frac{\partial F}{\partial X} \right|_{X_D} X_A + 1/2 \left. \frac{\partial^2 F}{\partial X^2} \right|_{X_D} X_A^2$$

$$+ \dots + 1/k! F^k(X_D) X_A^k + R_k(X_D, X_A); (F^k = \frac{\partial^k}{\partial X^k})$$

i.e., a Taylor-series approximation to the desired function<sup>1</sup>.

---

<sup>1</sup>This development assumes  $F(X)$  is an analytic function of the real variable  $X$ ;  $|X| < 2^n$ . The approach could be extended to include a discrete number of points of discontinuity in this range of  $X$ .

Let

$$Z_D = Z_{D0} + \Sigma \Delta Z_D = Z_{D0} + N$$

Then

$$Z_{D0} + N + Z_A = Z_{D0} + N + \left[ F_D(X_D) - Z_{D0} - N + F'(X_D) X_A + 1/2 F''(X_D) X_A^2 + \dots \right]$$

Thus, to implement hybrid function generation with incremental digital transfer we let

$$Z_D = Z_{D0} + \Sigma \Delta Z_D = Z_{D0} + N$$

Then

$$Z_{D0} + \Sigma \Delta Z_D + Z_A = Z_{D0} + N + F(X_D) - Z_{D0} - N + F'(X_D) X_A + 1/2 F''(X_D) X_A^2 + \dots$$

and

$$Z_A = F(X_D) - N - Z_{D0} + F' X_A + 1/2 F'' X_A^2 + \dots$$

As in the case of the other hybrid computing elements, there are scaling restrictions on the function generator, viz.:

$$|F| \leq 2^n$$

$$|dx/dt|, |dF/dt| < 1/2T$$

Since

$$dF/dt = (\partial F / \partial x) (dx/dt)$$

Then  $\partial F / \partial x = F' \leq 1$  insures the satisfaction of the output rate-scaling limitation.

Note also that each digital function  $F^k(x_D)$  must be generated with an accuracy consistent with the accuracy of the analog system.

Let  $i_k$  be the number of bits required in the generation of  $F^k(x_D)$ ;

let  $p$  be the accuracy of the analog system in per cent of one m.u.

Then  $i_k$  must be greater than the smaller of the two quantities:

$$\ln_2 (100/p) \quad \text{or} \quad \ln_2 (100 F^k \max / p)$$

Similarly, an analysis of the function must be made to determine how many terms are required to achieve an accuracy of approximation consistent with the rest of the computing system.

Let  $k$  be the number of term required. If the absolute accuracy of the system is approximately  $p/100$  m.u.,  $k$  must be chosen to satisfy:

$$R_k \leq p/100 \text{ m.u.}$$

but, assuming  $F(X)$  is analytic in  $X$ , then for some value of

$$x_A = x_A';$$

$$R_k = \frac{1}{(k+1)!} F^{k+1}(x_D) (x_A')^{k+1}; \quad x_D = 0, \pm 1, \dots, \pm(2^n - 1)$$

Since  $x_A \leq 1$

$$\text{Then} \quad R_k \leq \frac{1}{(k+1)!} F^{k+1}(x_D)_{\max} \leq \frac{p}{100}$$

Or

$$(k+1)! \geq 100/p \ F_{\max}^{k+1}(X_D); X_D = 0, \pm 1, \pm 2, \dots, \pm(2^n - 1)$$

satisfies the requirement that  $R_k$  be less than  $p/100$  m.u. Conversely, for a given number of terms in the approximation, this inequality places weak bounds on the class of functions that can be accurately approximated, viz.:

$$F_{\max}^{k+1} \leq \frac{p(k+1)!}{100} \text{ m.u.}$$

Example A:  $Y = KX^2$

As an introductory example, consider the generation of the function  $Y = F(X) = KX^2$ .  $K$  must be chosen subject to the limitations  $|F| \leq 2^n$ ,  $|F'| \leq 1$ ,  $|X| < 2^{n+1}$ . These restrictions lead to the inequalities

$$K \leq 1/2^n \quad (|F| \leq 2^n)$$

$$K \leq 1/2^{n+1} \quad (|F'| \leq 1)$$

The latter inequality is obviously the stronger limitation, so that the function to be generated in a hybrid system with  $n$  bits plus sign would be

$$Y = \frac{X^2}{2^{n+1}}$$

At the point  $X = X_D$  the function has the finite Taylor series expansion:

$$Y = \frac{X_D^2}{2^{n+1}} + \frac{X_D}{2^n} X_A + \frac{1}{2^{n+1}} X_A^2$$

Thus the error involved in using a linear segment approximation to the function is bounded by

$$R_K = R_1 = \frac{1}{2^{n+1}} X_A^2 < 1/2^{n+1}$$

For the case  $n = 3$ , this error is  $1/16$  m.u. which is probably too high (unless the analog system accuracy is worse than 6 per cent). Thus for  $n = 3$ , the generation of  $Y = KX^2$  would require the use of parabolic interpolation (squaring of  $X_A$ ) for an accuracy consistent with the other computing elements.

Example B:  $Y = KX^m$

The same scaling restrictions again apply; here we have:

$$|Y| = |KX^m| < 2^n; \quad |X| < 2^n; \quad |X_D| < 2^{n-1}$$

$$\frac{\partial Y}{\partial X} = mKX^{m-1} < 1$$

or

$$K \leq \frac{1}{m(2^n)^{m-1}} = \frac{1}{m2^{n(m-1)}}$$

The truncation error to be expected from using analog interpolations of order 1, 2, ...k can be estimated from

$$R_k \leq \frac{m(m-1) \dots (m-k+1)}{(k+1)! m2^{n(m-1)}} (X_D)^{m-k-1}; \quad X_D < 2^{n-1}$$

or

$$R_k \leq \frac{m(m-1) \dots (m-k+1)}{(k+1)! \, m 2^{n(m-1)}} (2^n - 1)^{m-k-1}$$

For  $n = 3$ , this becomes

$$R_k = \frac{7^{m-k-1} (m-1) \dots (m-k)}{8^{m-1} (k+1)!}$$

Table D.1 shows the truncation error to be expected from using a  $k$ -th-order approximation to  $KX^m$  for the case  $n = 3$ ,  $m = 1, 2, 3$ , and 4. The above analysis can be easily extended to cover the case of a general polynomial of any order. For a given resolution, a hybrid function generator with more digital bits and less analog accuracy would be simpler, since fewer high-order analog interpolation terms would have to be used.

Example C:  $Y = A \sin aX$

This example of a common transcendental function will be carried out for  $n = 3$  bits. Thus, the scaling limitations lead to the following:

$$|Y| = |F(x)| = |A \sin aX| \leq 2^3 = 8 \rightarrow A \leq 8$$

$$|F'| = |Aa \cos aX| \leq 1 \quad Aa \leq 1$$

To permit coverage of two quadrants ( $-\pi/2 < aX < \pi/2$ ), choose  $a = \pi/16$ . Then the maximum allowable value of  $A$  is  $16/\pi = 5.1$ . For simplicity, one would probably pick  $A = 5$ . Thus the function

to be generated would be  $Y = 5 \sin (\pi X/32)$ . The  $k$ -th derivative of  $F(X)$  is thus bounded by

$$\left| F^k \right| = \left| \frac{\partial^k}{\partial X^k} 5 \sin \pi X/32 \right| \leq 5(\pi/16)^k$$

and the resulting truncation error resulting from a  $k$ -th-order approximation is

$$R_k < \frac{F^{k+1}}{(k+1)!} \leq \frac{5(\pi/16)^{k+1}}{(k+1)!}$$

Table D.2 shows the accuracy to be expected from 1st, 2nd, and 3rd-order approximations to the function ( $n = 3$ ). It shows that if the analog accuracy of the hybrid system is 1 per cent of a machine unit, then a 2nd-order approximation is required to achieve a consistent degree of accuracy in the hybrid generation of  $Y = 5 \sin (\pi X/16)$ .

The above examples indicate that for  $n = 3$ , a hybrid function generator using combined linear and quadratic interpolations will generate a relatively broad class of analytic functions with an accuracy of 0.1 per cent of full-scale, or better. It should be emphasized that only a relatively crude quadratic interpolation would normally be required, utilizing a 2-3 bit precision in  $F^{ii}(x_D)$  and an analog squaring circuit of perhaps 5-10 per cent accuracy.

Figure 1.1c illustrates the general organization of a hybrid function generator. The digital logic would be relatively

complex and would perform the following operations:

- a. Upon receipt of a  $\Delta X$  input pulse, and the DC level  $S(\Delta X)$ , the  $X_D$  register would be updated. The existence of  $\Delta X$  would be stored for future use (the polarity of  $\Delta X$  would also be stored if for some reason it were not available throughout the entire digital operating cycle).
- b. A forward or backward difference  $\Delta F(X_D, \Delta X)$  would be added to the R-register, depending upon the existence and polarity of  $\Delta X$ . Some form of parallel or serial coding logic would be required to generate  $\Delta F(X_D, \Delta X)$ .
- c. Additional coding logic would generate  $F'(X_D)$  and  $F''(X_D)$  (and possibly higher-order derivative values) for use in the analog interpolation operations.
- d. Following the above digital operations, the comparator states would be interrogated, and if a carry were required, a  $\pm 1$  m.u. correction would be made to the R-register.

To simplify the set-up of different functions, some type of removable plugboard patching system for programming the digital logic operations would be desirable.

TABLE D.1

## ACCURACY OF GENERATION OF FUNCTION

$$Y = KX^m \quad (n = 3)$$

m	k	Accuracy: Per Cent of 1 m.u.	Accuracy: Per Cent of 16 m.u.
2	1	6.7	0.418
	2	0	0
3	1	10.9	0.68
	2	0.52	0.032
	3	0	0
4	1	14.4	0.9
	2	1.37	0.086
	3	0.049	0.0031
	4	0	0

TABLE D.2

## ACCURACY OF GENERATION OF FUNCTION

$$Y = F(X) = 5 \sin (\pi X/16); n = 3$$

Order of Approximation	Accuracy: Per Cent of 1 m.u.	Accuracy: Per Cent of Full-Scale (16 m.u.)
k=1 (linear)	9.6	0.6
k=2 (quadratic)	0.63	0.039
k=3 (cubic)	0.033	0.002

## Appendix E

### ANALYSIS OF ERRORS IN DIGITAL DIFFERENTIAL ANALYZERS

#### E.1 TRUNCATION ERRORS: Z-TRANSFORM APPROACH

A z-transform approach for estimating truncation errors in DDA's has been used by several investigators (see Ref. 7, 12, and 21). It is important to note that in closed loop computation, the integrators cannot use updated values of the input functions (see Ref. 25). At the same time, in order to keep the complexity of the logic systems in the DDA to a practical level, a relatively simple integration rule must be used. The above considerations usually make it necessary to restrict the choice of an integration rule to either the simple rectangular (Euler) rule or a two-point rule which hereafter will be referred to as the Open Trapezoidal rule. If  $x_n$  and  $\dot{x}_n$  are the values of the integrator output and input signals, respectively, at iteration times,  $t = nT$ , these rules may be written:

$$x_{n+1} = x_n + T \dot{x}_n \quad (\text{Rectangular})$$

$$x_{n+1} = x_n + T \left[ \frac{3}{2} \dot{x}_n - \frac{1}{2} \dot{x}_{n-1} \right] \quad (\text{Trapezoidal})$$

---

<sup>1</sup>In this Appendix,  $T$  corresponds to  $T_I$  in Chapter 5, i.e., it is the DDA iteration time.

Using z-transform methods, the sampled-data transfer function corresponding to these integration rules are:

Rectangular

$$\frac{X(z)}{\dot{X}(z)} = \frac{Tz}{(z-1)}$$

Trapezoidal

$$\frac{X(z)}{\dot{X}(z)} = \frac{T}{2} \frac{3z - 1}{z(z-1)}$$

where

$X(z)$  = z-transform of the function  $x_n = x(t = nT)$

$$X(z) = \sum_{n=0}^{\infty} z^{-n} x_n; \quad z = e^{sT}$$

The effect of truncation errors due to this sampled-data approximation to the desired integrator transfer function may be examined with a variety of techniques (see Ref. 10, 11, and 33). For the present discussion, a root perturbation method similar to that used by Nelson (Ref. 21) is chosen.

The technique presented here can be applied to any n-th order time-invariant linear system representable by the differential equation

$$a_n \frac{dx^n}{dt^n} + a_{n-1} \frac{dx^{n-1}}{dt^{n-1}} + \dots + a_1 \frac{dx}{dt} + a_0 = f(t)$$

Corresponding to the linear system there will be a characteristic equation

$$a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0$$

with  $n$  roots,  $s_k$ ; ( $k = 1, 2, \dots, n$ ) in the complex frequency plane ( $s$ -plane).

If the  $s_k$  are shown, it is then possible to find the location of the perturbed roots from the following procedure:<sup>1</sup>

- a. Given the actual integrator transfer function,  $F_T(z)$ , and the roots of the desired system,  $s_k$ , solve for the roots of the DDA difference equation,  $z_{ki}$ , in the  $z$ -plane given by

$$1/F_T(z) = s_k$$

- b. Find the corresponding roots of the difference equation in the  $s$ -plane,  $s_{ki}$  from the relationship:

$$s_{ki} = -1/T \ln z_{ki}$$

- c. The  $s_{ki}$  are the roots of the actual sampled data system.<sup>2</sup>

Comparison of the  $s_{ki}$  to the  $s_k$  will show how much effect the truncation errors have on the natural modes of response of the system, and will also show that, in general, spurious modes of response are generated.

---

<sup>1</sup>This method fails if the system has multiple roots (repeated eigenvalues).

<sup>2</sup>More precisely, the  $s_{ki}$  are the roots lying in the infinite strip  $-\pi/T < \text{Im} \{s\} < \pi/T$ , corresponding to the  $z_{ki}$ .

### E.1.1 RECTANGULAR (EULER) INTEGRATION

Assuming a DDA which uses integrators with the z-transform transfer function

$$F_T(z) = \frac{X(z)}{\dot{X}(z)} = \frac{Tz}{z-1}$$

The effect of truncation errors on the characteristic roots of a linear system can be analyzed as follows:

Let  $s$  be a root of the unperturbed system in question.

Then 
$$1/F_T(z) = s$$

Or 
$$\frac{z-1}{Tz} = s$$

Then 
$$z(1 - sT) = 1$$

Or 
$$z(T) = 1/1 - sT$$

The Taylor series expansion of  $z(T)$  yields

$$z(T) = 1 + sT + (sT)^2 + (sT)^3 + \dots (sT)^n + \dots$$

Note that if there were no root perturbation, the Taylor series for  $z(T)$  would be

$$z(T) = e^{sT} = 1 + sT + \frac{(sT)^2}{2} + \frac{(sT)^3}{6} + \dots \frac{(sT)^n}{n!} + \dots$$

It is then apparent that truncation errors arise from the fact that  $z(T) \neq e^{sT}$

### E.1.1.1 ROOT PERTURBATION: RECTANGULAR INTEGRATION

It can be shown that

$$\ln X = (X-1) + \frac{1}{2} (X-1)^2 + \frac{1}{3} (X-1)^3 - \frac{1}{4} (X-1)^4 + \dots$$

Substituting

$$z(T) \simeq 1 + sT + (sT)^2 + (sT)^3 + (sT)^4$$

in the above expression, we have

$$\ln z(T) \simeq sT + (sT)^2/2 + (sT)^3/3$$

Now let  $s_1(T)$  be the root in the s-domain corresponding to  $z(T)$ ; i.e.,

$$z(T) = e^{s_1(T)T}$$

Or

$$s_1(T) = 1/T \ln (T)$$

Then

$$s_1(T) \simeq s + (sT)^2/2 + (sT)^3/3$$

Or

$$s_1(T) \simeq s \left( 1 + sT/2 + (sT)^2/3 \right)$$

This last equation shows that the truncation error due to rectangular integration produces a shift in the system roots of order  $(sT)$ .

### E.1.2 OPEN TRAPEZOIDAL INTEGRATION

Assuming a DDA which uses integrators with the z-transform transfer function

$$F_T(z) = \frac{X(z)}{X(z)} = \frac{T}{2} \frac{(3z-1)}{z(z-1)}$$

The effect of truncation errors on the characteristic roots of a linear system can be analyzed as follows:

Let  $s$  be a root of the unperturbed system in question.

$$\text{Then} \quad 1/F_T(z) = s$$

$$\text{Or} \quad \frac{z(z-1)}{3z-1} \cdot \frac{2}{T} = s$$

$$\text{Then} \quad \frac{2}{T} \cdot \frac{z^2}{s} - (3 + 2/sT)z + 1 = 0$$

$$\text{Or} \quad z(T) = 3sT/4 + 1/2 \pm \sqrt{9/16(sT)^2 + sT/4 + 1/4}$$

It is already apparent that one effect of the difference-equation approximation is the creation of two roots in the z-domain for each root of the original system.

The Taylor series expansion of  $z(T)$  yields

$$\text{Major root:} \quad z_A(T) \simeq 1 + sT/2 + (sT)^2/2 - (sT)^3/4 - (sT)^4/8$$

$$\text{Minor root:} \quad z_B(T) \simeq sT/2 - (sT)^2/2 + (sT)^3/4 + (sT)^4/8$$

Note that if there were no root perturbation, we should find only one root in the  $z$ -domain, viz.:

$$z(T) = e^{sT} \simeq 1 + sT + (sT)^2/2 + (sT)^3/6 + (sT)^4/24$$

It is then apparent that the truncation errors arise for two reasons:

a.  $z_A(T) \neq e^{sT}$

b.  $z_B(T) \neq 0$

#### E.1.2.1 PERTURBATION OF THE PRINCIPAL ROOT: TRAPEZOIDAL INTEGRATION

It can be shown that

$$\ln x \simeq (x-1) - \frac{1}{2} (x-1)^2 + \frac{1}{3} (x-1)^3 - \frac{1}{4} (x-1)^4$$

Substituting

$$z_A(T) \simeq 1 + sT + (sT)^2/2 - (sT)^3/4 - (sT)^4/8$$

in the above expression, we have

$$\ln z_A(T) \simeq sT - 5/12(sT)^3 + 1/4(sT)^4$$

Now let  $s_2(T)$  be the root in the  $s$ -domain corresponding to  $z_1(T)$ ;

i.e.,

$$z_A(T) = e^{s_2 T}$$

Or

$$s_2 = 1/T \ln z_A(T)$$

Then

$$s_2(T) \simeq s - 5/12 s^3 T^2 + 1/4 s^4 T^3 + \dots$$

$$s_2(T) \simeq s \left( 1 - 5/12 (sT)^2 + 1/4 (sT)^3 \right)$$

This last equation shows that the truncation error produces a shift in the system roots of the order  $(sT)^2$ .

### E.1.3 AN EXAMPLE: THE SINE LOOP

Consider now the simple sine-loop configuration of Figure 2.1. If ideal integrators are used, the system has the characteristic equation:

$$s^2 + \omega_0^2 = 0; a_1 a_2 = \omega_0^2$$

with corresponding roots:

$$s_1 = \pm j \omega_0$$

#### E.1.3.1 RECTANGULAR INTEGRATION

From the equation derived in Section E.1.1.1, the root perturbation caused by rectangular integration can be estimated:

$$s_1(T) \simeq \pm j \omega_0 \left[ 1 - \frac{(\omega_0 T)^2}{3} \right] + \frac{\omega_0^2 T}{2}$$

It can be seen that the principal effect on the root location is a shift of the system roots into the right half of the s-plane. There is also a slight reduction in natural frequency. The expressions for the relative errors in the location of the system roots are listed in Table E.1.

#### E.1.3.2 TRAPEZOIDAL INTEGRATION

From the equation derived in Section E.1.2.1, the perturbation of the principal system roots can be estimated:

$$s_2(T) \simeq \pm j \omega_0 (1 + 5/12 (\omega_0 T)^2) + 1/4 \omega_0^4 T^3$$

Here the principal effect on the root location is an increase in the natural frequency. There is also a slight shift of the system roots into the right half of the s-plane. The expressions for the fractional errors in the location of the system roots are listed in Table F.1.

#### E.1.4 COMPARISON OF THE TRUNCATION ERRORS: RECTANGULAR AND TRAPEZOIDAL INTEGRATION

Figures 5.2 and 5.3 and Tables 5.1 and E.1 summarize the effect of the truncation errors in these two integration rules. As expected, the truncation errors are greater for the rectangular rule. For the case of the undamped sinusoid, the frequency errors are about the same, however, the damping error is much higher for the rectangular case. Error estimates can be made for system roots

located elsewhere in the s-plane by the same method. The effect of the root perturbations on the actual solution values may be examined by the use of sensitivity equations similar to those developed in Appendix F.

## E.2 ROUND-OFF ERRORS

Computing errors due to round-off are not easily estimated, but assuming that round-off produces a relatively incoherent quantization error with a uniform amplitude distribution, Widrow (Ref. 38) has used statistical techniques to show that the approximate RMS round-off error is  $\Delta X / \sqrt{12} \simeq 0.3 \Delta X$ , where  $\Delta X$  is the value of the digital quantization level. When the quantization interval is small, as would be the case in accurate computation, the maximum absolute value of the machine round-off error can be estimated to be approximately equal to  $\Delta X$  (see also Ref. 9 and 25).

For purposes of this discussion, a parallel-organized incremental DDA is assumed (this class of DDA's includes most modern special-purpose machines). In an incremental DDA, the maximum rate-of-change of machine variables is  $\Delta X / T_I$ . Thus, for a full-scale sine wave

$$X(t) = A \sin \omega t$$

$$\left| dX/dt \right| < A \omega$$

If the round-off error is assumed to be approximately  $\Delta X$ :

$$\text{round-off error} \simeq \Delta X < A \omega T_I$$

Or

$$\Delta X/A > \omega T_I = 2\pi f T_I$$

Thus, for incremental machines, the round-off error (expressed as a per cent of half-scale) is

$$e_{\text{ROUND-OFF}} \simeq \omega T \cdot 100 = 200\pi f T_I \text{ per cent}$$

Since most modern commercial DDA's use open trapezoidal integration and incremental data transfer, it is reasonable to assume that in precise computations ( $T_I \ll 1/\omega$ ) the round-off errors will predominate. Thus the accuracy-bandwidth capability of typical DDA's can be characterized by the expression

$$\frac{\text{error}}{\text{frequency}} \simeq 2\pi T_I 100 \text{ per cent; } (T_I \ll 5/12\omega)$$

Equivalently, DDA's of this type have a capability of computing approximately  $1/T_I$  distinguishable increments-per-second.

Figure 5.1 also shows the gross accuracy-bandwidth capability for a parallel-organized incremental DDA, using trapezoidal integration.

Table E.1

FRACTIONAL ERRORS IN THE LOCATION OF  
THE CHARACTERISTIC ROOTS OF A SINE LOOP<sup>\*</sup>

Integration Rule	$\alpha/\omega_0$ , Per Cent	$\Delta\omega/\omega_0$ , Per Cent
Rectangular	$\omega_0 T/2$	$-\frac{(\omega_0 T)^2}{3}$
Trapezoidal	$1/4 (\omega_0 T)^3$	$+ 5/12 (\omega_0 T)^2$

<sup>\*</sup>Perturbed solution is assumed to be of the form

$$X(t) = A e^{\alpha t} \cos (\omega_0 + \Delta\omega)t$$

with roots,  $s = \alpha + j(\omega_0 + \Delta\omega)$

## Appendix F

### ROOT PERTURBATION EFFECTS

#### F.2 DECAYING EXPONENTIAL

Consider the time function:

$$X(t) = A e^{-\alpha t}$$

To estimate the effect of a perturbation in the parameter  $\alpha$  on the value of the function  $X$ , we differentiate  $X$  with respect to  $\alpha$ :

$$\frac{\partial X}{\partial \alpha} = -At e^{-\alpha t}$$

At a particular time  $t = T$ ,

$$\frac{\partial X}{\partial \alpha} = -A T e^{-\alpha T} = -T X(T)$$

Thus for a small perturbation,  $\Delta\alpha$ , we find the error in the value of  $X(T)$  is

$$\Delta X = -T X(T) \Delta\alpha = -T A e^{-\alpha T} \Delta\alpha$$

If we consider  $X$  as an error in the value of  $X(T)$ , then we can find the time  $T$  when the error is greatest

$$\begin{aligned} \frac{\partial (\Delta X)}{\partial T} &= - \left[ A e^{-\alpha T} - A \alpha T e^{-\alpha T} \right] \Delta\alpha \\ &= -X(T) [1 - \alpha T] \Delta\alpha \end{aligned}$$

Setting

$$\frac{\partial \Delta X}{\partial T} = 0$$

We find that for  $A \neq 0$ ,  $\Delta \alpha \neq 0$

$$T = 1/\alpha = \text{one time constant.}$$

In words, the error in  $X(t)$  is most sensitive to an error in  $\alpha$  at  $t = \text{one time constant}$ .

$$\Delta X_{\max} = -A/e (\Delta \alpha / \alpha)$$

In the decaying exponential problem discussed in Section 4.3,

$A = 60$  volts,  $\alpha = 50$ :

$$\Delta X_{\max} = 22 \frac{\Delta \alpha}{\alpha} \text{ volts; at } T = 1/50 \text{ sec}$$

or  $k = 16 \text{ rms}$

or  $\frac{\Delta \alpha}{\alpha} = 0.045 \Delta X$

Thus, in this problem, it is difficult to detect small errors in  $\alpha$ , since a 0.45 per cent error in  $\alpha$  is required to cause only a 0.1 volt error at  $t = \text{one time constant}$ . The actual error curves shown in Figures 4.3b and 4.5d, e, and f did not reveal any significant information that could lead to a conclusion that the solution errors were necessarily due to an error in  $\alpha$ .

## F.2 SINE LOOP

### F.2.1 FREQUENCY ERRORS

Consider the time function:

$$X(t) = A e^{-\alpha t} \sin \omega t$$

To estimate the effect of a perturbation in the parameter on the value of  $X$ , we again differentiate:

$$\frac{\partial X}{\partial \omega} = A \left[ e^{\alpha T} \cos \omega T + \sin \omega T e^{\alpha T} \right]$$

This error in  $X$  has local extrema after  $N$  cycles, that is, at  $T = 2\pi N/\omega$ ;  $N = 0, 1, 2, \dots$ ; at these times

$$\Delta X = 2\pi AN \frac{\Delta \omega}{\omega}$$

or

$$\frac{\Delta \omega}{\omega} = \frac{\Delta X}{2\pi AN}$$

Thus computer solution error due to an error in the natural frequency of the computer loop can be studied by observing the amount of solution error (in volts) which occurs after a solution time of  $N$  periods. After several cycles, this error will usually be large enough to permit detection of even small errors in  $\omega$ . Consider the problem of Section 4.4, where  $A = 60$ ,  $\omega = 50$ , we have after five cycles ( $N = 5$ ):

$$\frac{\Delta \omega}{\omega} = \frac{\Delta X}{600\pi}; \Delta X \text{ measured after five cycles of computing of } X = 60 \sin 50t \text{ (k = 502.65)}$$

### F.2.2 DAMPING ERRORS

If the computer solution for the sine loop configuration has any exponential decay or growth, this will be most evident at the extrema of the solution. Consider the function

$$X(t) = A e^{-\alpha t} \cos \omega t$$

By a similar analysis to that in Section F.2.1, one finds that after  $N$  cycles of a cosine function, the solution error is related to  $\alpha$  by:

$$\Delta X = 2\pi AN \frac{\alpha}{\omega}$$

or

$$\frac{\alpha}{\omega} = \frac{\Delta X}{2\pi AN}$$

Again, in the problem of Section 4

$$\frac{\alpha}{\omega} = \frac{\Delta X}{600\pi} ; \Delta X \text{ measured after five cycles of computing } X = 60 \cos 50t$$

( $k = 502.65$ )

## REFERENCES

1. Bohn, E. V., "A PPM Analog Computer," IRE Trans. PGEC, Vol. EC-9, No. 1, Sept., 1960, 256 ff.
2. Bradley, R. E., and Genna, J. F., Design of a 1 MC Iteration Rate DDA, Hazeltine Technical Development Center, Inc., Indianapolis, Indiana, circa, 1961.
3. Brubaker, T. A., "Precision Analog Memory has Extended Frequency Response," Electronics, Sept. 29, 1961, pp. 141-143.
4. Connelly, M. E., "Real-Time Analog-Digital Computation," IRE Trans. PGEC, Vol. EC-11, No. 1, Feb., 1962, pp. 31-41.
5. Fernandez, M., and Macomber, G., Inertial Guidance Engineering, Englewood Cliffs, New Jersey: Prentice-Hall, 1962.
6. Gibbons, J. P., Experiences With a DDA-GPDC Hybrid System, National Aviation and Space Agency, Huntsville, Alabama, March, 1962.
7. Gilliland, M. C., "The Spectral Evaluation of Iterative Differential Analyzer Integration Techniques," Proc. West. Joint Computer Conference, Vol. 18, May, 1961, pp. 507-518.
8. Gray, H. J., Jr., "Numerical Methods in Digital Real-Time Simulation," Quar. of Appl. Math., Vol. 12, No. 2, July, 1954, pp. 133-140.
9. Gschwind, H. W., "Digital Differential Analyzers," Electronic Computers, Edited by P. Von Handel, Englewood Cliffs, New Jersey: Prentice-Hall, 1961.
10. Gurk, H. M., "The Use of Stability Charts in Synthesis of Numerical Quadrature Formulae," Quar. Appl. Math., Vol. 13, No. 1, April, 1955, pp. 73-78.
11. Hamming, R. W., Numerical Methods for Scientists and Engineers, New York: McGraw-Hill, 1962.
12. Hills, F. B., A Study of Incremental Computation by Difference Equations, Electronic Systems Laboratory Report No. 7849-R-1, Cambridge, Massachusetts: Mass. Inst. of Technology, May, 1958.
13. Hills, F. B., "Characteristics and Limitations of Digital Computer Solutions to Differential Equations," Draft Ph.D. Thesis, Mass. Inst. of Technology, circa, 1959.

14. Koelle, H. H., Handbook of Astronautical Engineering, New York: McGraw-Hill, 1961.
15. Korn, G. A., and Korn, T. M., Electronic Analog Computers, Draft 3rd Edition, New York: McGraw-Hill, in preparation.
16. Korn, G. A., "Exact Design Equations for Operational Amplifiers with Four-Terminal Computing Networks," IRE Trans. PGEC, Vol. EC-11, No. 1, Feb., 1962, pp. 82-83.
17. Korn, G. A., "The Impact of the Hybrid Analog-Digital Techniques on the Analog Computer Art," Proc. IRE, Vol. 50, No. 5, May, 1962 Anniv. Issue, pp. 1077-1086.
18. Maybach, R. L., O'Grady, E. P., and Wait, J. V., "A Master Control Clock for a Hybrid Differential Analyzer," Dept. of Elec. Engr. ACL Memo No. 81, Univ. of Arizona, 1963.
19. Meissinger, H. F., "Parameter Influence Coefficients and Weighting Functions Applied to Perturbation Analysis of Dynamic Systems," Proc. 3rd Int. Congress on Analog Computation, Opatija, Yugoslavia, Sept., 1961.
20. Mitchell, J. M., and Ruhman, S., "The Trice - A High Speed Incremental Computer," IRE Nat. Conv. Record, 1958, Pt. 4, pp 206-216.
21. Nelson, D. J., "DDA Error Analysis Using Sampled Data Techniques," Proc. West. Joint Comp. Conf., 1961.
22. Nelson, D. J., A Foundation for the Analysis of Analog-Oriented Combined Computer Systems, Radioscience Laboratory Report TR 1002-1, SEL-62-069, Stanford, California, April, 1962.
23. Packard Bell Electronics, "Impact Prediction by Means of Incremental Computing," Applications Note Prepared by Manufacturer Los Angeles, California, circa, 1961.
24. Packard Bell Electronics, "An Impact Predictor for Ballistic Missiles," Applications Note Prepared by Manufacturer, Los Angeles, California, circa, 1961.
25. Palevsky, M., "The Digital Differential Analyzer," Computer Handbook, Edited by G. A. Korn and H. D. Huskey, New York: McGraw-Hill, 1961, Chapter 19.
26. Peterson, G. R., "A Discussion of Differential Analyzer Arithmetic," Ph.D. Dissertation, Univ. of Arizona, 1961.

27. Schmid, H., A High-Accuracy Partial Analog to Digital Converter, Link Aviation Report DLR 557, Binghamton, New York, Aug., 1960.
28. Schmid, H., High Accuracy, Single-Variable, Linear Segment Hybrid Function Generator, Link Aviation Report DLR 558, Binghamton, New York, Oct., 1960.
29. Schmid, H., "Combined Analog-Digital Computing Elements," Proc. West. Joint. Comp. Conf., May, 1961.
30. Schmid, H., "Linear-Segment Function Generator," IRE Trans. PGEC, Vol. EC-11, No. 5, Dec., 1962, pp 780-788.
31. Skramstad, H. K., "A Combined Analog-Digital Differential Analyzer," Proc. East. Joint Comp. Conf., Dec., 1959, pp 94-100.
32. Truitt, T. D., Introduction to DDA's, Technical Report, Princeton, New Jersey: Electronic Associates, Inc., undated.
33. Vichnevetsky, R., A Spectral Method of Analyzing the Truncation Error in the Finite Difference Technique, Technical Report of the European Computation Centre, Brussels, Belgium: Electronic Associates, Inc., circa, 1961.
34. Wait, J. V., and Mitchell, B. A., "A Simple Solid-State Digital-to-Analog Converter for Hybrid Computing Systems," Dept. of Elec. Engr., ACL Memo No. 61, Univ. of Arizona, Feb. 10, 1963.
35. Wait, J. V., "An Interpolation Waveform Generator for use in Hybrid Computing Systems," Dept. of Elec. Engr., ACL Memo No. 64, Univ. of Arizona, Feb. 11, 1963.
36. Wait, J. V., and Hampton, R. L. T., "A Solid-State Analog Comparator for Hybrid Analog-Digital Computers," Dept. of Elec. Engr., ACL Memo No. 63, Univ. of Arizona, Jan. 20, 1963.
37. Wait, J. V., "A Read-Out System for a Hybrid Differential Analyzer," Dept. of Elec. Engr., ACL Memo No. 80, Univ. of Arizona, May, 1963.
38. Widrow, B., "Statistical Analysis of Amplitude-Quantized Sampled-Data Systems," AIEE Applications and Industry, Vol. 52, Jan., 1961, pp. 555-567.
39. Brubaker, T. A., "The Design, Development and Applications of the ASTRAC Computer," ACL Memo No. 76, University of Arizona, 1963.
40. Schmid, H., "A Hybrid Operational Computing System," Rough Draft of Master's Thesis, Syracuse University, 1963.