

AN ABSTRACT OF THE THESIS OF

WILLIAM HAROLD CHARNHOLM for the Master of Science
(Name) (Degree)

Electrical and
in Electronics Engineering presented on May 3, 1971
(Major) (Date)

Title: DESIGN AND EVALUATION OF AN ADAPTIVE ELEMENT
WHICH COMPENSATES FOR DEADBAND AS APPLIED TO
A GUIDANCE SYSTEM

Abstract approved Redacted for Privacy
S. A. Stone

Employing compensators which adapt to errors in system components is one way of meeting the increasingly tight specifications of modern control systems. Indeed this is often the only way that such specifications can be met in systems with unchangeable or best available parts. One type of error found in components is deadband (i. e., the output does not respond to the input for small inputs). This thesis shows the design and evaluation of a digital compensator which minimizes deadband errors.

To illustrate deadband compensation, a guidance system is chosen which guides a ship to a specific point on the ocean called the target. The control equations for this guidance system automatically compensate for forces acting on the ship, such as currents and wind, which cause it to drift off course and miss the target. During actual

cruises however, it did drift off course and it was determined that deadband associated with the ship's response to rudder displacements was the cause. Four different deadband compensators were tested but only one, an adaptive element, gave the guidance system the desired accuracy. The adaptive element generated sufficient artificial error to cause the ship to return to the desired heading despite the deadband. The general principle of this element can be used in a variety of other closed loop systems which suffer from the effects of deadband.

Design and Evaluation of an Adaptive Element
Which Compensates for Deadband as Applied
to a Guidance System

by

William Harold Charnholm

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

June 1972

APPROVED:

Redacted for Privacy

Professor of Electrical and Electronics Engineering
in charge of major

Redacted for Privacy

Head of Department of Electrical and Electronics Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented May 3, 1971

Typed by Muriel Davis for William Harold Charnholm

ACKNOWLEDGMENT

I would like to thank my major professor S. A. Stone and Mr. D. O. Crozier for their valuable criticism of this thesis. Also I wish to express my thanks to Miss Martinez who typed the original manuscript.

TABLE OF CONTENTS

<u>Chapter</u>		<u>Page</u>
I	EXAMPLE OF SYSTEM WITH DEADBAND	1
	1.0 Description of System	2
	1.1 Field Results	8
II	COMPUTER MODEL OF SHIP GUIDANCE SYSTEM	11
	2.0 Description of Model	11
	2.1 Verification of Model	15
III	DEADBAND COMPENSATING TECHNIQUES	21
	3.0 Introducing Symmetrical Bias	21
	3.1 Dithering the System	21
	3.2 Increasing System Open Loop Gain	24
IV	ADAPTIVE DEADBAND COMPENSATOR	26
	4.0 Description of Adaptive Compensator	26
	4.1 Computer Model of Compensator	28
	4.2 Adaptive Compensator Results of Ship Guidance System Model	29
V	CONCLUSION	33
	BIBLIOGRAPHY	
	APPENDICES	35
	I ADAPTIVE COMPENSATOR WIRING DIAGRAM	35
	II GUIDANCE SYSTEM COMPUTER SIMULATION LISTING WITH ADAPTIVE DEADBAND COM PENSATOR	53
	III INTEGRATED CIRCUITS COMPUTER SIMULATION LISTING USED IN THE ADAPTIVE DEADBAND COMPENSATOR	59

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	Geometry of Guidance System	3
2	Geometry of Guidance System Viewed on a Flat Earth	5
3	Guidance System Block Diagram	7
4	Path of Ship without Deadband Compensation	9
5	Guidance System Model Block Diagram	12
6	Ship Hardware Model Block Diagram	14
7	Deadband Model	14
8	Listing of Computer Simulation of Guidance System without Deadband Compensation	16
9	Simulated and Actual Path of Ship without Deadband Compensation	18
10	Simulated Cross Course Velocity as a Function of Time for the Guidance System without Deadband Compensation	19
11	Three Types of Deadband Compensators	22
12	Symmetrical Bias Deadband Compensator	23
13	Dither Deadband Compensator	23
14	The Effect of Increasing Open Loop Gain on Deadband	25
15	Adaptive Deadband Compensator's Characteristics	27
16	Adaptive Deadband Compensator with Decaying Input	30
17	Simulated Path of Ship with Adaptive Deadband Compensation	31

<u>Figure</u>		<u>Page</u>
18	Adaptive Compensator's Input and Output Signals as a Function of Time for a Simulated Run	32
I-1	Adaptive Compensator Block Diagram	36
I-2	Direction Detector Wiring Diagram	40
I-3	Logic Box	41
I-4	Counter Wiring Diagram	46
I-5	Selector Switch Wiring Diagram	48
I-6	Adder Wiring Diagram	50
I-7	Adder and Subtractor Block Diagram	52
II-1	Listing of Computer Simulation of Guidance System with Adaptive Deadband Compensator	54
III-1	Listing of Computer Simulation of Integrated Circuits Used in Adaptive Deadband Compensator	60

LIST OF TABLES

<u>Table</u>		<u>Page</u>
I-1	Selector Switch Truth Table	42
I-2	Logic Box Truth Table	44

DESIGN AND EVALUATION OF AN ADAPTIVE ELEMENT WHICH COMPENSATES FOR DEADBAND AS APPLIED TO A GUIDANCE SYSTEM

I. EXAMPLE OF SYSTEM WITH DEADBAND

Deadband is a non-linearity characterized by a lack of response to small inputs. Although seldom designed into a system, it is frequently present in the hardware, with detrimental effects on performance. As an example of a system containing deadband, a ship guidance system is chosen. This system operates as a closed loop to guide various ships to a pre-determined point on the ocean called the target.

The system loop can be broken down into three main portions: the control unit, the ship's hardware, and the ship's kinematics. The control unit, atop a stationary platform in the ocean, monitors the position of the ship with a radar. By processing this information through a digital computer coupled to a transmitter, the control unit sends rudder angle commands to the ship. Hardware aboard the ship receives these commands and converts them into rudder displacements. The kinematics of the ship define its response to rudder displacements in terms of its position on the ocean. This position is monitored by the control unit, thus closing the system loop.

Deadband is contained in that portion of the kinematics which defines the ship's response to rudder displacements. For small

rudder displacements, no change in ship heading occurs. This causes the ship to slowly drift off course and miss the target.

1.0 Description of System

The system is employed in the oil industry where oil exploration at sea is conducted. Contact must be maintained between oil platforms and the shore in order to carry provisions and other necessary supplies. This is accomplished by a ship guidance system which can operate under all visibility conditions. A diagram of the complete system with the control unit atop an oil platform and the ship in the ocean below is shown in Figure 1.

The ship, cruising at a constant speed, is to reach a target oil platform. A low visibility condition exists and navigation must be performed automatically. It is necessary that the ship pass within 100 feet of the target in order for its pilot to see it and dock the ship manually.

Guidance commands are computed by a control unit stationed atop a master oil platform, which monitors the ship's position by a radar in terms of the radar-to-ship spherical variables: range (RRS), azimuth (ARS), and elevation (ERS). Target oil platform coordinates are inserted into the control unit before a run in terms of its north (XT) and east (YT) distances from the control unit. From these five signals, which define the position of the ship and the target relative

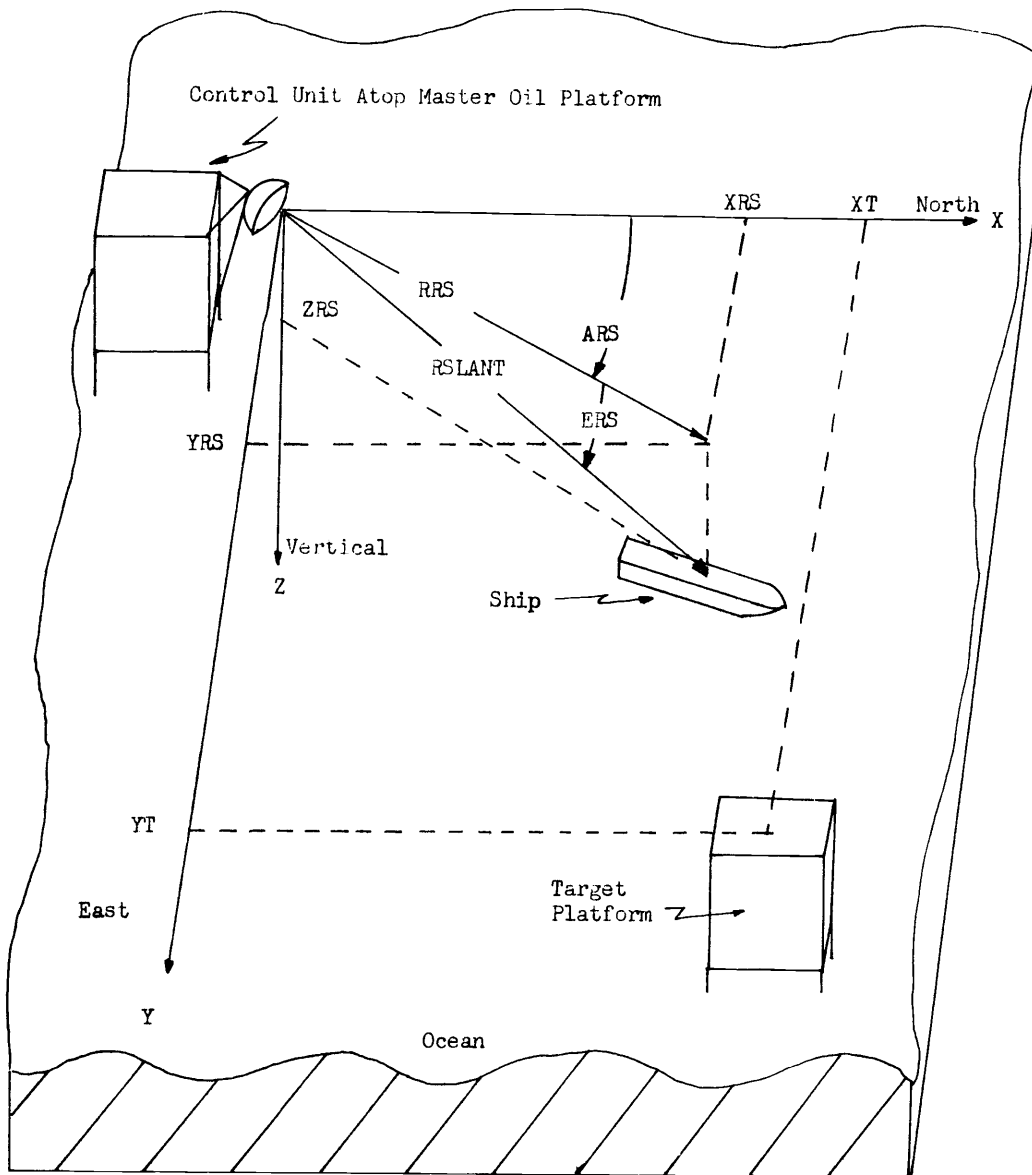


Figure 1. Geometry of Guidance System

to the radar, the control unit generates rudder angle commands to guide the ship to its target.

The control unit performs its computations based on the assumption that the earth is flat. This assumption is justified since the errors introduced are small as compared to the desired system accuracy. For example, earth curvature accounts for altitude and range errors less than three feet for a range of 10,000 feet between the radar and target platform. The desired miss distance accuracy of the system, 100 feet, is greater by more than an order of magnitude than this error. Thus, the control equations can be developed by viewing Figure 1 on a flat ground plane, as shown in Figure 2.

In Figure 2, the ship is cruising with a speed and heading designated by V_S and S_I , respectively. It is at a range (R_{ST}) from the target oil platform. The guidance control law is to minimize the component of the ship's velocity vector which is perpendicular to a line drawn from the ship to the target (R_{ST}). This velocity component is called cross course velocity and is designated as V_{CC} in Figure 2.

The control unit calculates V_{CC} by generating a coordinate system centered at the ship. This system, called the inertial axis system, has two axes; one in a north-south direction and the other in an east-west direction. The position of the target is calculated in this coordinate system and appears to move as the ship approaches it. The target's position with respect to the ship in terms of north (X_{ST})

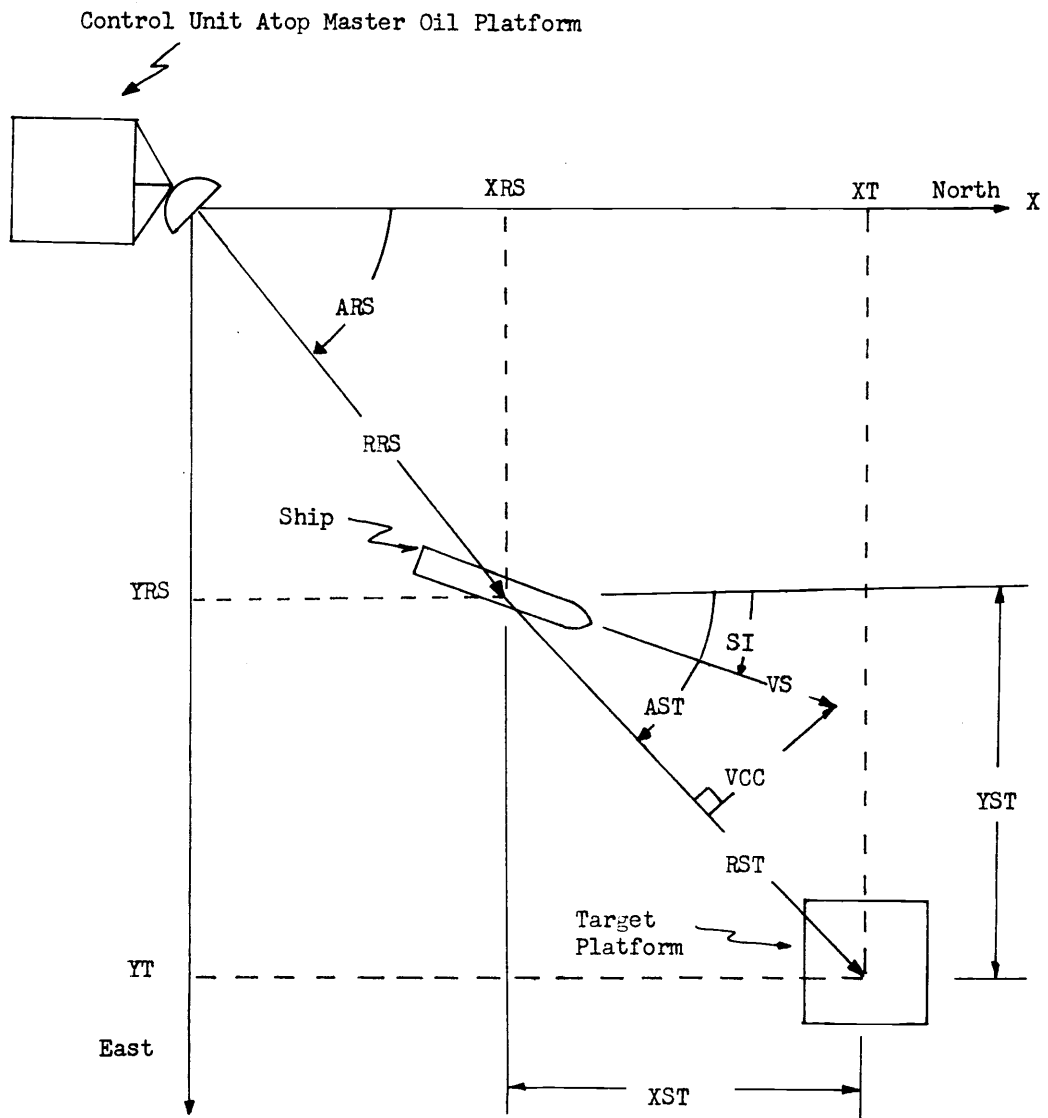


Figure 2. Geometry of Guidance System Viewed on a Flat Earth

and east (YST) coordinates are converted into ship-to-target range (RST) and azimuth angle (AST). It is in this coordinate system that the cross course velocity is calculated.

Target movement, as measured in the coordinate system centered at the ship, has two orthogonal velocity components; that found by the derivative of RST, and that found by the derivative of AST multiplied by RST. The derivative of RST is the component of the ship's velocity which will cause the ship to hit the target while the derivative of AST multiplied by RST is that component of the ship's velocity which will cause it to miss the target. This second component of the ship's velocity is called cross course velocity and is used as an error signal in the ship guidance control loop.

From the ship's cross course velocity, a rudder command is calculated by a simple sign inversion and gain multiplication. This command is sent to the ship and its response in the ocean is monitored by the radar unit. A block diagram of the closed loop is shown in Figure 3.

In Figure 3, commanded rudder displacements (DEL_C) sent to the ship are processed through a receiver and actuator unit which drive the rudder to an angle (DEL_R). The next block represents the ship's turning inertia in the water and its position in the ocean with respect to the control unit radar. Using this position in terms of radar-to-ship azimuth angle (AR_S) and range (RR_S), the control unit

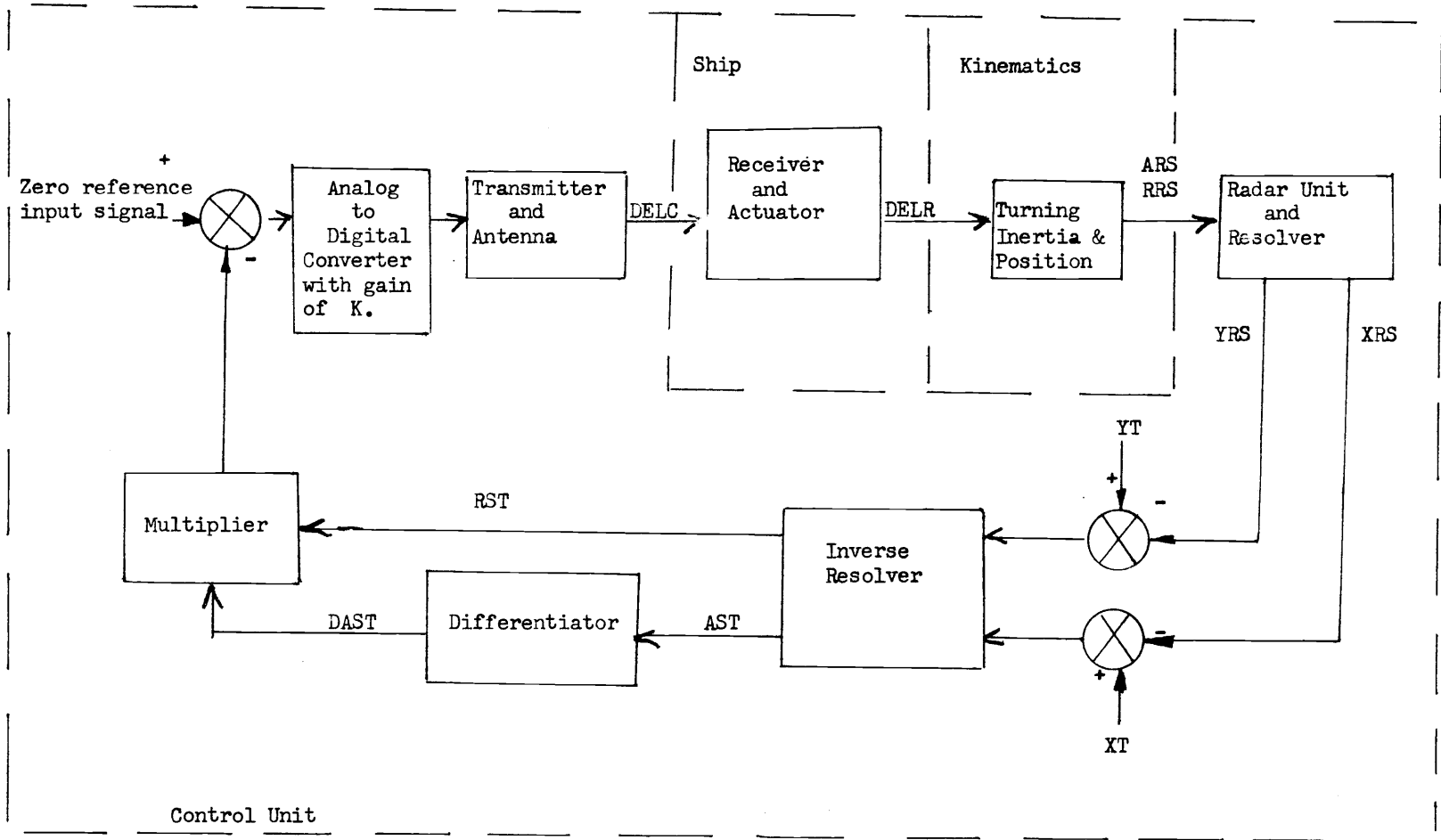


Figure 3. Guidance System Block Diagram

calculates ship position in cartesian coordinates (XRS, YRS). Subtracting XRS and YRS from the target oil platform position (XT, YT) results in a new cartesian coordinate system centered at the ship. An inverse servo resolver converts the new cartesian system to polar coordinates in terms of ship-to-target range (RST) and azimuth angle (AST). The derivative of AST is designated as DAST in Figure 3 and is generated by a tachometer connected to the inverse resolver. From DAST and RST the control unit uses a servo multiplier to calculate both the sign and magnitude of VCC. This signal is sent through an analog-to-digital converter with a gain of K to generate rudder commands to the ship, thus closing the control loop.

1.1 Field Results

Several test runs were made at a constant speed of 40 feet/second. The target was 10,000 feet from the ship when control became automatic. The initial orientation of the ship's velocity vector with respect to the target was parametrically varied, with data recorded from both the control unit atop the master oil platform and the ship.

Plots of the ship cruising to the target were recorded from the control unit. In most of these runs the ship developed errors in its heading and missed the target. An example of the data recorded is shown in Figure 4 where the ship was initially heading north. As can be seen, the system did not meet the desired accuracy of 100 feet

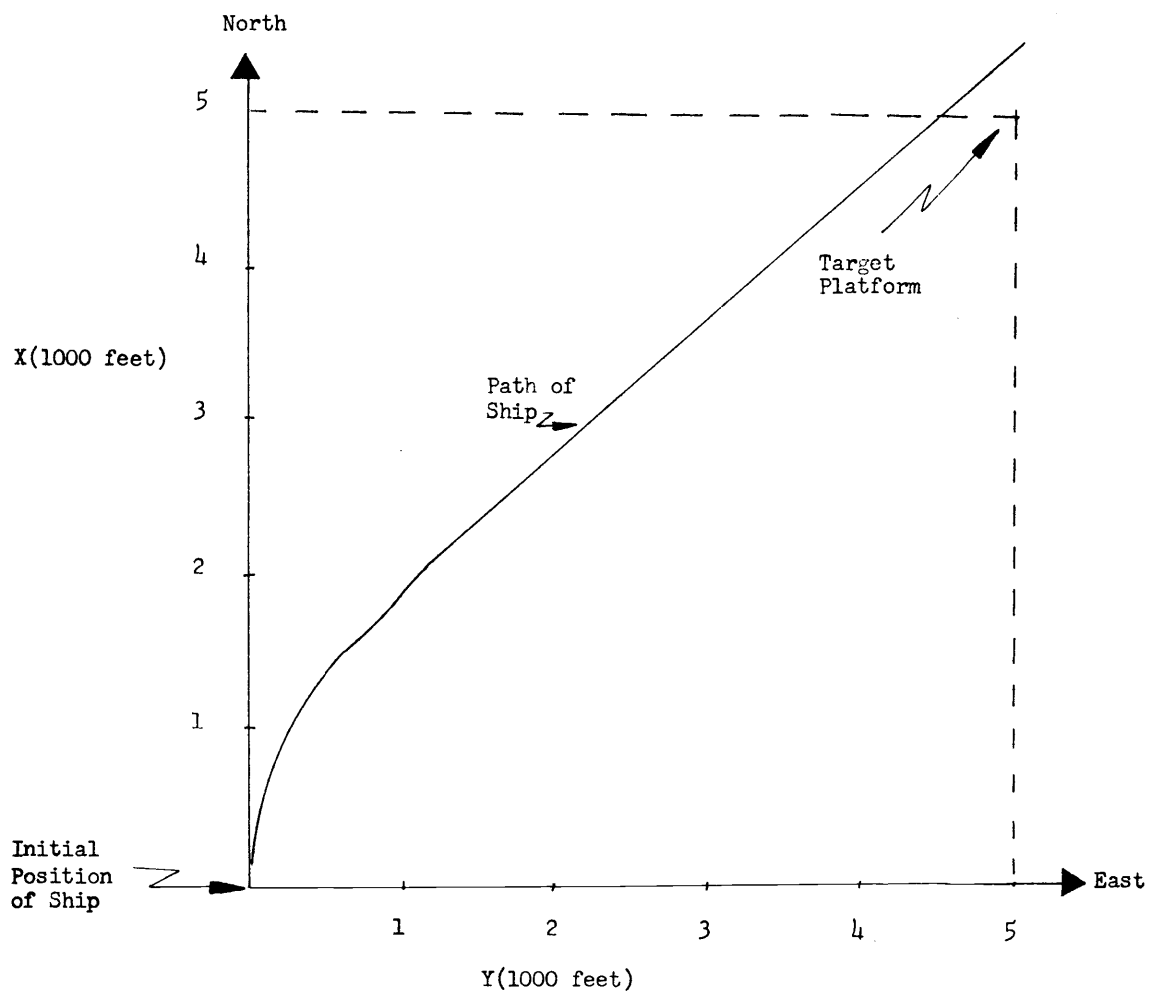


Figure 4. Path of Ship Without Deadband Compensation

for it missed the target by approximately 500 feet. To isolate the cause of this error, the ship's heading, as monitored with a heading gyro, was checked against rudder deflections made during the run. It was concluded that the ship responded correctly to large rudder commands. For small rudder commands, however, no heading rate was generated, thus showing that the ship exhibited deadband characteristics in the water.

Tests were made on other ships to determine the cause and size of the deadband. It was found that it was caused by the ship's hull profile and could vary from one to ten degrees, depending on the type of ship. To diminish the effect of the deadband, modifications were made on the control unit. This was because a variety of ships could be guided by the control unit and it would be expensive to change all of these.

II. COMPUTER MODEL OF SHIP GUIDANCE SYSTEM

Before modifications could be made on the control unit to compensate for deadband, some means of predicting their effect had to be established. Since the system contained non-linearities (the deadband and multiplier) and a digital section in the control unit, no attempt was made to study the system analytically. Instead, it was more feasible to study the system by modeling it on a digital computer and inserting models of candidate deadband compensators.

2.0 Description of Model

The block diagram of the guidance system, shown in Figure 5, was programmed on an interactive, time-sharing, computer terminal.

2.0.1 Ship Hardware

The ship hardware transfer function describes the receiver and rudder-drive system. Figure 6 shows a block diagram of the ship's hardware. Since the receiver has a unity transfer function, it is not shown in this figure and will be left out of the discussion hereafter.

The rudder-drive system consists of a motor-drive unit with a position feedback element which displaces the ship's rudder to the commanded value. The largest time constant of the rudder-drive system was 0.1 seconds. Since the time constant of the motor is

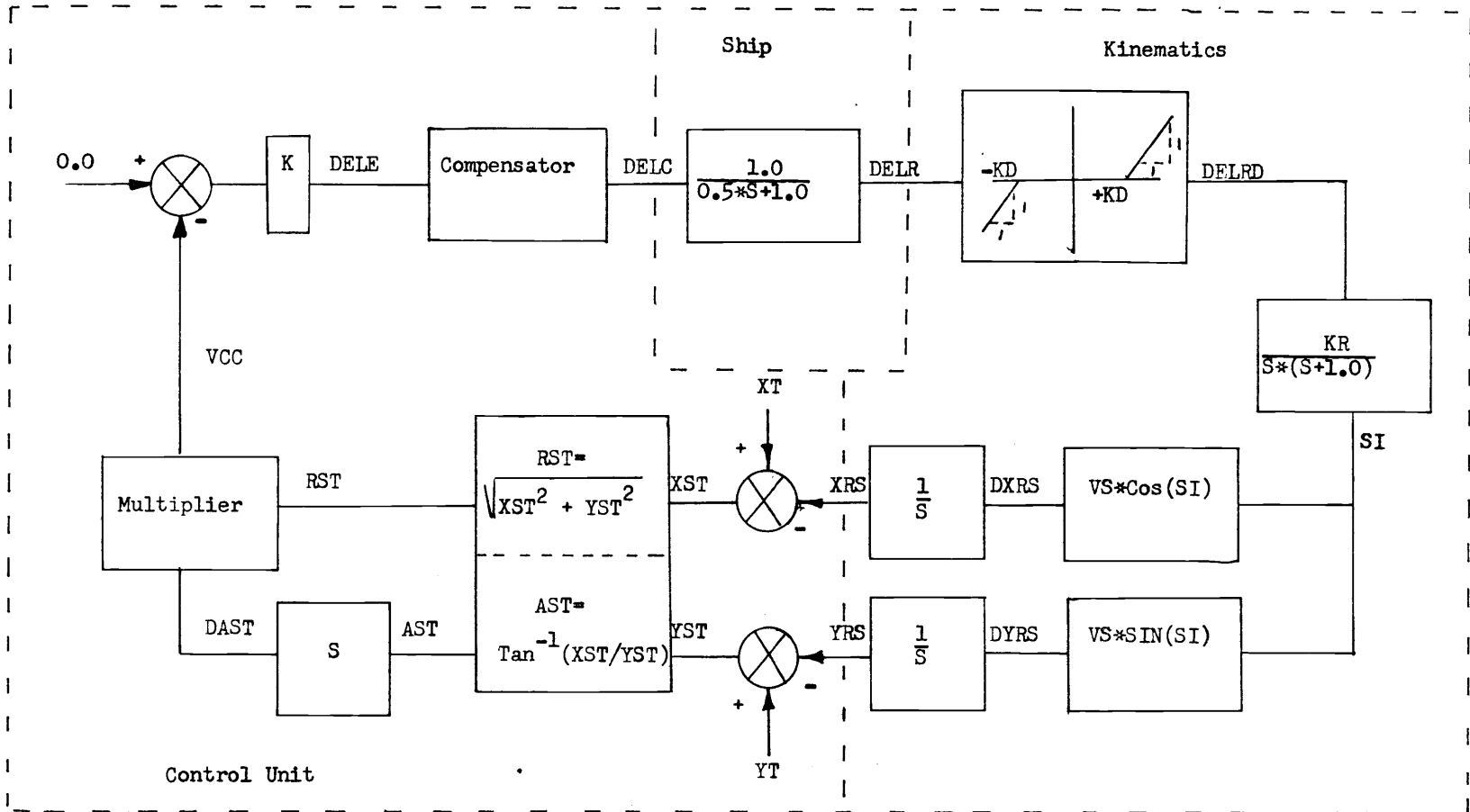


Figure 5. Guidance System Model Block Diagram

small compared to 0.5 seconds, only its integrating effect is included in the simulation.

2.0.2 Ship Kinematics

Due to the effects of the hull profile on steering, a deadband exists in the ship's turning-rate response to rudder displacements. The deadband was found to be symmetrical and exhibited the properties shown in Figure 7.

The gain between the turning rate of the ship and rudder displacements was measured to be 1.33 degrees/second/degree. This value was measured when the ship was cruising at a speed of 40 feet/second and rudder deflections were less than 20 degrees. In making the model of the system, it was assumed that this value of gain would remain constant through the entire range of rudder angles.

2.0.3 Ground Control Unit

The time response of the hardware in the control unit was monitored. All time constants were found to be an order of magnitude smaller than the time constant associated with the ship's inertia and for this reason they were left out of the simulation model. Only the control laws of the control unit were included in the model.

Using Figure 5, the entire guidance system was programmed on an interactive time-sharing, computer terminal system in the

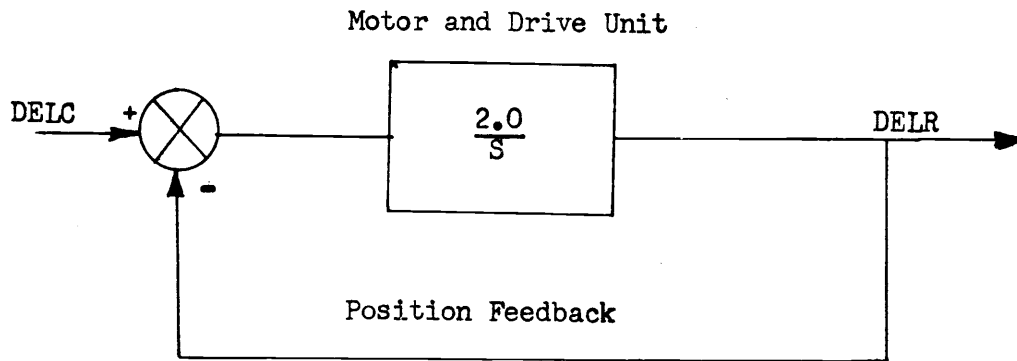


Figure 6. Ship Hardware Model Block Diagram

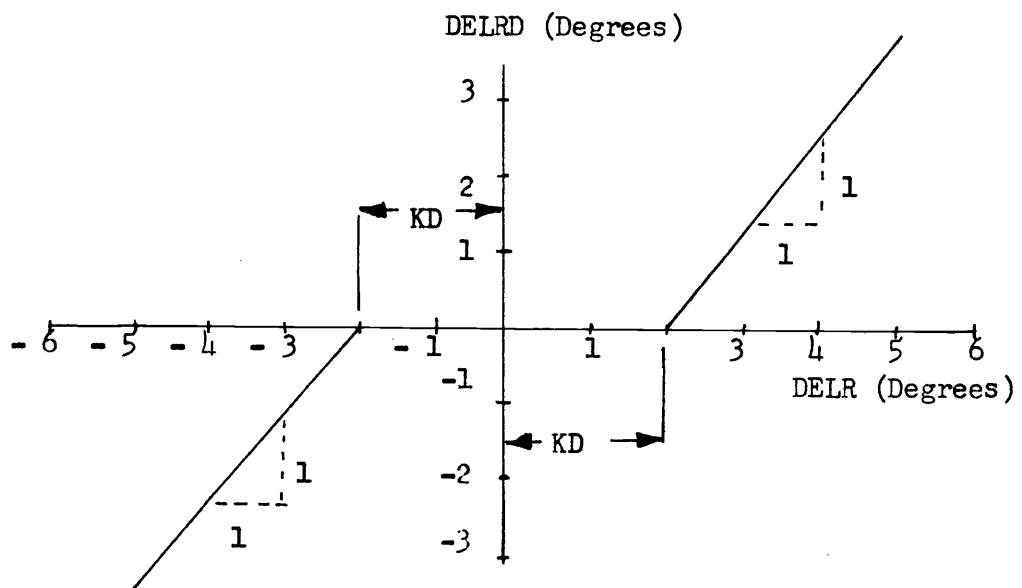


Figure 7. Deadband Model

Fortran IV language. The trapezoidal rule for integration (3) was used with a time interval of 0.1 second. This time interval was less than one-fifth the smallest time constant of the model. Figure 8 is a listing of the computer simulation of the guidance system without dead-band compensation.

2.1 Verification of Model

Ground track plots and cross course velocity versus time charts were generated for various initial headings of the ship. These results correlated closely with the field runs and it was concluded that an accurate simulation model had been developed.

Figures 9 and 10 are an example of the data recorded from the model. The ship was initially heading north with the target at 10,000 feet in range at an azimuth angle of 45 degrees. As can be seen from the ground track plot, the ship missed the target by 450 feet. This was due to the deadband which allowed the ship to develop an uncorrected error in its heading as shown in the plot of cross course velocity versus time.

The initial value of the ship's cross course velocity was approximately 28 feet/second. This resulted in a rudder command which was outside of the deadband region, and the ship changed its heading rapidly. Cross course velocities less than KD/K caused the rudder to operate in the deadband region. For the values of K and KD used,

Figure 8. Listing of Computer Simulation of Guidance System without Deadband Compensation

```

1 OPEN(2,"DELE",OUTPUT,SYMBOLIC)
2 OPEN(3,"VCC",OUTPUT,SYMBOLIC)
3 OPEN(4,"XY",OUTPUT,SYMBOLIC)
4 DIMENSION TIP(200,3)
5 REAL KD,K,KR
6 DISPLAY"WITHOUT COMPENSATION"
7 DISPLAY" "
8 DISPLAY" "
9 DISPLAY" "
10 SI=0.
11 XT=7000.
12 YT=7000.
13 XRS=0.
14 YRS=0.
15 VS=40.
16 DISPLAY"SI =",SI,"      XT =",XT,"      YT =",YT,"      VS="
,VS,"      KD =",KD
17 DISPLAY" "
18 DISPLAY" "
19 KD=10.
20 KR=VS/30.
21 K=1.
22 DT=.1
23 DSI=0.
24 DELR=0.
25 XST=XT-XRS
26 YST=YT-YRS
27 RST=SQRT[XST**2+YST**2]
28 AST=57.296*ATAN(YST/XST)
29 VCC=+VS*SIN[(SI-AST)/57.296]
30 ASTCLD=AST
31 KIP=0
32 DISPLAY"      TIME(SECONDS)  XRS(FEET)      YRS(FEET)      VCC(FEET/SECOND)"
33 DISPLAY" "
34 TIME=0.
35 WRITE(1,201)TIME,XRS,YRS,VCC
36 DC 500 I500=1,100
37 KIP=KIP+1
38 DC 400 I400=1,30

```

```

39 TIME=TIME+DT
40 VB=-VCC
41 DELE=K*VR
400 DELC=DELE
700 DDELR=2.*(DELC-DELR)
701 DELR=DELR+DDELR*DT
702 DELRD=0.
703 IF(DELR.GT.KD)DELRD=DELR-KD
704 IF(DELR.LT.-KD)DELRD=DELR+KD
705 DDSI=KR*DELRD-DSI
706 DSI=DSI+DDSI*DT
707 SI=SI+DSI*DT
708 DXRS=VS*COS(SI/57.296)
709 DYRS=VS*SIN(SI/57.296)
710 XRS=XRS+DXRS*DT
711 YRS=YRS+DYRS*DT
712 XST=XT-XRS
713 YST=YT-YRS
714 RST=SQRT(XST**2+YST**2)
715 AST=57.296*ATAN(YST/XST)
716 DAST=(AST-ASTOLD)/DT
717 ASTOLD=AST
718 VCC=-RST*DAST/57.296
719 IF(RST.LT.500.) GO TO 600
720 400 CONTINUE
721 WRITE(3,200) TIME,VCC
722 WRITE(4,200) YRS,XRS
723 TIP(KIP,1)=TIME
724 TIP(KIP,2)=DELE
725 TIP(KIP,3)=DELC
726 IF((KIP/2.-AINT(KIP/2.)).EQ.0.) GO TO 500
727 WRITE(1,201)TIME,XRS,YRS,VCC
728 500 CONTINUE
729 600 WRITE(1,201)TIME,XRS,YRS,VCC
730 DC 300 L=1,KIP
731 300 WRITE(2,200)TIP(L,1),TIP(L,2)
732 200 FORMAT(E12.3,$,$,F12.3,$,$)
733 DISPLAY" "
734 DISPLAY" "
735 DISPLAY" RST =",RST
736 201 FORMAT(4(4X,F10.4))
737 ONE=1.E37
738 WRITE(2,200)ONE,ONE
739 DC 301 L=1,KIP
740 301 WRITE(2,200)TIP(L,1),TIP(L,3)
741 END

```

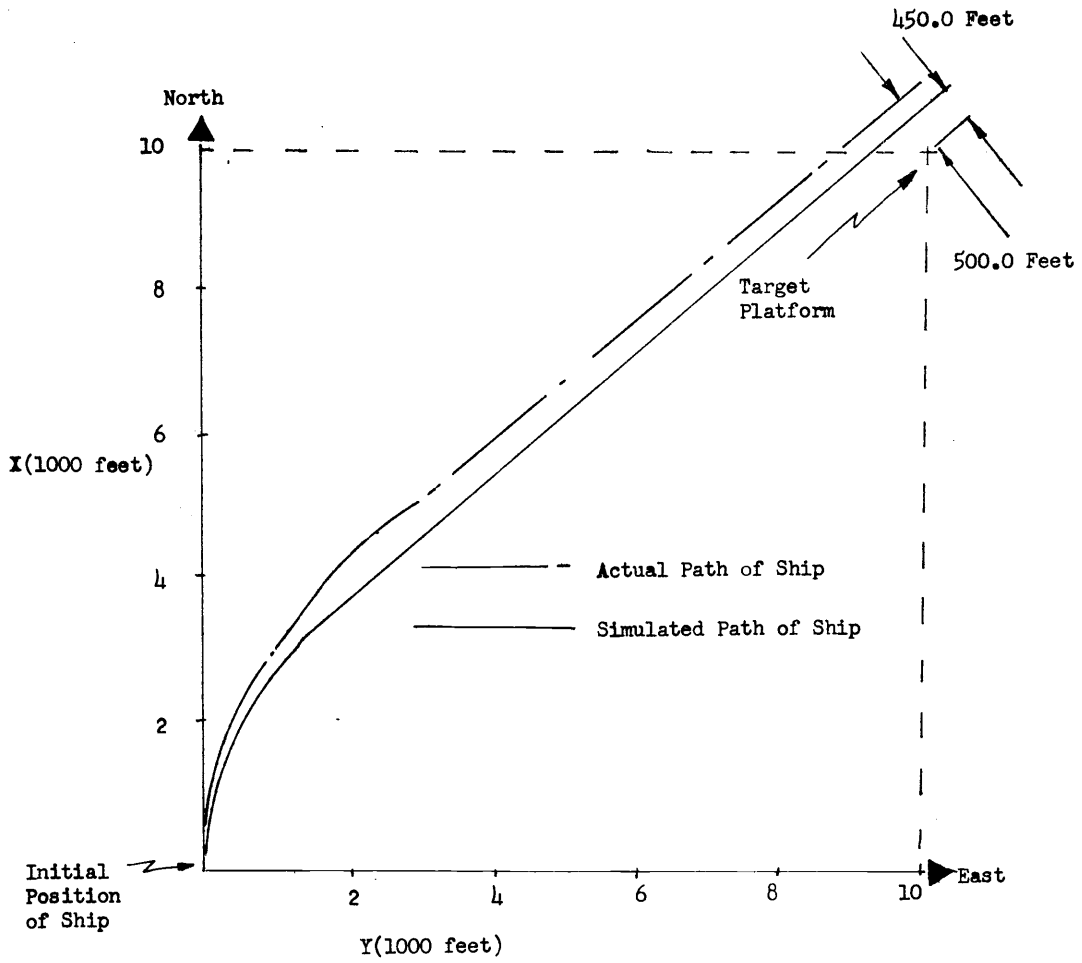


Figure 9. Simulated and Actual Path of Ship without Deadband Compensation

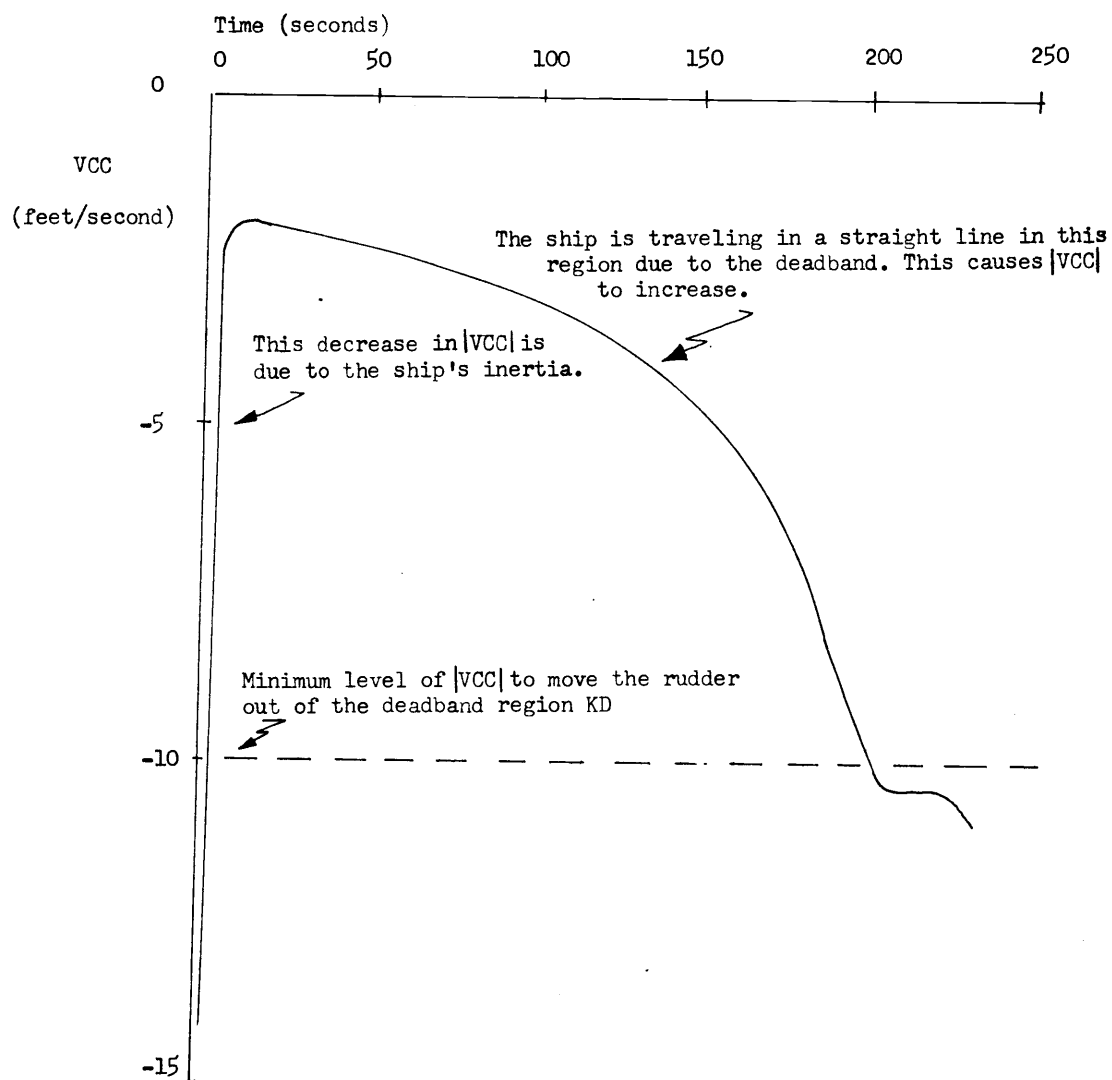


Figure 10. Simulated Cross Course Velocity as a Function of Time for the Guidance System without Deadband Compensation

this value of cross course velocity was ten feet/second. Thus, in Figure 10, only the ship's inertia in the water caused it to diminish its cross course velocity to less than ten feet/second.

After the ship's heading was no longer changing, an increase in its cross course velocity occurred, for the ship was traveling in a straight line which did not intersect the target. If no guidance were present, the cross course velocity would build up to the ship's velocity as it passed the target. However, when the cross course velocity became greater than ten feet/second, the ship again responded to rudder commands and the guidance system, control-loop error was held at roughly ten feet/second as seen in Figure 10.

III. DEADBAND COMPENSATING TECHNIQUES

Three conventional deadband compensating techniques were considered. In each case they were inserted between DELE and DELA of the model. Figure 11 shows the compensators tested.

3.0 Introducing Symmetrical Bias

By introducing a symmetrical bias it was hoped that the effect of the deadband would be eliminated. Figure 12 displays the compensator, the ship hardware transfer function, and the deadband. From a steady-state standpoint, the deadband would not introduce a bias in its output if the bias of the compensator was equal to the deadband distance KD .

As can be seen from Figure 12, if KD equaled five degrees, a ten degree rudder command step function is generated whenever the ship's cross course velocity changes sign. Thus, from a closed loop transient standpoint, this technique failed because it caused undue wear on the ship's rudder-drive hardware.

3.1 Dithering the System

The dither technique operates on the principle that the time average of the output signal of the deadband equals that of its input. This is shown in Figure 13. The main objection to this technique is

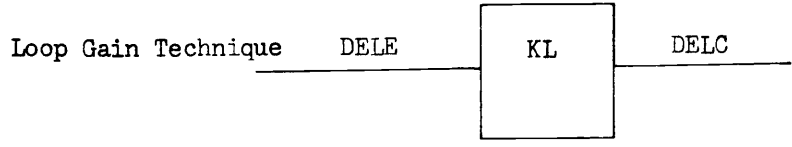
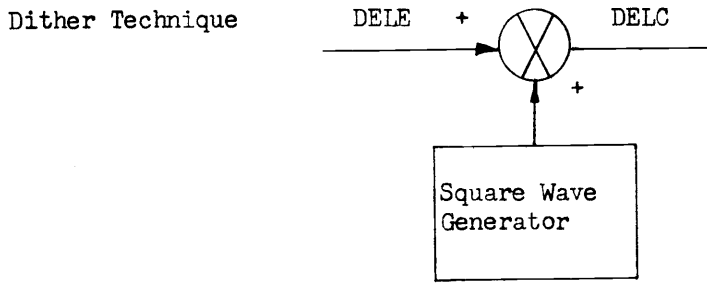
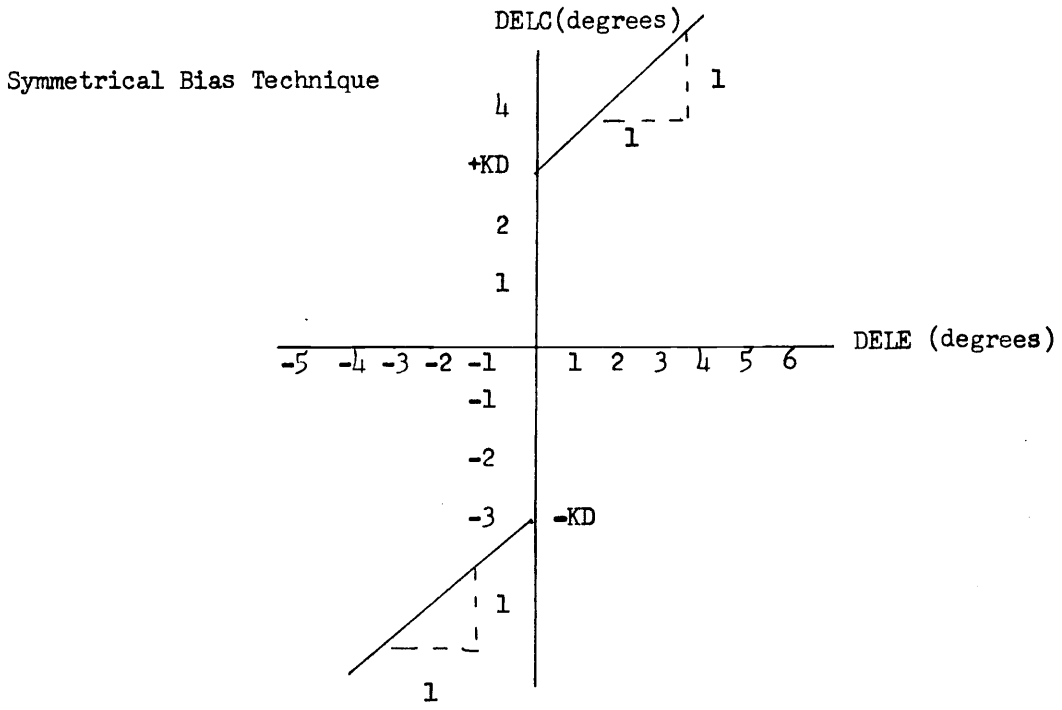


Figure 11. Three Types of Deadband Compensators

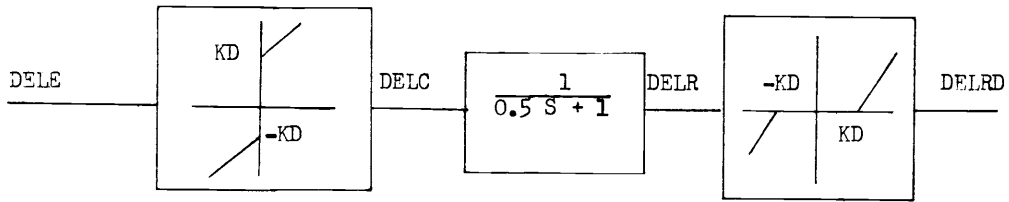


Figure 12. Symmetrical Bias Deadband Compensator

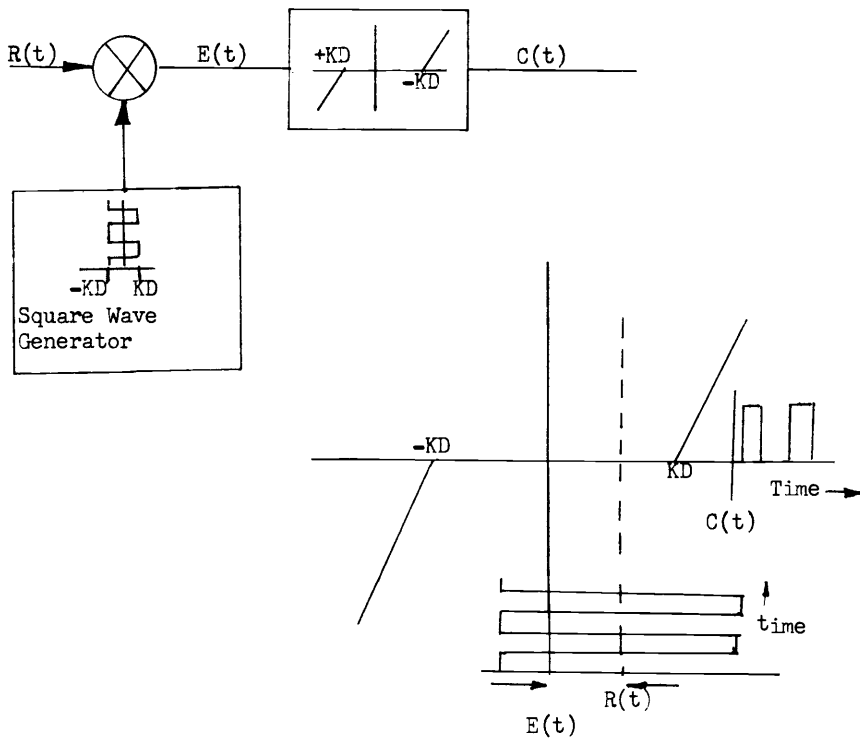


Figure 13. Dither Deadband Compensator

that it also required the ship's rudder to move rapidly. From a reliability standpoint, this movement was detrimental since it caused undue wear on the system.

3.2 Increasing System Open Loop Gain

The third technique, increasing the open loop gain, had the effect of diminishing the deadband as shown in the simplified model of the system in Figure 14. Diagram A in Figure 14 is a closed loop system of the same type used on the ship's guidance system. When the output signal was greater than KD/K , the system operated outside of the deadband region and the deadband was equivalent to the subtractor as shown in diagram B of Figure 14. This subtractor was brought back to the input by dividing it by K as shown in diagram C of Figure 14. If $R(t)$ is a step input of amplitude R , the steady-state value of the output can be written as $C(\infty) = R - KD/K$. Thus, by increasing the loop gain K , the biasing effect of the deadband KD was reduced.

This was unsuccessful, however, for by increasing loop gain the system became unstable. Thus the deadband had to be compensated by an unconventional technique.

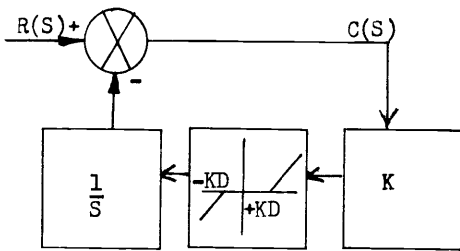


Diagram A

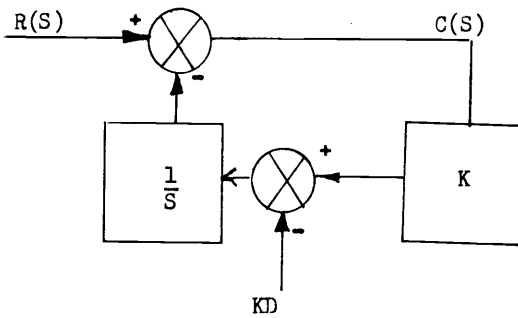


Diagram B

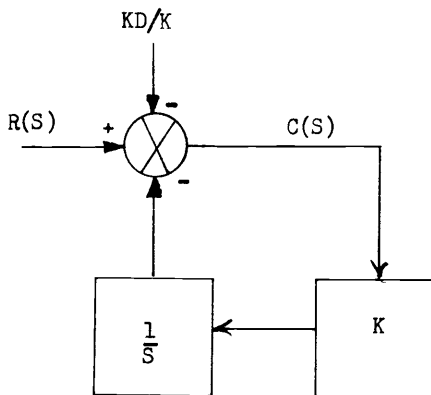


Diagram B

If $R(s)$ is a unit step function the final value of the output is:

$$C(\infty) = 1 - KD/K$$

Thus by increasing the open loop gain K the effect of KD is diminished.

Figure 14. The Effect of Increasing Open Loop Gain on Deadband

IV. ADAPTIVE DEADBAND COMPENSATOR

The fourth technique was to incorporate an adaptive deadband compensator. The compensator contained digital integrated circuits to eliminate the error caused by the deadband. Appendix I gives a wiring diagram of the compensator. Since the preceding deadband compensators had undesirable characteristics, this compensator had to be carefully designed and tested by modeling it on a digital computer.

4.0 Description of Adaptive Deadband Compensator

The main function of the adaptive deadband compensator was to add a synthetic rudder command to the rudder deflection calculated by the control unit whenever the absolute value of the ship's cross-course velocity remained constant or decreased. This is shown graphically in Figure 15 by plotting the adaptive compensator's input and output signals as a function of time. Note that there is a region of input values, below which the synthetic rudder command goes to zero, independent of the derivative of the input. This region prevents the compensator from operating on signals where the signal-to-noise ratio becomes significant.

The adaptive deadband compensator derives its ability to reduce the detrimental effects of the deadband from the fact that it is operating in a closed-loop, guidance system. When the ship's rudder is

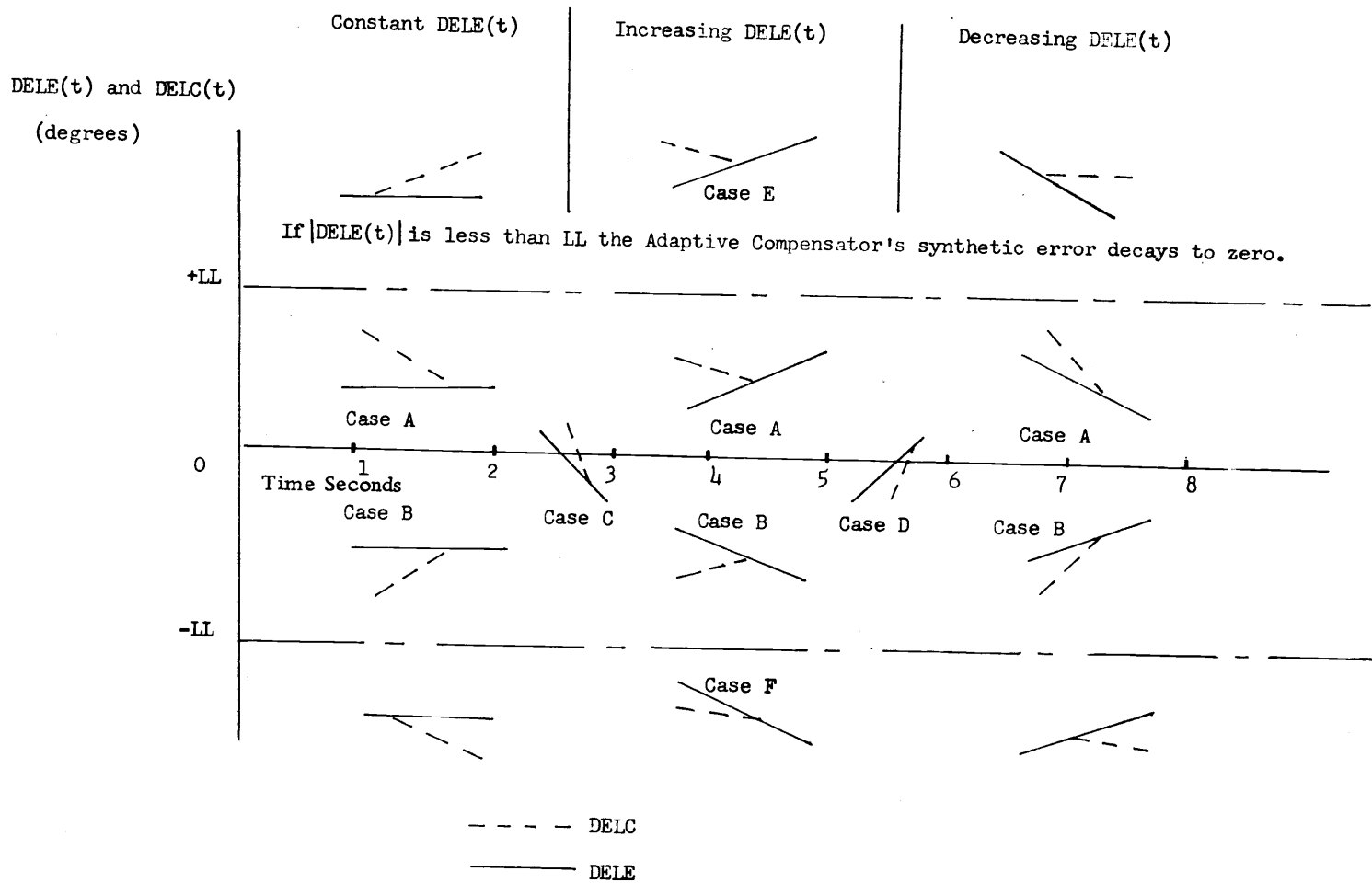


Figure 15. Adaptive Deadband Compensator's Characteristics

moved beyond the deadband region, the ship changes its heading and decreases its cross-course velocity. The compensator monitors this decrease in cross-course velocity and tries to drive VCC as small as possible by increasing the synthetic rudder command. When the cross-course velocity is within a region where its signal-to-noise ratio becomes significant, no further decrease in cross-course velocity is attempted and the compensator causes its synthetic rudder command to go to zero. At this point, the compensator waits for the cross-course velocity to go out of the threshold region, and it again generates a synthetic rudder command which increases until its input signal is again forced back to the threshold region.

4.1 Computer Model of Compensator

The model of the compensator was programmed on two levels including the "wiring of the compensator" and the "integrated circuits" used in the compensator. By subdividing the model in this manner, it could more closely model the actual adaptive compensator, for it too was broken down in these two levels.

The model of the integrated circuit was written as a subroutine package. This was used by the main program called the Wiring of the Compensator Program listed in Appendix II. Subroutines were written which simulated all IC's used in the compensator. These are listed in Appendix III and include a: JK flipflop, D type flipflop,

Bistable latch, NOR gate, NAND gate, four-input adder, and two-input adder.

As a test of the adaptive compensator, signals were fed into the model and its response was plotted. In all cases the compensator model operated as predicted. Figure 16 shows a sample run made where the input to the compensator decreased slowly and then fell rapidly. In this case, the adaptive compensator's ability to handle a zero crossover smoothly and a threshold cutoff and turn on was tested.

4.2 Adaptive Compensator Results of Ship Guidance System Model

The model of the compensator was inserted in the ship guidance model to evaluate its effect on the problem of deadband. Computer runs were made for various initial headings and ranges of the ship with respect to the target. Plots were made of the ship's path and the compensator's input and output signals as a function of time. Also the ship's miss distance from the target was printed for each run. In all cases the ship's miss distance decreased an order of magnitude from the results without the adaptive compensator.

Figures 17 and 18 show an example of these results where the target was initially 10,000 feet and at an angle of 45 degrees from the ship. As can be seen from Figure 17, the ship came within 100 feet of the target. Figure 18 is a plot of the compensator's input and output signals. Notice that the input signal was forced toward zero as a synthetic rudder command was added to its output.

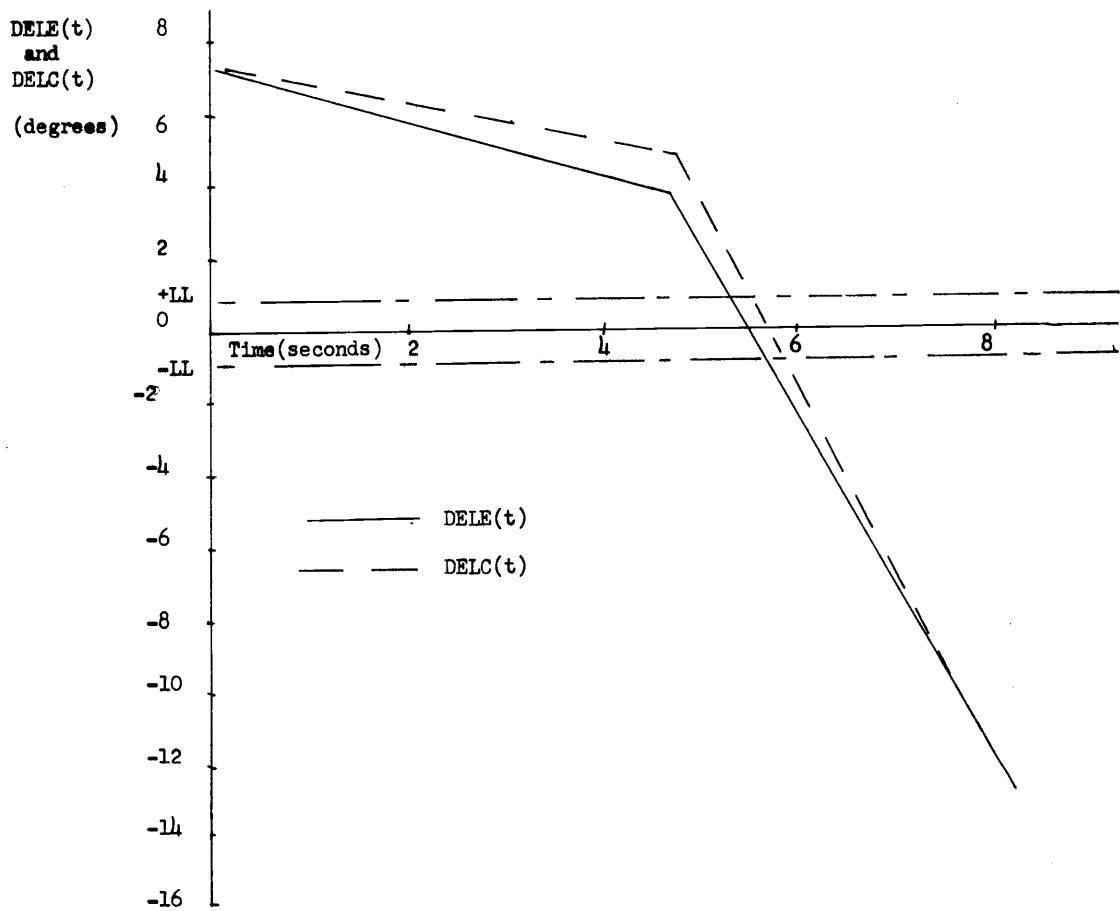


Figure 16. Adaptive Compensator with Decaying Input

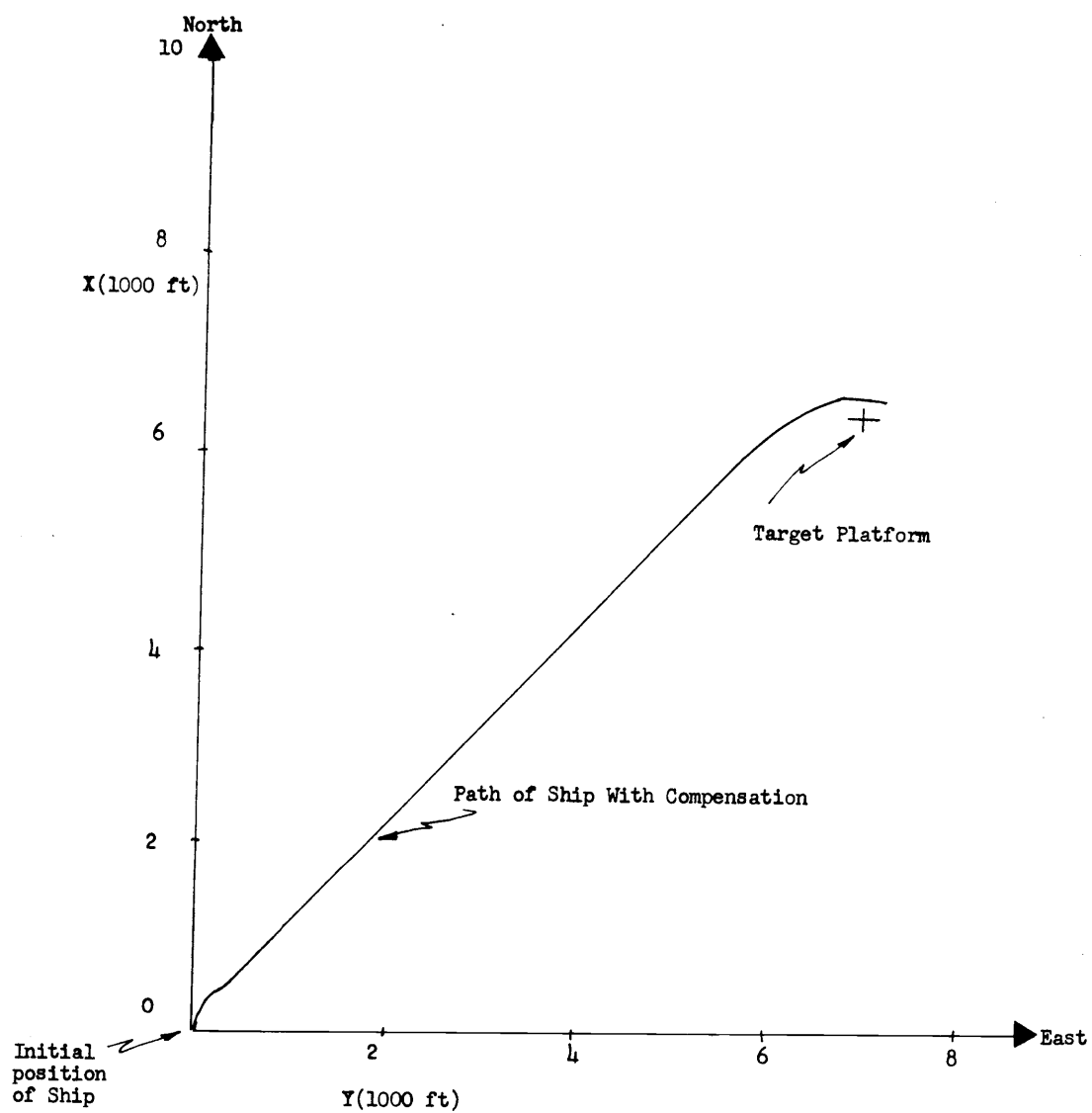


Figure 17. Simulated Path of Ship with Adaptive Deadband Compensation

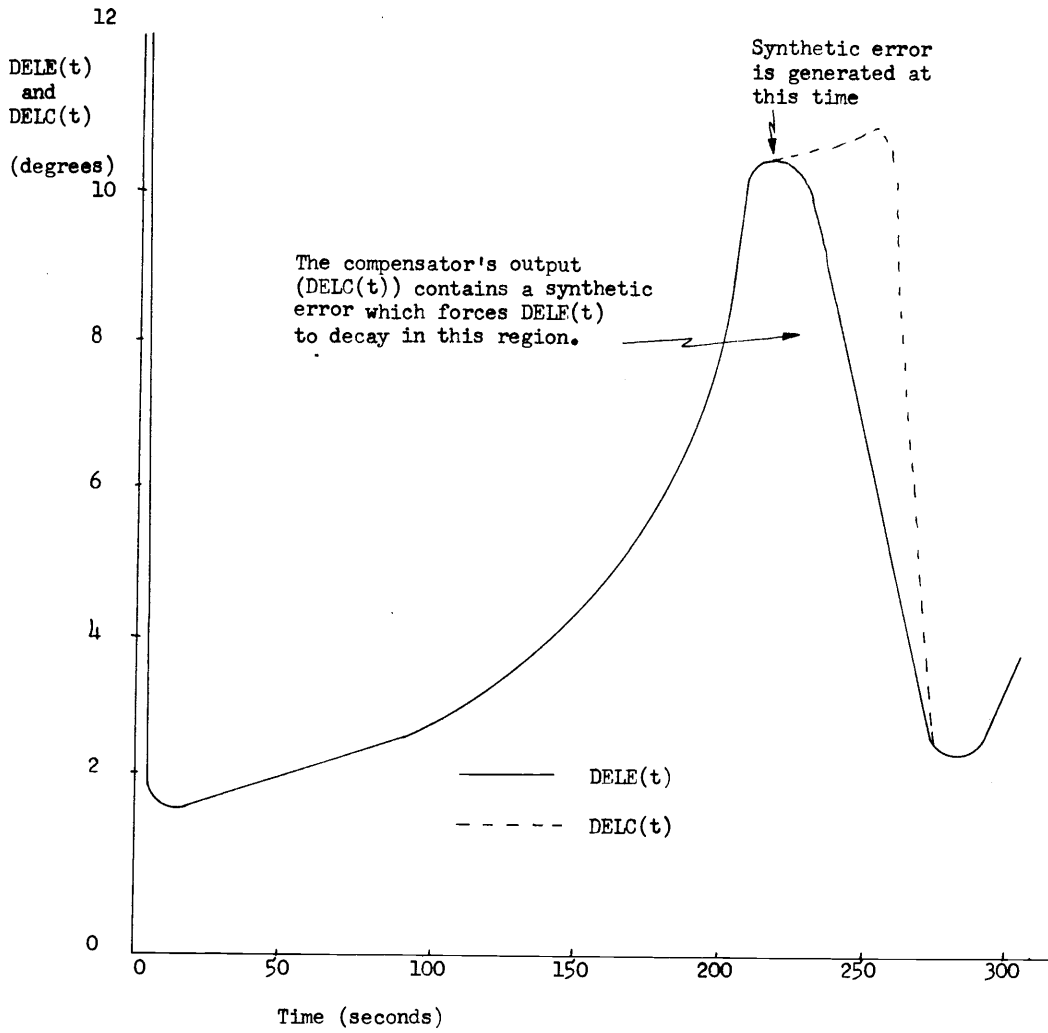


Figure 18. Adaptive Compensator's Input and Output Signals as a Function of Time for a Simulated Run

V. CONCLUSION

Only the system with the adaptive compensator gave the desired results. The other three compensators tested either caused the ship's guidance loop to become unstable or produced undue wear on the system.

The adaptive compensator technique could be employed in other closed loop systems where the effects of deadband are detrimental to their performance. While the other techniques investigated are less expensive to implement, they have characteristics which can diminish the reliability of the system.

BIBLIOGRAPHY

1. Motorola Semiconductor Products Incorporated. Motorola Digital Integrated Circuits, MRTL MC900/MC800 Series. Phoenix, Arizona, July 1967. v.p.
2. Motorola Semiconductor Products Incorporated. The Integrated Circuit Data Book, Supplement 1. Phoenix, Arizona, 1968. v.p.
3. Southworth, R. W. Digital Computations and Numerical Methods. New York, McGraw-Hill, 1965. 508 p.
4. Texas Instruments Incorporated. TTL Integrated Circuits From Texas Instruments. Houston, Texas, 1968. 47 p.

APPENDICES

APPENDIX I

ADAPTIVE COMPENSATOR WIRING DIAGRAM

The adaptive deadband compensator processes rudder commands calculated by the control unit into compensated commands which contain a synthetic signal capable of driving the rudder beyond the deadband region. This synthetic signal is an increasing or decreasing ramp which can either be added to or subtracted from the calculated commands as a function of both the magnitude and derivative of these commands.

A block diagram of the compensator is shown in Figure I-1. It contains six boxes which are composed of integrated circuits capable of processing parallel binary signals. Specifications and wiring diagrams for the integrated circuits used can be found in manuals written by Motorola Semiconductor Products Incorporated (1, 2) and Texas Instruments Incorporated (4). The function of the compensator is to process rudder commands calculated by the control unit ($DELE(t)$) into compensated commands ($DELC(t)$) which are synchronized with the control unit. The clock pulses designated as CP synchronize the compensator with the control unit and is used as a frequency base. This frequency base determines the rate at which the direction detector samples $DELE(t)$ and the rate at which the counter generates a ramp.

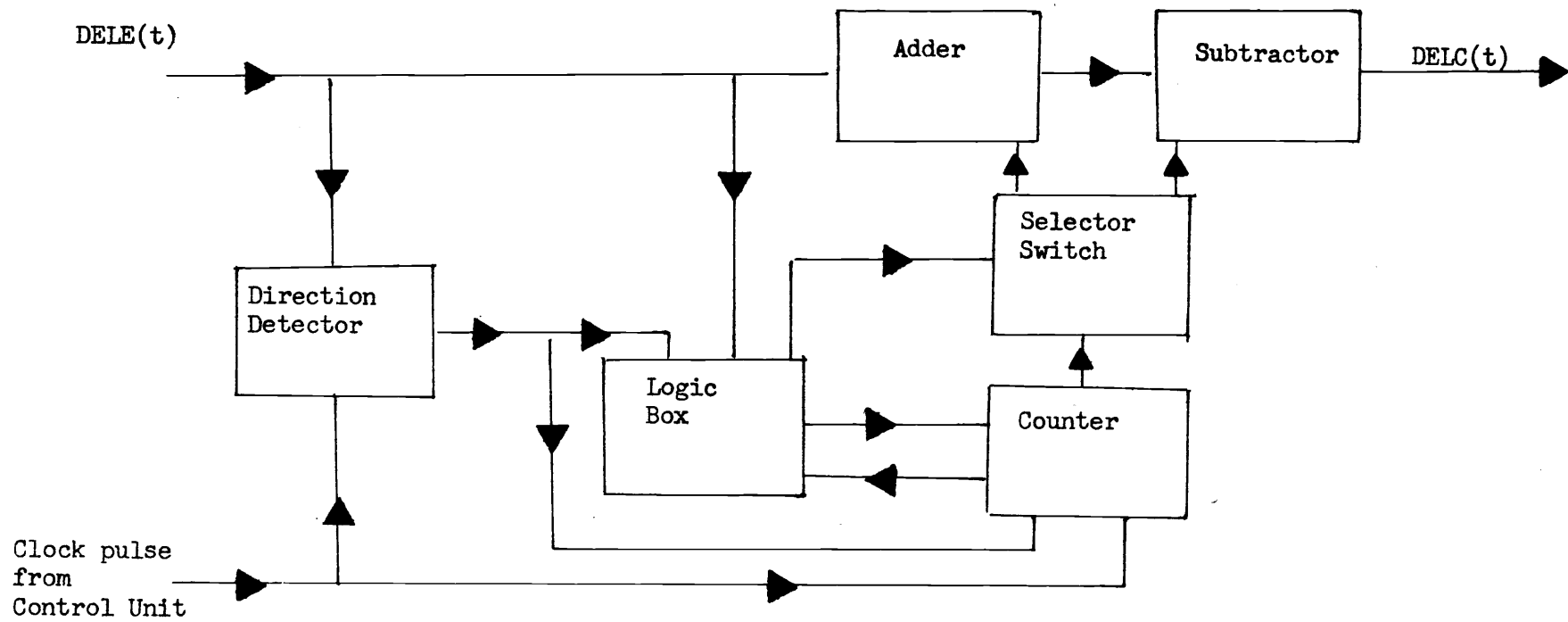


Figure I-1. Adaptive Compensator Block Diagram

Both the rudder command calculated by the control unit and the compensated rudder commands are nine bit parallel binary numbers and have the same scaling. A 30 degree starboard rudder command which causes the ship to turn toward its starboard is 111111111. A zero degree rudder command is 100000000 and a 30 degree port rudder command is 000000000.

The direction detector determines if the binary form of $DELE(t)$ is less than, equal to, or greater than the previous value sampled ($DELE(t-1)$). Based on this information, the magnitude of $DELE(t)$ and the output level of the counter, the logic box commands the counter to either count up or down one count for each clock pulse. The output of the counter is sent to the selector switch which routes it to either the adder or subtractor boxes upon command of the logic box. If the counter output is routed to the adder box it is added to $DELE(t)$ and if it is routed to the subtractor box it is subtracted from $DELE(t)$. Thus the compensator determines when and how to inject a ramp into the rudder command calculated by the control unit.

I.1 Direction Detector

The purpose of the direction detector is to determine if the binary form of the rudder command calculated by the control unit is less than, equal to, or greater than the previous value calculated. Ordinarily this can be done by subtracting its previous value

DELE(t-1) from the present value DELE(t) and detecting the sign and magnitude of the difference. Since integrated circuits which can subtract are not available, a carry-bit technique is employed which makes integrated circuits which add appear to be subtractors. The first step in this technique is to subtract the binary form of DELE(t-1) from $111111111 - \text{DELE}(t-1) + \text{DELE}(t)$. An increase in DELE(t) is detected by a plus one on the tenth bit called the carry-bit of the result of this step. The carry-bit is zero if DELE(t) has decreased or remained constant. If all nine of the bits are one in the second step, DELE(t) equals DELE(t-1) and no change occurs.

A diagram of the direction detector is shown in Figure I-2. DELE(t) consists of nine parallel bits designated as TWO8, TWO7, TWO6, TWO5, TWO4, TWO3, TWO2, TWO1, and TWO0 which designate its most significant bit (MSB) through its least significant bit (LSB) respectively. Bistable latches store the previous value of DELE(t) and output this previous value in its complemented form. This has the effect of calculating $111111111 - \text{DELE}(t-1)$. This is summed with DELE(t) by the parallel adders and the carry-bit is calculated. The carry-bit is designated as DIR and represents the direction of DELE(t). If INC is one, DELE(t) is increasing and if it is zero, DELE(t) is decreasing or remaining constant. To determine if DELE(t) is constant, a NAND gate is wired to the output terminals of the adders. When DELE(t) equals DELE(t-1), all of the adder

outputs are one and the output of the NAND gate becomes low. This output is called CHANGE. Thus, by monitoring the output of the NAND gate and the carry-bit of the adders, the direction of DELE(t) is determined.

I. 2 Logic Box

The logic box processes the rudder command calculated by the control unit and information from both the direction detector and the counter to control the position of the selector switch and the direction of the counter's ramp. Figure I-3 gives a wiring diagram for the logic box. Signals into the box are termed CHANGE, INC, ZER, TWO8, TWO7, TWO6, TWO5, and TWO4. The signals CHANGE and INC come from the direction detector and signify if the rudder command calculated by the control unit has increased, decreased, or remained constant. The signals TWO8 through TWO4 represent the five most significant bits of DELE(t). From these input signals the control unit generates two signals labeled ADD and DIRB.

The position of the selector switch is controlled by a signal derived by the logic box termed ADD. When ADD is plus one, the selector switch connects the counter to the adder and when it is zero, the counter is connected to the subtractor. Table I-1 is a truth table for the selector switch command ADD which was derived from Figure I-3. Note that when the signal ZER equals zero the synthetic rudder

DELE(t)

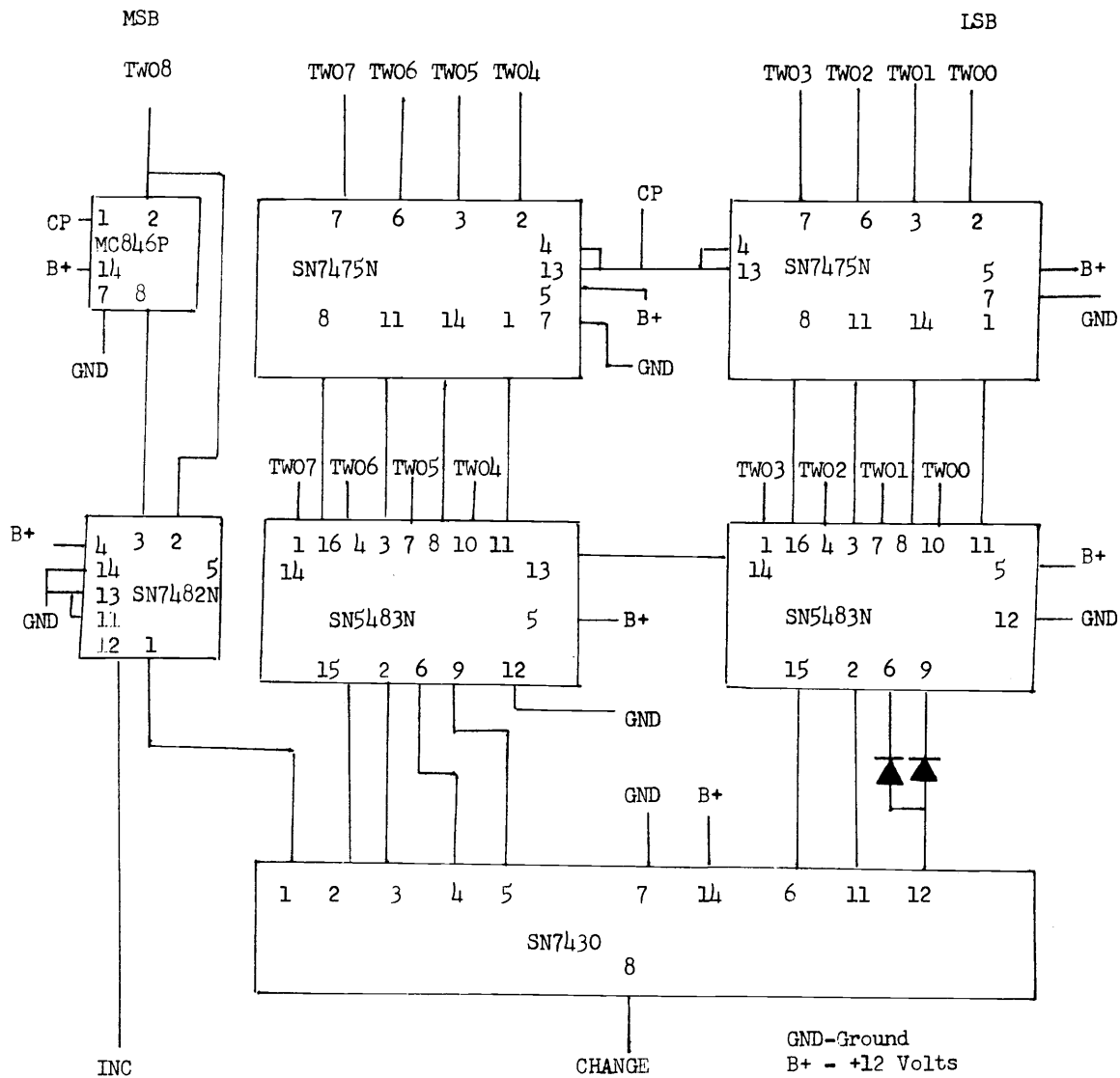


Figure I-2. Direction Detector Wiring Diagram

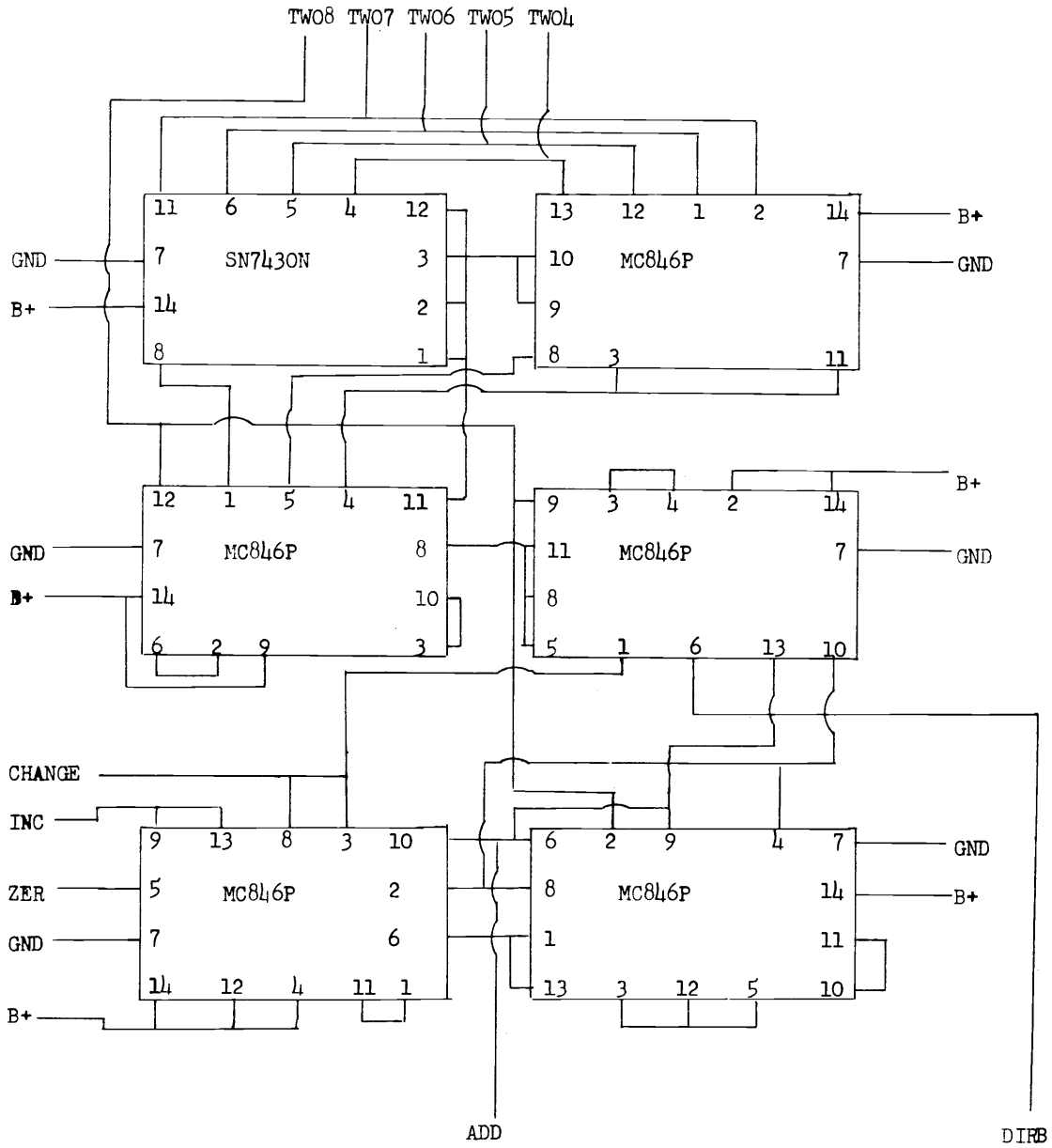


Figure I-3. Logic Box

Table I-1. Selector Switch Truth Table

TWO8	C5D7	ADD
<p>Most Significant Bit Of The Rudder Command Calculated By The Control Unit</p> <p>When TWO8 is a rudder command, one is calculated which causes the ship to turn to the right.</p>	<p>Zero Detection Signal Of Counter</p> <p>When the ramp generated by the counter is 0 degrees, C5D7 equals 0.</p>	<p>Command Sent To The Adder-Subtractor Switch From The Logic Box</p> <p>The ramp generated by the counter is added to the rudder command calculated by the control unit when ADD is 1.</p>
1	1	No Change
1	0	1
0	1	No Change
0	0	0

command is not equal to zero degrees and the signal ADD cannot change. This keeps the compensated rudder command from having a sharp jump whenever the rudder command calculated by the guidance system goes through zero degrees.

The counter command labeled DIRB causes the counter to increase or decrease its output according to the following Boolean equation which can also be written by inspection of Figure I-3.

$$\begin{aligned} \text{DIRB} = & \overline{(\overline{\text{TWO8}} + \overline{\text{TWO6}} + \overline{\text{TWO5}} + \overline{\text{TWO4}})} * (\overline{\text{TWO8}} + \overline{\text{TWO7}} + \overline{\text{TWO6}} + \\ & \overline{\text{TWO5}} + \overline{\text{TWO4}}) * (\overline{\text{ADD}} + \overline{\text{TWO8}}) * (\overline{\text{ADD}} + \overline{\text{TWO8}}) \\ & * (\overline{\text{CHANGE}} + \overline{\text{ADD}} * \overline{\text{INC}} + \overline{\text{ADD}} * \overline{\text{INC}}) \end{aligned}$$

Table I-2 is a truth table for the above Boolean equation. When DIRB is zero, the counter increases its output and when DIRB is one, the output decreases. The truth table lists all of the cases in which the counter increases its output. Each of these cases is labeled and can be found in Figure 15. All other cases not listed in detail in the truth table cause the counter to decrease its output until it reaches zero degrees. A synthetic rudder command is generated whenever DELE(t) is remaining constant or approaching zero degrees and is outside of the lower limits. As the synthetic rudder command increases, DELE(t) is forced to decrease until the signal-to-noise ratio of DELE(t) becomes significant. At this point, DELE(t) lies within the lower limit region as detected by the signals TWO8 through TWO4

Table I-2. Logic Box Truth Table

CHANGE	INC	ADD	TWO8	TWO7+TWO6+TWO5+TWO4	$\overline{\text{TWO7+TWO6+TWO5+TWO4}}$	DIRB	Graphical Representation Shown In Figure 15
1 or 0	1 or 0	1 or 0	1	0	1 or 0	1	A
1 or 0	1 or 0	1 or 0	0	1 or 0	0	1	B
1 or 0	1 or 0	1	0	1 or 0	1 or 0	1	C
1 or 0	1 or 0	0	1	1 or 0	1 or 0	1	D
1	1	1	1 or 0	1 or 0	1 or 0	1	E
1	0	0	1 or 0	1 or 0	1 or 0	1	F

For all other cases DIRB equals 0 causing the counter to increase its output ramp.

NOMENCLATURE

CHANGE equals 1 when DELE(t) does not equal DELE(t-1).

INC equals 1 when the binary form of DELE(t) is greater than DELE(t-1).

ADD equals 1 when the ramp generated by the counter is added to DELE(t).

DELE(t) is composed of bits TWO0 through TWO8. TWO0 is its least significant bit and TWO8 is its most significant bit.

DIRB is the signal generated in the logic box which causes the counter to either increase or decrease its ramp. When DIRB equals 1 the ramp generated by the counter decreases.

and the synthetic rudder command is forced to decrease to zero. In this manner, the effects of the deadband are diminished by the adaptive compensator.

I.3 Counter

The function of the counter is to generate a positive synthetic rudder command ramp which can either increase to +15 degrees when the signal labeled DIRB from the logic box is zero, or decrease to zero degrees when the signal is one. It sends this ramp to the selector switch and a signal labeled ZER back to the logic box. ZER is equal to one when the ramp is at zero.

Figure I-4 is a wiring diagram of the counter. The synthetic rudder command generated by the counter is designated by C5F1, C5F7, C5G1, C5G7, C5H1 and C5H7 which represent its least significant bit through its most significant bit respectively. Two parallel adders have positive feedback through three D type flipflops. This feedback is delayed by one clock pulse and is summed with a signal derived from the logic box termed DIRB which controls the direction of the synthetic rudder command ramp generated by the combination of DIRB and the feedback causes the counter to either increase or decrease its output by one count each clock pulse.

When DIRB is zero the counter increases its output by one count every clock pulse until it reaches 15 degrees. At this point, the

clock pulse signal to the flipflops becomes permanently high. This latches the counter's output to 15 degrees until DIRB changes back to plus one.

To make the counter decrease, the logic box sets DIRB equal to plus one. This causes the counter to decrease its output until it reaches zero degrees. The output will stay at zero degrees until DIRB changes to zero. A NAND gate is wired to the counter output and sends a signal labeled ZER to the logic box. This signal is plus one when the counter's output is not zero degrees and zero when the output is zero degrees.

I. 4 Selector Switch

The selector switch receives a ramp from the counter and sends this ramp to either the adder or subtractor box as dictated by the logic box.' It also has the function of sending a zero input command to the box not selected.

Figure I-5 is a wiring diagram of the selector switch. The signal labeled ADD represents the logic box's output which commands the selector switch to connect the counter to either the adder when it is one, or to the subtractor when it is zero. If the variable ADD is plus one, the subtractor's input, whose bits are labeled S0 through S5, is set equal to zero, and if ADD is zero the adder's input, whose bits are labeled A0 through A1, is set equal to zero. Thus, the

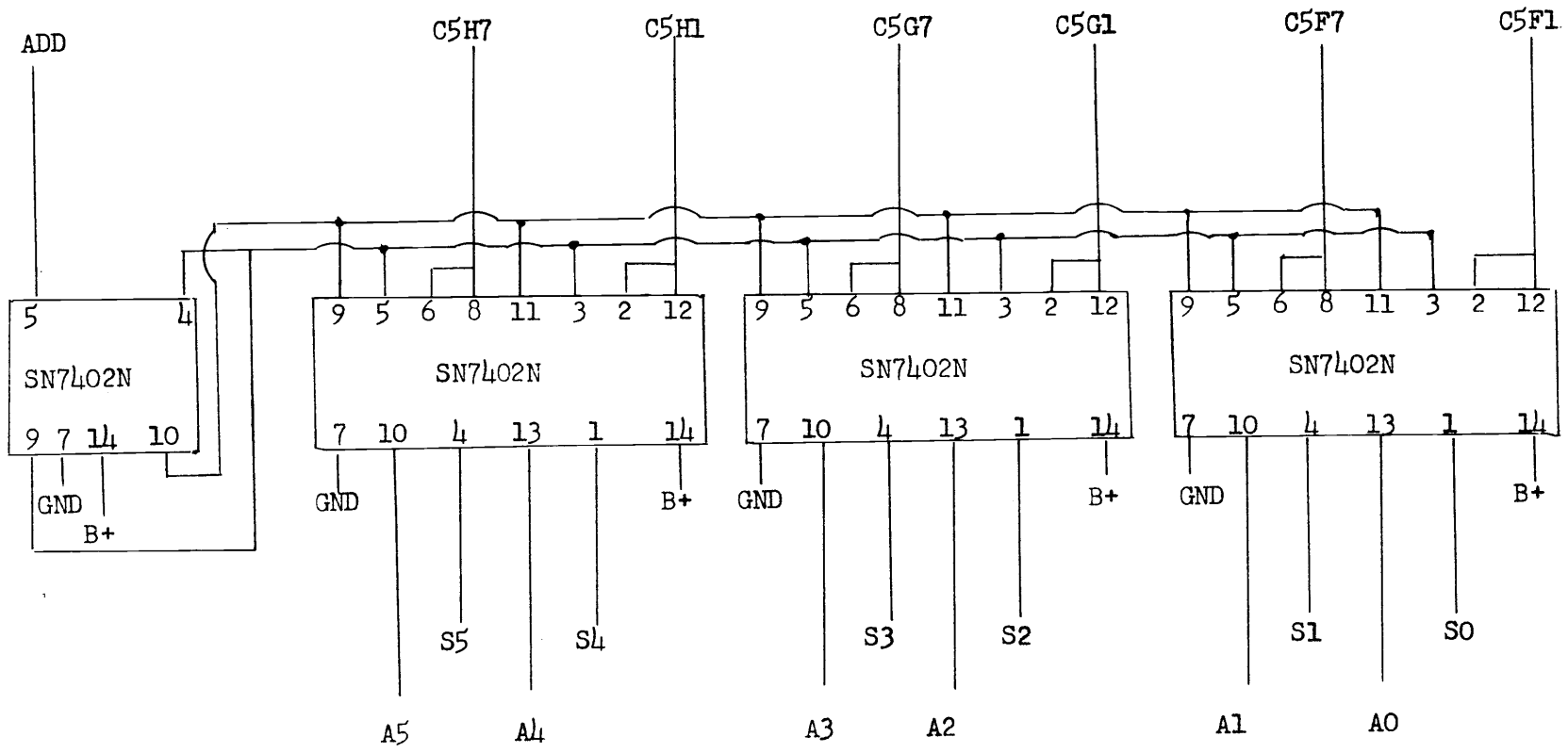


Figure I-5. Selector Switch Wiring Diagram

adder and the subtractor receive either a ramp from the counter or a zero generated by the selector switch.

I.5 Adder

The purpose of the adder is to add the output of the selector switch to the rudder command calculated by the control unit. The adder also complements and limits its output. The output is complemented so that both the adder and subtractor can be made identical as will be shown later.

Figure I-6 gives a wiring diagram of the adder. The two input signals to the adder are the nine bits of DELE(t) designated as TWO8 through TWO0 and the signal from the selector switch whose bits are labeled A0 through A5. The NAND gates have the dual function of inverting and limiting the maximum adder output to +30 degrees. The bits of the adder's output signal are designated as C2D5, C2D8, C2D14, C2D3, C2E5, C2E8, C2E14, C2E3 and C2F3 which represent its least significant bits through its most significant, respectively. If either the tenth or eleventh bit of the sum of DELE(t) and the selector switch output is one, a number larger than +30 degrees is being generated and the output of the NAND gates stays at +30 degrees. This keeps the output of the adder at the maximum possible nine bit number if a larger number is being generated.

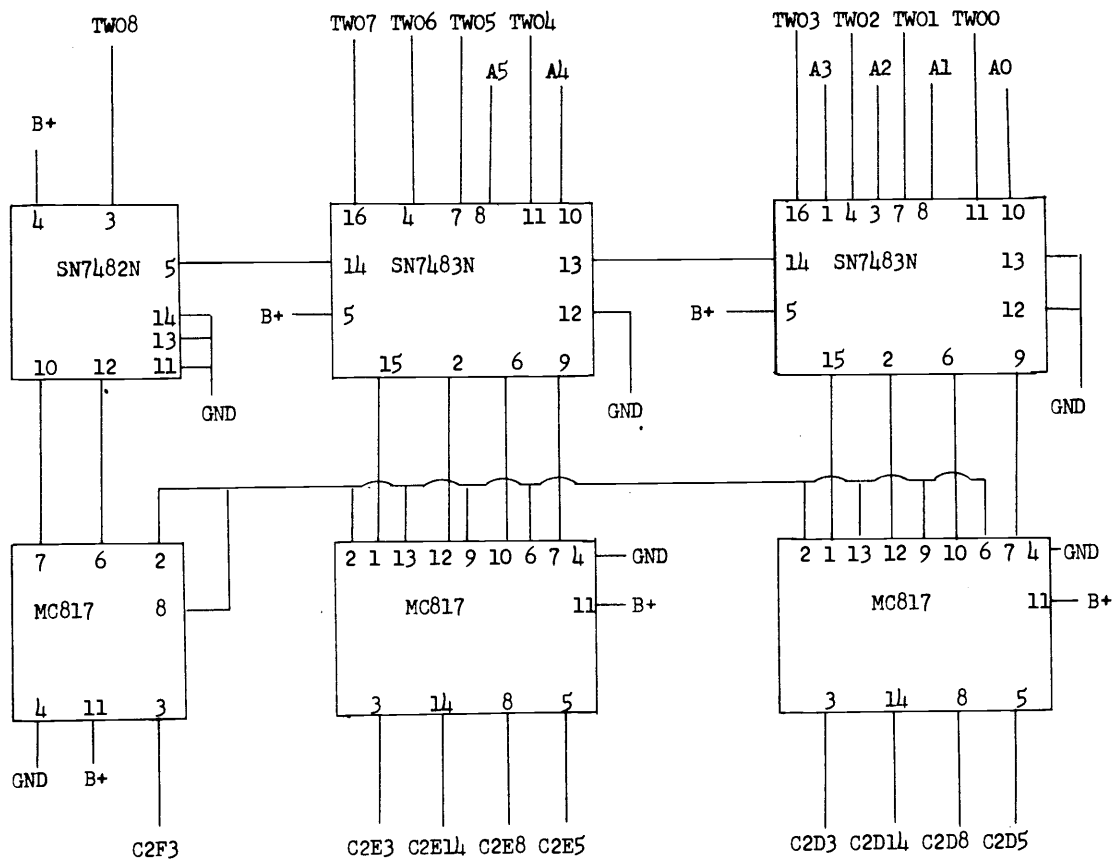


Figure I-6. Adder Wiring Diagram

I.6 Subtractor

The subtractor is identical to the adder and is located after it in the compensator. This is shown in Figure I-7 where A and S are inputs from the selector switch to the adder and subtractor boxes, respectively. Only one box is connected to the counter by the selector switch at a time and the opposite unit's input is set equal to 000000000. Thus, if the selector switch connects the adder to the counter, the subtractor's input, S, is set equal to 000000000.

Since both the adder and subtractor boxes complement their output, A is added to $DELE(t)$ and S is subtracted from it. When the adder is operating, its output is complemented twice: once by the adder and once by the subtractor. The net effect is that A is added to $DELE(t)$. When the subtractor is operating, S is added to the complement of $DELE(t)$ and this sum is complemented. This has the net effect of subtracting S from $DELE(t)$. Thus the process of addition and subtraction can be performed with two identical units connected in cascade.

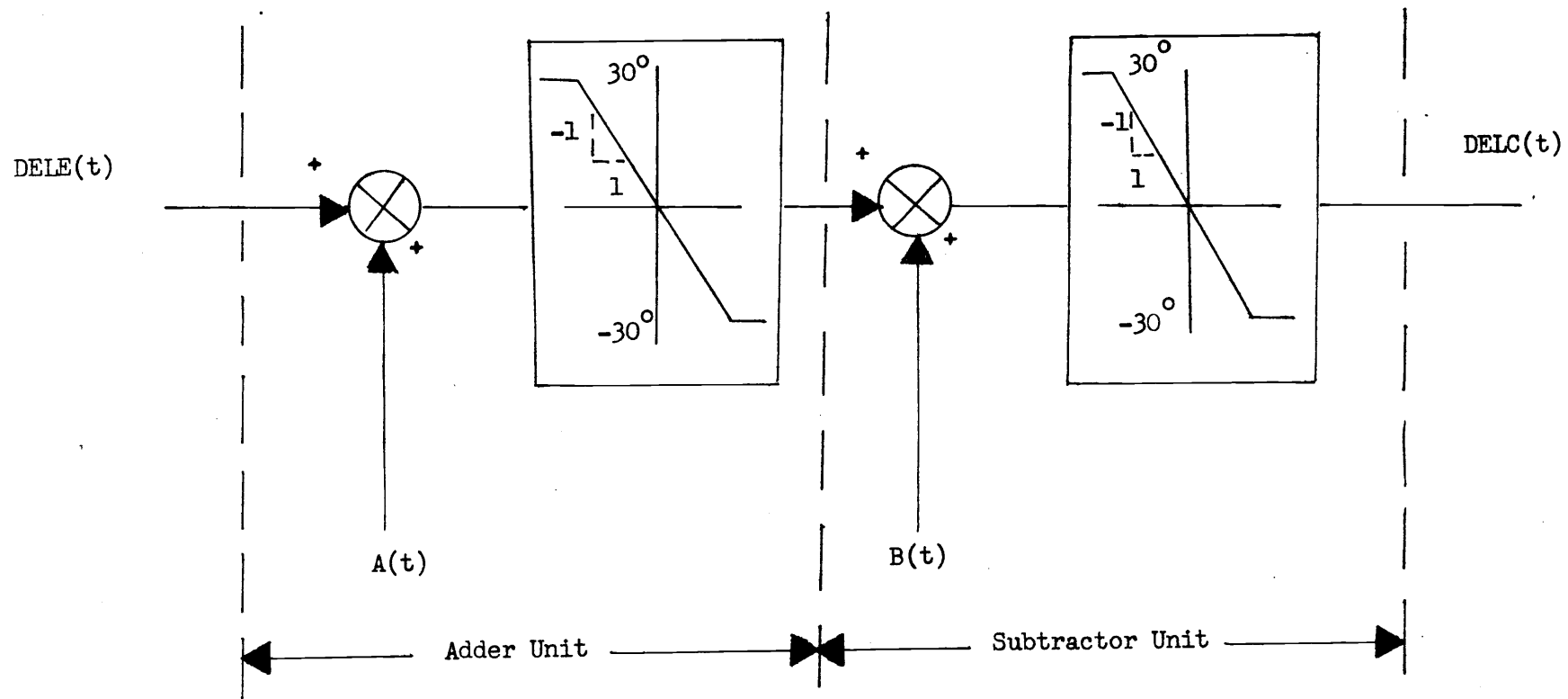


Figure I-7. Adder and Subtractor Block Diagram

APPENDIX II

GUIDANCE SYSTEM COMPUTER SIMULATION
LISTING WITH ADAPTIVE DEADBAND COMPENSATOR

The guidance system with the adaptive deadband compensator was simulated on an interactive computer. A listing of the simulation without subroutines is given in this Appendix. The statement numbers in the listing are indicative of their location in the program.

Figure II-1. Listing of Computer Simulation of Guidance System with Adaptive Deadband Compensator

```

1 OPEN(2,"DELE",OUTPUT,SYMBOLIC)
1.1 OPEN(3,"VCC",OUTPUT,SYMBOLIC)
1.2 OPEN(4,"XY",OUTPUT,SYMBOLIC)
1.3 DIMENSION TIP(200,3)
1.4 REAL KD,K,KR,INTCP,N,INC
1.5 DISPLAY"SYSTEM WITH COMPENSATION"
1.6 DISPLAY" "
1.7 DISPLAY" "
1.8 DISPLAY" "
1.9 SI=0.
2 XT=7000.
2.1 YT=7000.
2.2 XRS=0.
2.3 YRS=0.
2.4 VS=40.
2.5 KD=10.
2.6 DISPLAY"SI =",SI,"      XT =",XT,"      YT =",YT,"      VS =",VS,"KD =",KD
2.7 DISPLAY" "
2.8 DISPLAY" "
2.9 KR=VS/30.
3 K=1.
3.1 DT=.1
3.2 DSI=0.
3.3 DELR=0.
3.4 XST=XT-XRS
3.5 YST=YT-YRS
3.6 RST=SQRT[XST**2+YST**2]
3.7 AST=57.296*ATAN[YST/XST]
3.8 VCC=+VS*SIN[(SI-AST)/57.296]
3.9 ASTOLD=AST
4 KIP=0
4.1 T=0
4.2 TIME=0.
4.3 R=0.
4.4 P=1.

```


Figure II-1. (continued)

```

4.5 N=0.
4.6 CP=0.
4.7 CPCLD=ABS[CP-1]
4.8 C5D7=0.
4.9 DISPLAY"    TIME(SECOND)    XRS(FEET)    YRS(FEET)    VCC(FEET/SECOND)"
5 DISPLAY" "
5.1 WRITE(1,201)TIME,XRS,YRS,VCC
5.2 DC 500 I500=1,100
5.3 KIP=KIP+1
5.4 DC 400 I400=1,30
5.5 TIME=TIME+DT
5.6 VB=-VCC
5.7 DELE=K*VB
5.8 A1=AINT[ AINT[DELE*256./30.+256.]/2. ]
5.9 IF(DELE.GE.30)A1=255.
6 IF(DELE.LE.-30.)A1=0.
6.1 IF(RST.LT.1500.)GC TC 11
6.2 DELC=DELE
6.3 GC TC 12
6.4 11 CONTINUE
6.5 A2=AINT[A1/2. ]
6.6 A3=AINT[A2/2. ]
6.7 A4=AINT[A3/2. ]
6.8 A5=AINT[A4/2. ]
6.9 A6=AINT[A5/2. ]
7 A7=AINT[A6/2. ]
7.1 A8=AINT[A7/2. ]
7.2 A9=AINT[A8/2. ]
7.3 TWC0=AINT[DELE*256./30.+256]-A1*2.
7.4 IF(DELE.GE.30.) TWC0=1.
7.5 IF(DELE.LE.-30.) TWC0=0.
7.6 TWC1=A1-A2*2.
7.7 TWC2=A2-A3*2.
7.8 TWC3=A3-A4*2.
7.9 TWC4=A4-A5*2.
8 TWC5=A5-A6*2.
8.1 TWC6=A6-A7*2.
8.2 TWC7=A7-A8*2.
8.3 TWC8=A8-A9*2.
8.4 CP=ABS[CP-1]
8.5 DC 5 J=1,2
8.6 INTCP=ABS[CP-1].AND.CPCLD
8.7 CPCLD=CP

```

Figure II-1. (continued)

```

100 CALL RL7475I[INTCP,TWC0,Q1,C1B1,TWC1,Q2,C1B14,TWC2,Q3,C1B11
,TWC3,Q4,C1B8,CPO1B,T]
101 CALL RL7475I[INTCP,TWC4,Q1,C1C1,TWC5,Q2,C1C14,TWC6,Q3,C1C11
,TWC7,Q4,C1C8,CPO1C,T]
102 CALL D778I[INTCP,TWC8,R,Q1,C1D8,P,R,Q2,QB2,T,CPO1D,SQ1N1D,S
Q2N1D]
104 CALL AD7483I[N,C1B1,TWC0,C1F9 ,C1B14,TWC1,C1F6,C1B11,TWC2,
C1F2,C1B8,TWC3,C1F15,C1F14]
105 CALL AD7483I[C1F14,C1C1,TWC4,C1G9,C1C14,TWC5,C1G6,C1C11,TWC
6,C1G2,C1C8,TWC7,C1G15,C1G14]
106 CALL AD7482I[C1G14,TWC8,C1D8,C1H1,N,N,INC ,CNP4]
108 CALL ND7430I[C1F9,C1F6,C1F2,C1F15,C1G9,C1G6,C1G2,C1G15,C1H1
,CHANGE]
150 T=T+.5
151 5 CONTINUE
200 CALL ND846P[C5D7,P,C6A6,N,N,C2,N,N,C3,N,N,C4]
201 CALL ND846P[C6A6,TWC8,C6B3,N,N,C2,N,N,C3,N,N,C4]
202 CALL ND846P[C6B3,C6A6,C6B11,N,N,C2,N,N,C3,N,N,C4]
203 CALL ND846P[C6B11,P,C6B8,C6B3,P,C6B6,N,N,C3,N,N,C4]
204 CALL JK856P[CP,C6B8, C6B6,P,ADDB,ADD,N,N,Q2,QB2,T,CPO,SQ1N
6R,SQ2N6B]
205 CALL ND846P[ADDB,INC,C6A8,P,INC,C6A11,P,P,C3,P,P,C4]
206 CALL ND846P[C6A11,ADD,C6A3,N,N,C2,N,N,C3,N,N,C4]
207 A3A8CH=C6A3.AND.C6A8.AND.CHANGE
208 CALL ND846P[TWC8,P,C6E11,N,N,C2,N,N,C3,N,N,C4]
209 CALL ND7430I[C6E11,TWC7,TWC6,TWC5,TWC4 ,C6E11,C6E11,C6E11,C6E11,C6C8]
210 CALL NCR817I[TWC7,TWC6,C6D3,TWC5,C6E11,C6D14,TWC4 ,C6E11,C6D5,C6E11,C6E11,C6D8]
211 D3D14=C6D3.AND.C6D14
212 D5D8=C6D5.AND.C6D8
213 CALL ND846P[D3D14,D5D8,C6E6,N,N,C2,N,N,C3,N,N,C4]
214 CALL ND846P[C6C8,C6E6,C6E3,N,N,C2,N,N,C3,N,N,C4]
215 CALL ND846P[TWC8,ADDB,C6F8,C6E11,ADD,C6F11,C6E3,P,C6E8,A3A
8CH,P,C6F3]
216 F8F11E=C6F8.AND.C6F11.AND.C6E8
217 CALL ND846P[F8F11E,C6F3,DIRB,N,N,C2,N,N,C3,N,N,C4]
218 IF(T.LT.2.)DIRB=0.
300 CALL ND846P[CP,P,C5A3,DIRB,P,C5A6,P,P,C3,P,P,C4]
301 CALL JK856P[C5A3,P,P,C5A6,C5E6,C5E5,P,P,Q2,QB2,T,CPC5E,SQ1
N5E,SQ2N5E]
302 CALL ND846P[C5A3,C5E6,C5A9,P,P,C2,P,P,C3,P,P,C4]
303 CALL AD7483I[N,C5F14,P,C5B9,C5F8,DIRB,C5B6,C5G14,DIRB,C5B2,
C5G8,DIRB,C5H15,C5H14]
304 CALL AD7483I[C5B14,C5H14,DIRB,C5C9,C5H8,DIRB,C5C6,N,N,C5C2,
C5C2,C5A6,C5C15,C5C14]

```

305 CALL ND846P[C5A9,C5C15,C5A12,P,P,C2,P,P,C3,P,P,C4]
 306 CALL D778[C5A12,C5B9,R,C5F14,C5F1,C5B6,R,C5F8,C5F7,T,CP05F
 ,SQ1N5F,SQ2N5F]
 307 CALL D778[C5A12,C5B2,R,C5G14,C5G1,C5B15,R,C5G8,C5G7,T,CP05
 G,SQ1N5G,SQ2N5G]
 308 CALL D778[C5A12,C5C9,R,C5H14,C5H1,C5C6,R,C5H8,C5H7,T,CP05H
 ,SQ1N5H,SQ2N5H]
 309 CALL ND7430[C5F1,C5F7,C5G1,C5G7,C5H1,C5H7,P,P,P,C5D7]
 408 CALL NCR817[CADD,N,C4A3,N,N,C,N,N,C,N,N,C]
 416 CALL NCR817[C4A3,N,C4A8 ,N,N,C,N,N,C,N,N,C]
 424 CALL NCR817[C4A8,C5F1,C4B8,C4A3,C5F1,C4B14,C4A8,C5F7,C4B3,
 C4A3,C5F7,C4B5]
 432 CALL NCR817[C4A8,C5G1,C4C8,C5G1,C4A3,C4C14,C4A8,C5G7,C4C3,
 C4A3,C5G7,C4C5]
 440 CALL NCR817[C4A8,C5H1 ,C4D8,C4A3,C5H1,C4D14,C4A8,C5H7,C4D3
 ,C4A3,C5H7,C4D5]
 500 CALL AD7483[N,TWC0,C4B14,C2A9,TWC1,C4B5,C2A6,TWC2,C4C14,C2
 A2,TWC3,C4C5,C2A15,C2A14]
 501 CALL AD7483[C2A14,TWC4,C4D14,C2B9,TWC5,C4D5,C2B6,TWC6,N,C2
 B2,TWC7,N,C2B15,C2B14]
 502 CALL AD7482[C2B14,TWC8,N,C2C1,N,N,C2C12,C2C10]
 503 CALL NCR817[C2C10,C2C12,C2F5,N,N,C,N,N,C,N,N,C]
 504 CALL NCR817[C2F5,N,C2F8,N,N,C,N,N,C,N,N,C]
 505 CALL NCR817[N,C2F5,C2F14,C2C1,C2F8,C2F3,N,N,C,N,N,C]
 506 CALL NCR817[C2F8,C2B15,C2E3,C2F8,C2B2 ,C2E14,C2F8,C2B6,C2E
 8,C2F8,C2B9,C2E5]
 507 CALL NCR817[C2F14,C2A15,C2D3,C2F14,C2A2,C2D14,C2F14,C2A6,C
 2D8,C2F14,C2A9,C2D5]
 600 CALL AD7483[N,C2D5,C4B8,C3A9,C2D8,C4B3,C3A6,C2D14,C4C8,C3A
 2,C2D3,C4C3,C3A15,C3A14]
 601 CALL AD7483[C3A14,C2E5,C4D8,C3B9,C2E8,C4D3,C3B6,C2E14,N,C3
 B2,C2E3,N,C3B15,C3B14]
 602 CALL AD7482[C3B14,C2F3,N,C3C1 ,N,N,C3C12,C3C10]
 603 CALL NCR817[C3C10,C3C12,C3F5,N,N,C,N,N,C,N,N,C]
 604 CALL NCR817[C3F5,N,C3F8,N,N,C,N,N,C,N,N,C]
 605 CALL NCR817[N, N, C, C3F5,N,C3F14,N,N,C,C3F8,C3C1,
 C3F3]
 606 CALL NCR817[C3C10,C3C12,C3F5,N,N,C,N,N,C,N,N,C]
 607 CALL NCR817[C3F8,C3B15,C3E3,C3F8,C3B2,C3E14,C3F8,C3B6,C3E8
 ,C3F8,C3B9,C3E5]
 608 CALL NCR817[C3F14,C3A15,C3D3,C3F14,C3A2,C3D14,C3F14,C3A6,C
 3D8,C3F14,C3A9,C3D5]
 640 DELCO=DELCO

Figure II-1. (continued)

```

700      DELC=(C3D5+C3D8*2.+C3D14*4.+C3D3*8.+C3E5*16.+C3E8*32.+C3E14*64.+C3E3*128.+C3E3*256
) *30./256.-30.
701      12 CONTINUE
702      DDELR=2.*(DELC-DELR)
703      DELH=DELR+DDELR*DT
704      DELF)=0.
705      IF(DELR.GT.KD) DELRD=DELR-KD
706      IF(DELR.LT.-KD) DELRD=DELR+KD
707      DDSI=K8*DELRD-DSI
708      DSI=DSI+DDSI*DT
709      SI=SI+DSI*DT
710      DXRS=VS*CCSI/57.2961
711      DYRS=VS*SI/57.2961
712      XRS=XRS+DXRS*DT
713      YRS=YRS+DYRS*DT
714      XST=XT-XRS
715      YST=YT-YRS
716      RST=SQRT(XST**2+YST**2)
717      AST=S7.296*ATAN(YST/XST)
718      DAST=(AST-ASTOLD)/DT
719      ASTOLD=AST
720      VCC=-RST*DAST/57.296
721      IF(RST.LT.50.) GO TO 600
722      400 CONTINUE
723      WRITE(3,200) TIME,VCC
724      WRITE(4,200) YRS,XRS
725      TIP(KIP,1)=TIME
726      TIP(KIP,2)=DELE
727      TIP(KIP,3)=DELC
728      IF((KIP/2.-AINT(KIP/2.)).EQ.0.) GO TO 500
729      WRITE(1,201) TIME,XRS,YRS,VCC
730      500 CONTINUE
731      600 WRITE(1,201) TIME,XRS,YRS,VCC
732      DC 300 L=1,KIP
733      300 WRITE(2,200) TIP(L,1),TIP(L,2)
734      200 FORMAT(E12.3,S,S,E12.3,S,S)
735      ONE=1.E37
736      WRITE(2,200) ONE,ONE
737      DC 301 L=1,KIP
738      301 WRITE(2,200) TIP(L,1),TIP(L,3)
739      201 FORMAT(4(4X,F10.4))
740      END

```

APPENDIX III

INTEGRATED CIRCUITS COMPUTER SIMULATION
LISTING USED IN THE ADAPTIVE DEADBAND COMPENSATOR

The integrated circuits used in the deadband compensator were simulated on an interactive computer. The simulation program was used as a subroutine package for the main program of the adaptive deadband compensator.

Figure III-1. Listing of Computer Simulation of Integrated Circuits
Used in Adaptive Deadband Compensator

```

900 SUBROUTINE JK856P(CP,J1,K1,SD,O1,OB1,J2,K2,O2,OB2,T,CPO,SO
1NP1,SQ2NP1)
900.3 REAL J1,J2,K1,K2
900.4 IF(T.GT.0) GO TO 10
900.8 CPO=ABS(CP-1.)
901.2 O1=0
901.6 O2=0
902 S01NP1=O1
902.4 SQ2NP1=O2
902.8 10 DCP=CP-CPO
903.2 CPO=CP
903.6 IF(DCP.EQ.0.)GO TO 20
904 IF(DCP.LT.0) O1=S01NP1
904.4 IF(DCP.LT.0) O2 =SQ2NP1
904.8 IF(DCP.GT.0) S01NP1=J1.AND.ABS(K1-1).OR.ABS(O1-1).AND.J1.A
ND.K1.OR.ABS(J1-1).AND.ABS(K1-1).AND.O1
905.2 IF(DCP.GT.0) SQ2NP1=J2.AND.ABS(K2-1).OR.ABS(O2 -1.).AND.J2
.AND.K2.OR.ABS(J2 -1).AND.ABS(K2-1).AND.O2
905.6 20 CONTINUE
906 O1=O1.OR.ABS(SD-1)
906.4 O2=O2.OR.ABS(SD-1)
906.8 OB1=ABS(O1-1)
907.2 OB2=ABS(O2-1)
907.6 RETURN
908 END
908.4 SUBROUTINE D778(CP,D1,CD1,O1,OB1,D2,CD2,O2,OB2,T,CPO,SQ1NP
1,SQ2NP1)
908.8 IF(T.GT.0) GO TO 10
909.2 CPO=ABS(CP-1)
909.6 O1=0
910 O2=0
910.4 SQ1NP1=O1
910.8 SQ2NP1=O2
911.2 10 ECP=CP-CPO

```

```
911.6 CPO=CP
912 IF(DCP.LT.0) Q1=SQ1NP1
912.4 IF(DCP.LT.0) Q2 =SQ2NP1
912.8 IF(DCP.GT.0) SQ1NP1=D1.AND.ABS[CD1-1].CR.CD1
913.2 IF(DCP.GT.0) SQ2NP1=D2.AND.ABS[CD2-1].CR.CD2
913.6 20 CONTINUE
914 Q1=Q1.CR.CD1
914.4 Q2=Q2.CR.CD2
914.8 QB1=ABS[Q1-1]
915.2 QB2=ABS[Q2-1]
915.6 RETURN
916 END
916.4 SUBROUTINE BL7475[CP,D1,Q1,QB1,D2,Q2,QB2,D3,Q3,QB3,D4,Q4,Q
B4,CPO,T]
916.8 IF(T.EQ.0) CPO=ABS[CP-1]
917.2 IF(CP.GT.0.) GC TC 10
917.6 DCP=CP-CPO
918 CPO=CP
918.4 IF(DCP.LT.0) GC TC 10
918.8 GC TC 20
919.2 10 Q1=D1
919.6 Q2=D2
920 Q3=D3
920.4 Q4=D4
920.8 QB1=ABS[Q1-1]
921.2 QB2=ABS[Q2-1]
921.6 QB3=ABS[Q3-1]
922 QB4=ABS[Q4-1]
922.4 20 RETURN
922.8 END
923.2 SUBROUTINE NCR817[A1,B1,C1,A2,B2,C2,A3,B3,C3,A4,B4,C4]
923.6 CR1=A1.CR.B1
924 CR2=A2.CR.B2
924.4 CR3=A3.CR.B3
924.8 CR4=A4.CR.B4
925.2 C1=ABS[CR1-1]
925.6 C2=ABS[CR2-1]
926 C3=ABS[CR3-1]
926.4 C4=ABS[CR4-1]
926.8 RETURN
927.2 END
927.6 SUBROUTINE ND7430[A1,A2,A3,A4,A5,A6,A7,A8,A9,C]
928 AND=A1.AND.A2.AND.A3.AND.A4.AND.A5.AAND.A6.AND.A7.AND.A8.AN
D.A9
```

Figure III-1. (continued)

```

928.4 C=ABS[AND-1]
928.8 RETURN
929.2 END
929.6 SUBROUTINE ND846P[A1,B1,C1,A2,B2,C2,A3,B3,C3,A4,B4,C4]
930 C1B=A1.AND.B1
930.4 C2B=A2.AND.B2
930.8 C3B=A3.AND.B3
931.2 C4B=A4.AND.B4
931.6 C1=ABS[C1B-1]
932 C2=ABS[C2B-1]
932.4 C3=ABS[C3B-1]
932.8 C4=ABS[C4B-1]
933.2 RETURN
933.6 END
934 SUBROUTINE AD7483[CN,A1,B1,S1,A2,B2,S2,A3,B3,S3,A4,B4,S4,C
NP4]
934.4 A1B1=A1.CR.B1
934.8 CNB1=CN.CR.B1
935.2 CNA1=CN.CR.A1
935.6 S1=CN.AND.A1.AND.B1.CR.CN.AND.ABS[A1B1-1].CR.A1.AND.ABS[CN
B1-1].CR.B1.AND.ABS[CNA1-1]
936 CS1=CN.AND.A1.AND.B1.CR.CN.AND.A1.CR.CN.AND.B1.CR.A1.AND.B
1
936.4 A2B2=A2.CR.B2
936.8 CS1B2=CS1.CR.B2
937.2 CS1A2=CS1.CR.A2
937.6 S2=CS1.AND.A2.AND.B2.CR.CS1.AND.ABS[A2B2-1].CR.A2.AND.AB
S[CS1B2-1].CR.B2.AND.ABS[CS1A2-1]
938 CS2=CS1.AND.A2.AND.B2.CR.CS1.AND.A2.CR.CS1.AND.B2.CR.A2.AN
D.B2
938.4 A3B3=A3.CR.B3
938.8 CS2A3=CS2.CR.A3
939.2 CS2B3=CS2.CR.B3
939.6 S3=CS2.AND.A3.AND.B3.CR.CS2.AND.ABS[A3B3-1].CR.A3.AND.ABS[
CS2B3-1].CR.B3.AND.ABS[CS2A3-1]
940 CS3=CS2.AND.A3.AND.B3.CR.CS2.AND.A3.CR.CS2.AND.B3.CR.A3.AN
D.B3
940.4 A4B4=A4.CR.B4
940.8 CS3A4=CS3.CR.A4
941.2 CS3B4=CS3.CR.B4
941.6 S4=CS3.AND.A4.AND.B4.CR.CS3.AND.ABS[A4B4-1].CR.A4.AND.ABS[
CS3B4-1].CR.B4.AND.ABS[CS3A4-1]
942 CNP4=CS3.AND.A4.AND.B4.CR.CS3.AND.A4.CR.CS3.AND.B4.CR.A4.A
ND.B4
942.4 RETURN
942.8 END

```


Figure III-1. (continued)

```
943.2 SUBROUTINE AD7482[CN,A1,B1,S1,A2,B2,S2,CNP2]
943.6 A1B1=A1.CR.B1
944 CNB1=CN.CR.B1
944.4 CNA1=CN.CR.A1
944.8 S1=CN.AND.A1.AND.B1.CR.CN.AND.ABS[A1B1-1].CR.A1.AND.ABS[CN
B1-1].CR.B1.AND.ABS[CNA1-1]
945.2 CS1=CN.AND.A1.AND.B1.CR.CN.AND.A1.CR.CN.AND.B1.CR.A1.AND.B
1
945.6 A2B2=A2.CR.B2
946 CS1B2=CS1.CR.B2
946.4 CS1A2=CS1.CR.A2
946.8 S2=CS1.AND.A2.AND.B2.CR.CS1.AND.ABS[A2B2 -1].CR.A2.AND.AB
S[CS1B2-1].CR.B2.AND.ABS[CS1A2-1]
947.2 CNP2=CS1.AND.A2.AND.B2.CR.CS1.AND.A2.CR.CS1.AND.B2.CR.A2.A
ND.B2
947.6 RETURN
948 END
```