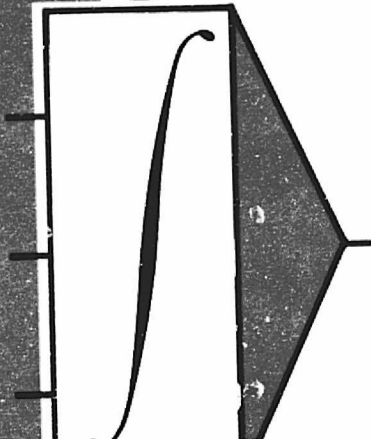
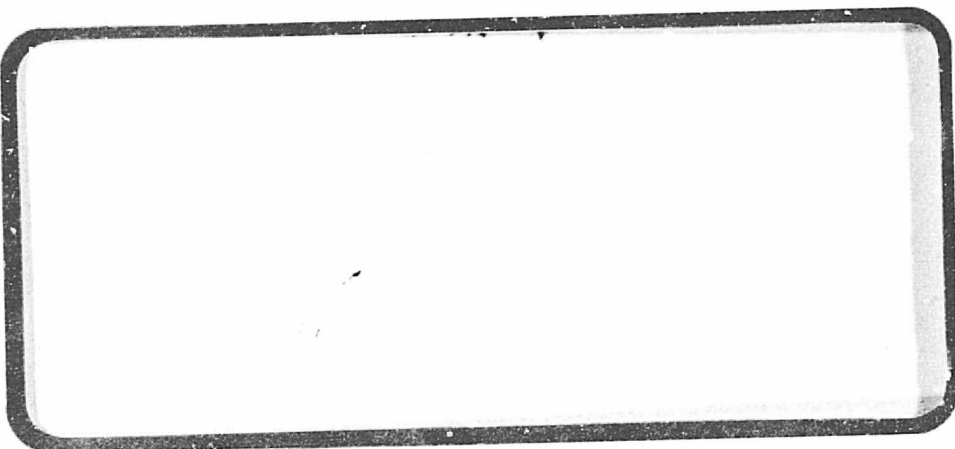


General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

CR160157



SPEED SCIENTIFIC SCHOOL

**THE
SCHOOL
OF
ENGINEERING
AND
APPLIED
SCIENCE**

(NASA-CR-160157) MICROPROCESSOR
REALIZATIONS OF RANGE RATE FILTERS Final
Report (Louisville Univ.) 216 p HC A10/MF
A01 CSCL 09E

N79-21780

Unclas
G3/60 23964



OF →

The University of Louisville
Louisville, Kentucky

40208



FINAL REPORT
MICROPROCESSOR REALIZATIONS
OF RANGE RATE FILTERS
NAS 9-15347

Prepared for
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
JOHNSON SPACE CENTER
HOUSTON, TEXAS

Prepared by
ELECTRICAL ENGINEERING DEPARTMENT
UNIVERSITY OF LOUISVILLE
LOUISVILLE, KY 40208

February 28, 1979

FINAL REPORT
MICROPROCESSOR REALIZATIONS
OF RANGE RATE FILTERS
NAS 9-15347

Approved by: Peter B. Aronhime
Peter B. Aronhime
Project Manager

Donald M. Fleischer
Donald M. Fleischer

February 28, 1979

ABSTRACT

This report evaluates the performance of five digital range rate filters. A range rate filter receives an input of range data from a radar unit and produces an output of smoothed range data and its estimated derivative range rate.

A comparative study of the filters is made through simulation on an IBM 370. Two of the filter designs are also implemented on a 6800 microprocessor-based system.

Comparisons are made on the bases of noise variance reduction ratios and convergence times of the filters in response to simulated range signals.

APPENDIX VIII

TABLE OF CONTENTS

	Page
APPROVAL PAGE	ii
ABSTRACT	iii
TABLE OF CONTENTS	iv
LIST OF TABLES	vi
LIST OF ILLUSTRATIONS	x
I. INTRODUCTION	1
II. DESCRIPTION OF FILTERS	3
A. Filter 1 - Constant Gains and Bandlimited Differentiator	3
B. Filter 2 - Alpha Beta Tracker	5
C. Filter 3 - N Sample Smoother and Cascaded Simple Average Smoother.	8
D. Filter 4 - Digital Tracking Filter	10
E. Filter 5 - Digital Tracking Filter with Skip Sampling	11
III. IBM 370 COMPUTER SIMULATIONS	14
A. Description of Simulating Program	14
B. Test Conditions	16
C. Test Results	17
1. Filter 1	20
2. Filter 2	62
3. Filter 3	95
4. Filter 4	105
5. Filter 5	114
IV. MICROPROCESSOR SIMULATIONS	125
A. Description of Microprocessor System	125
B. Description of Simulating Programs	126
1. Program FILTER 3	126
2. Program FILTER 5	130
C. Test Conditions	133
D. Test Results	134
1. Filter 3	134
2. Filter 5	134
CONCLUSIONS AND RECOMMENDATIONS	141
REFERENCES	143

BIBLIOGRAPHY	144
APPENDIX A. PROGRAM FLT	145
APPENDIX B. PROGRAM FILTER 3	168
APPENDIX C. PROGRAM FILTER 5	184
APPENDIX D. VARIANCE REDUCTION RATIO PROGRAM	200
APPENDIX E. NUMERICAL INTEGRATION	202

LIST OF TABLES

	Page
TABLE I. Filter 1 Range Standard Deviations(ft.) at 2500 samples/sec.	22
TABLE II. Filter 1 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	23
TABLE III. Filter 1 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	24
TABLE IV. Filter 1 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	25
TABLE V. Filter 1 Range Standard Deviations(ft.) at 2500 samples/sec.	26
TABLE VI. Filter 1 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	27
TABLE VII. Filter 1 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	28
TABLE VIII. Filter 1 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	29
TABLE IX. Filter 1 Range Standard Deviations(ft.) at 6.25 samples/sec.	42
TABLE X. Filter 1 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	43
TABLE XI. Filter 1 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	44
TABLE XII. Filter 1 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	45
TABLE XIII. Filter 1 Range Standard Deviations(ft.) at 6.25 samples/sec.	46
TABLE XIV. Filter 1 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	47
TABLE XV. Filter 1 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	48
TABLE XVI. Filter 1 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	49

TABLE XVII.	Filter 2 Range Standard Deviations(ft.) at 2500 samples/sec.	64
TABLE XVIII.	Filter 2 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	65
TABLE XIX.	Filter 2 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	66
TABLE XX.	Filter 2 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	67
TABLE XXI.	Filter 2 Range Standard Deviations(ft.) at 6.25 samples/sec.	82
TABLE XXII.	Filter 2 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	83
TABLE XXIII.	Filter 2 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	84
TABLE XXIV.	Filter 2 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	85
TABLE XXV.	Filter 3 Range Standard Deviations(ft.) at 2500 samples/sec.	97
TABLE XXVI.	Filter 3 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	97
TABLE XXVII.	Filter 3 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	98
TABLE XXVIII.	Filter 3 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	98
TABLE XXIX.	Filter 3 Range Standard Deviations(ft.) at 6.25 samples/sec.	101
TABLE XXX.	Filter 3 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	101
TABLE XXXI.	Filter 3 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	102
TABLE XXXII.	Filter 3 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	102
TABLE XXXIII.	Filter 4 Range Standard Deviations(ft.) at 2500 samples/sec.	106

TABLE XXXIV.	Filter 4 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	106
TABLE XXXV.	Filter 4 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	107
TABLE XXXVI.	Filter 4 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	107
TABLE XXXVII.	Filter 4 Range Standard Deviations(ft.) at 6.25 samples/sec.	110
TABLE XXXVIII.	Filter 4 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	110
TABLE XXXIX.	Filter 4 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	111
TABLE XL.	Filter 4 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	111
TABLE XLI.	Filter 5 Range Standard Deviations(ft.) at 2500 samples/sec.	117
TABLE XLII.	Filter 5 Range Rate Standard Deviations(ft./sec.) at 2500 samples/sec.	117
TABLE XLIII.	Filter 5 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 2500 samples/sec.	118
TABLE XLIV.	Filter 5 Range and Range Rate Variance Reduction Ratios at 2500 samples/sec.	118
TABLE XLV.	Filter 5 Range Standard Deviations(ft.) at 6.25 samples/sec.	121
TABLE XLVI.	Filter 5 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	121
TABLE XLVII.	Filter 5 Maximum Range(ft.) and Range Rate(ft./sec.) Errors at 6.25 samples/sec.	122
TABLE XLVIII.	Filter 5 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	122
TABLE XLIX.	Filter 3 Range Standard Deviations(ft.) at 6.25 samples/sec.	135
TABLE L.	Filter 3 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	135

TABLE LI.	Filter 3 Range and Range Variance Reduction Ratios at 6.25 samples/sec.	136
TABLE LII.	Filter 5 Range Standard Deviations(ft.) at 6.25 samples/sec.	138
TABLE LIII.	Filter 5 Range Rate Standard Deviations(ft./sec.) at 6.25 samples/sec.	138
TABLE LIV.	Filter 5 Range and Range Rate Variance Reduction Ratios at 6.25 samples/sec.	139

LIST OF ILLUSTRATIONS

	Page
FIGURE 1 - Filter 1 (continuous system)	3
FIGURE 2 - Filter 1 (discrete system)	4
FIGURE 3 - Filter 2	6
FIGURE 4 - Filter 3	9
FIGURE 5 - Filter 4	10
FIGURE 6 - Filter 5	12
FIGURE 7 - Filter 1 (Constant Gains) at 2500 samples/sec.	30
FIGURE 8 - Filter 1 (Constant Gains) at 2500 samples/sec.	31
FIGURE 9 - Filter 1 (Cosntant Gains) at 2500 samples/sec.	32
FIGURE 10 - Filter 1 (Constant Gains) at 2500 samples/sec.	33
FIGURE 11 - Filter 1 (Constant Gains) at 2500 samples/sec.	34
FIGURE 12 - Filter 1 (Constant Gains) at 2500 samples/sec.	35
FIGURE 13 - Filter 1 (BLD) at 2500 samples/sec.	36
FIGURE 14 - Filter 1 (BLD) at 2500 samples/sec.	37
FIGURE 15 - Filter 1 (BLD) at 2500 samples/sec.	38
FIGURE 16 - Filter 1 (BLD) at 2500 samples/sec.	39
FIGURE 17 - Filter 1 (BLD) at 2500 samples/sec.	40
FIGURE 18 - Filter 1 (BLD) at 2500 samples/sec.	41
FIGURE 19 - Filter 1 (Constant Gains) at 6.25 samples/sec.	50
FIGURE 20 - Filter 1 (Constant Gains) at 6.25 samples/sec.	51
FIGURE 21 - Filter 1 (Constant Gains) at 6.25 samples/sec.	52
FIGURE 22 - Filter 1 (Constant Gains) at 6.25 samples/sec.	53
FIGURE 23 - Filter 1 (Constant Gains) at 6.25 samples/sec.	54
FIGURE 24 - Filter 1 (Constant Gains) at 6.25 samples/sec.	55
FIGURE 25 - Filter 1 (BLD) at 6.25 samples/sec.	56

FIGURE 26 - Filter 1 (BLD) at 6.25 samples/sec.	57
FIGURE 27 - Filter 1 (BLD) at 6.25 samples/sec.	58
FIGURE 28 - Filter 1 (BLD) at 6.25 samples/sec.	59
FIGURE 29 - Filter 1 (BLD) at 6.25 samples/sec.	60
FIGURE 30 - Filter 1 (BLD) at 6.25 samples/sec.	61
FIGURE 31 - Filter 2 at 2500 samples/sec.	68
FIGURE 32 - Filter 2 at 2500 samples/sec.	69
FIGURE 33 - Filter 2 at 2500 samples/sec.	70
FIGURE 34 - Filter 2 at 2500 samples/sec.	71
FIGURE 35 - Filter 2 at 2500 samples/sec.	72
FIGURE 36 - Filter 2 at 2500 samples/sec.	73
FIGURE 37 - Filter 2 at 2500 samples/sec.	74
FIGURE 38 - Filter 2 at 2500 samples/sec.	75
FIGURE 39 - Figure 2 at 2500 samples/sec.	76
FIGURE 40 - Filter 2 at 2500 samples/sec.	77
FIGURE 41 - Filter 2 at 2500 samples/sec.	78
FIGURE 42 - Filter 2 at 2500 samples/sec.	79
FIGURE 43 - Filter 2 at 2500 samples/sec.	80
FIGURE 44 - Filter 2 at 2500 samples/sec.	81
FIGURE 45 - Filter 2 at 6.25 samples/sec.	86
FIGURE 46 - Filter 2 at 6.25 samples/sec.	87
FIGURE 47 - Filter 2 at 6.25 samples/sec.	88
FIGURE 48 - Filter 2 at 6.25 samples/sec.	89
FIGURE 49 - Filter 2 at 6.25 samples/sec.	90
FIGURE 50 - Filter 2 at 6.25 samples/sec.	91
FIGURE 51 - Filter 2 at 6.25 samples/sec.	92

FIGURE 52 - Filter 2 at 6.25 samples/sec.	93
FIGURE 53 - Filter 2 at 6.25 samples/sec.	94
FIGURE 54 - Filter 3 at 2500 samples/sec.	99
FIGURE 55 - Filter 3 at 2500 samples/sec.	100
FIGURE 56 - Filter 3 at 6.25 samples/sec.	103
FIGURE 57 - Filter 3 at 6.25 samples/sec.	104
FIGURE 58 - Filter 4 at 2500 samples/sec.	108
FIGURE 59 - Filter 4 at 2500 samples/sec.	109
FIGURE 60 - Filter 4 at 6.25 samples/sec.	112
FIGURE 61 - Filter 4 at 6.25 samples/sec.	113
FIGURE 62 - Filter 5 at 2500 samples/sec.	119
FIGURE 63 - Filter 5 at 2500 samples/sec.	120
FIGURE 64 - Filter 5 at 6.25 samples/sec.	123
FIGURE 65 - Filter 5 at 6.25 samples/sec.	124
FIGURE 66 - Flowchart for program FILTER 3	129
FIGURE 67 - Flowchart for program FILTER 5	132

I. INTRODUCTION

The range rate filters discussed in this report are digital filters designed to accept a digitized input from navigational equipment, representing a range value, and to output a smoothed range value and its estimated derivative, range rate. Five filter designs are evaluated. Each is based on different design concepts; hence, the filters have widely varying capabilities.

The purpose of this report is to present a comparative study of the five digital range rate filters. With this in mind, a section detailing the theory of operation for each filter is provided. As a comparative study, the main objective is to present actual test data in such a manner that direct comparisons can be easily drawn between the widely varying filter types. Based on this data, conclusions are made as to filter performance in terms of variance reduction ratios, convergence times and ease of implementation.

The report is divided into two major sections -- full-scale computer simulations and microprocessor simulations. In the first section, the five filter designs are implemented on an IBM 370 computer. Given the higher precision available on a full-scale computer, these simulations represent best case results. Each filter is subjected to the same tests performed at two sampling rates: 2500 samples/second and 6.25 samples/second. The data is presented in tables for each filter.

From the information obtained from the full-scale computer simulations, two filter designs are chosen to be implemented on a microprocessor based system. The limitations on speed and word size imposed by the microprocessor require special attention be given to programming the

filter algorithms. The resulting simulation data is presented along with equivalent IBM 370 simulation data for direct comparison.

II. DESCRIPTION OF FILTERS

A. Filter 1

Filter 1 is a digitized version of the continuous system described by the transfer functions

$$H_R(s) = \frac{RE}{RM} = \frac{sK1 + K2}{s^2 + sK1 + K2} \quad (1)$$

$$H_{RR}(s) = \frac{RDOT}{RM} = \frac{sK2}{s^2 + sK1 + K2} \quad (2)$$

for range and range rate, respectively. The block diagram for this analog system is shown below.

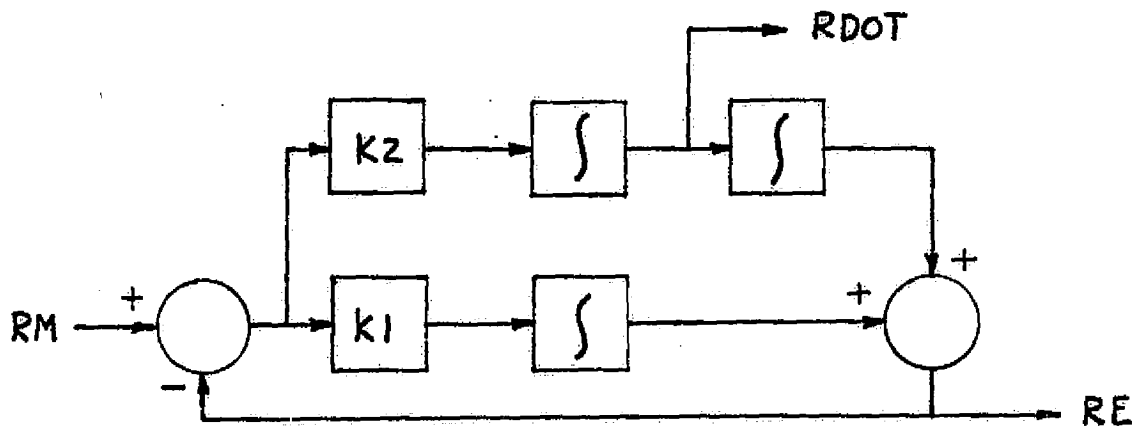


FIGURE 1 - Filter 1 (continuous system)

The discrete system is constructed by replacing the integrators in Figure 1 with discrete integrators using Euler's approximation. A delay must also be added to the feedback loop to make the system physically realizable. This method of mapping from the s-plane to the z-plane is

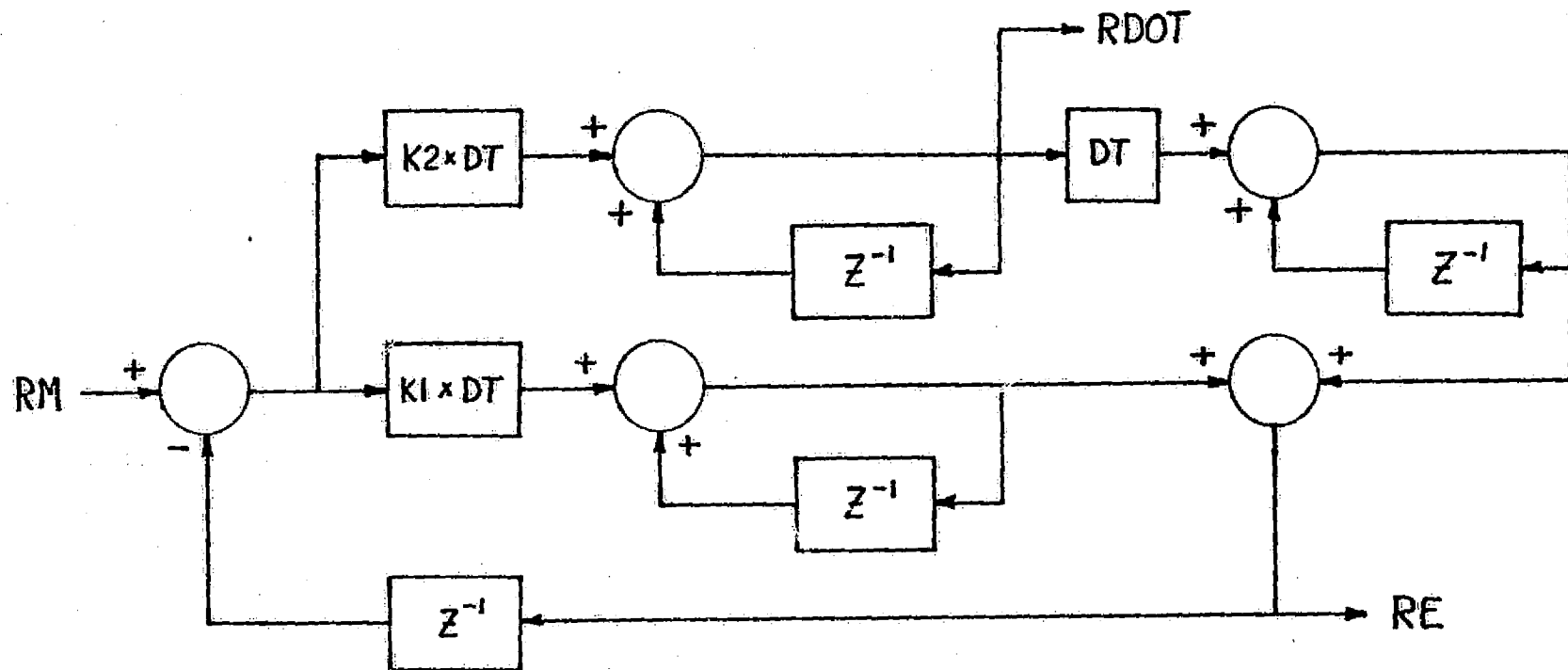


FIGURE 2 - Filter 1 (discrete system)

discussed in further detail in Appendix E. The resulting discrete system is presented in Figure 2. This system is characterized by the transfer functions:

$$H_R(z) = \frac{RE}{RM} = \frac{K1DT + K2DT^2 - z^{-1}(K1DT)}{1 + z^{-1}(-2 + K1DT + K2DT^2) + z^{-2}(1 - K1DT)} \quad (3)$$

$$H_{RR}(z) = \frac{RDOT}{RM} = \frac{K2DT - z^{-1}(K2DT)}{1 + z^{-1}(-2 + K1DT + K2DT^2) + z^{-2}(1 - K1DT)} \quad (4)$$

The classification of filters described by equations (1) and (2) is termed constant gains. This type of filter is simulated for five sets of parameter pairs, K1 and K2. In addition, if the parameter pairs are chosen such that

$$K1 = b\sqrt{2} \quad (5)$$

and

$$K2 = b^2, \quad (6)$$

where b is a frequency scaling parameter, the constant gains filter becomes a bandlimited differentiator.¹ Filter 1 is also simulated for five parameter sets chosen in accordance with equations (5) and (6).

B. Filter 2

Filter 2 is an alpha-beta tracking filter described by the difference equations:

$$RE(n) = RDI(n) + \alpha (RM(n) - RDI(n)) \quad (7)$$

$$RDOT(n) = RDOT(n - 1) + \frac{\beta}{DT}(RM(n) - RDI(n)) \quad (8)$$

$$RDI(n + 1) = RE(n) + DT(RDOT(n)) \quad (9)$$

A block diagram is presented in Figure 3.

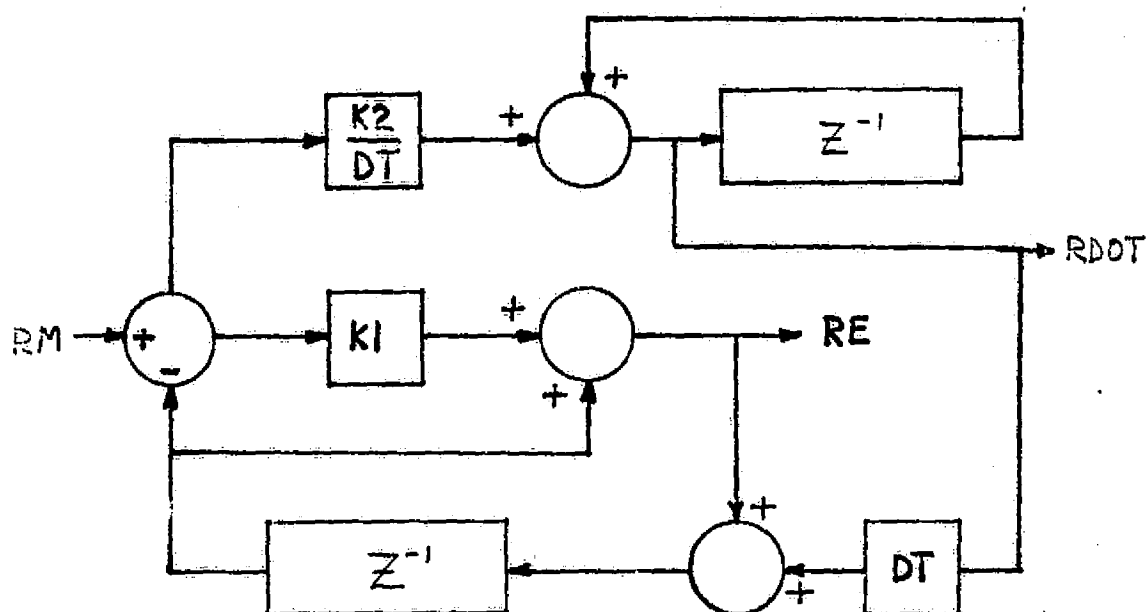


FIGURE 3 - Filter 2

At the n^{th} sample period three values are computed: the estimated range at that sample period, $RE(n)$; the estimated range rate at that sample period, $RDOT(n)$; and a predicted value of the range at the $(n + 1)^{\text{th}}$ sample period, $RDI(n+1)$. The input value is $RM(n)$, and DT is the sample interval. The response of this type filter is controlled by the parameters α and β . For this application, the relation

$$\beta = \frac{\alpha^2}{2 - \alpha} \quad (10)$$

is used to yield the maximum maneuver-following capabilities for a given level of noise rejection.² This results in the trade-off between noise rejection and maneuver-following capability being controlled by only one parameter, α .

The alpha-beta tracker remains a stable system for α and β bounded by³

$$\alpha > 0, \beta > 0 \text{ and } 2\alpha + \beta < 4. \quad (11)$$

For $\alpha = \beta = 1$ (a point on the curve denoted by (10) within the region specified by (11)), the system has a wide bandwidth and reacts quickly to transients. However, the transient-following capability is offset by poor noise rejection. The converse occurs for $\alpha \ll 1$, where the system is slow and has a comparatively low bandwidth. Noise rejection is much improved, but the system reacts more slowly to transients.

The transfer functions for the alpha-beta tracker are

$$H_R(z) = \frac{RE}{RM} = \frac{\alpha + (\beta - \alpha)z^{-1}}{1 + (-2 + \alpha + \beta)z^{-1} + (1 - \alpha)z^{-2}} \quad (12)$$

$$H_{RR}(z) = \frac{RDOT}{RM} = \frac{1}{DT} \frac{\beta - \beta z^{-1}}{1 + (-2 + \alpha + \beta)z^{-1} + (1 - \alpha)z^{-2}} \quad (13)$$

$$H_{Rp}(z) = \frac{RDI}{RM} = \frac{(\alpha + \beta)z^{-1} - \alpha z^{-2}}{1 + (-2 + \alpha + \beta)z^{-1} + (1 - \alpha)z^{-2}}, \quad (14)$$

where H_{Rp} is the transfer function for the predicted range, RDI.

It is interesting to note that the transfer function H_{RR} for Filter 1 can be made identical to H_{RR} for the alpha - beta tracker if the parameter pair $K1$ and $K2$ are chosen such that

$$\alpha = K1 * DT \quad (15)$$

and

$$\beta = K2 * DT^2 \quad (16)$$

The range transfer functions, H_R for Filter 1 and H_{Rp} for the alpha-beta tracker, are also very similar in that they differ only by a factor of z^{-1} . The main difference between the two filters is that the transfer function coefficients are dependent on sampling rate in Filter 1 and independent of sampling rate in Filter 2, the alpha-beta tracker. Simulation verifies that the two filters can be made to perform identically in the range rate mode and very closely in the range mode.

C. Filter 3

A functional block diagram of Filter 3 is presented in Figure 4. Eighteen registers containing the most recent and the previous seventeen range inputs are used in a cascaded simple average smoother to derive range rate. The sum of the six oldest range values is subtracted from the six most recent values. The result is a sum of six terms in which each term is the change in range that occurs over twelve sample periods. Division by six yields an average change in

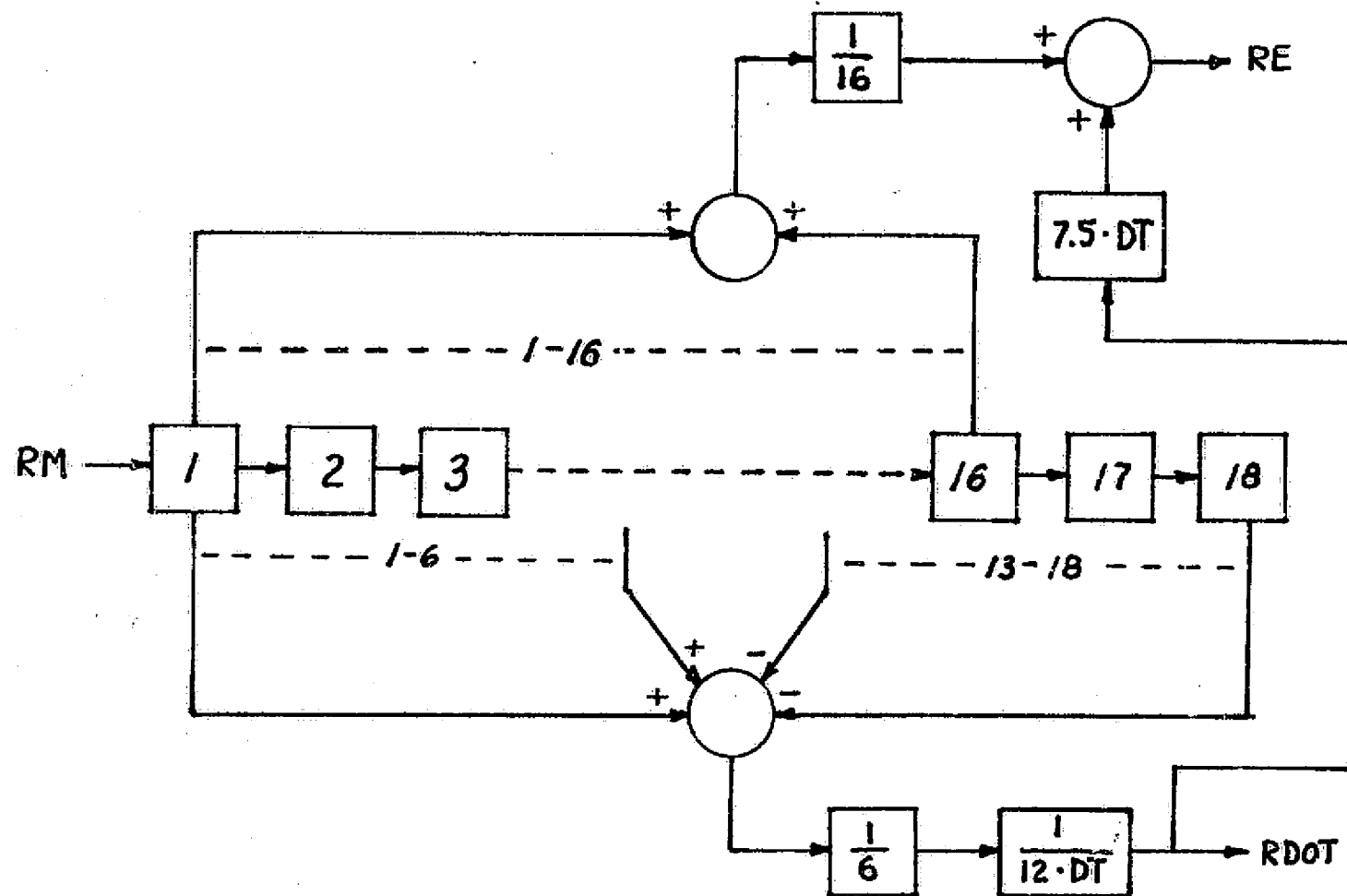


FIGURE 4 - Filter 3

range over twelve sample periods. Further division by $12 * DT$ then results in an estimated range rate value.

An N sample smoother is used to determine smoothed range. The sixteen most recent range inputs are averaged to yield a smoothed range delayed by 7.5 sampling periods from the current sampling period. The current range rate is used to determine a correction term that is added to the smoothed range to update it to the current sampling period.

D. Filter 4

Filter 4 is a digital tracking filter constructed by decreasing the number of storage registers and adding feedback to the design of Filter 3. A block diagram of Filter 4 is presented below.

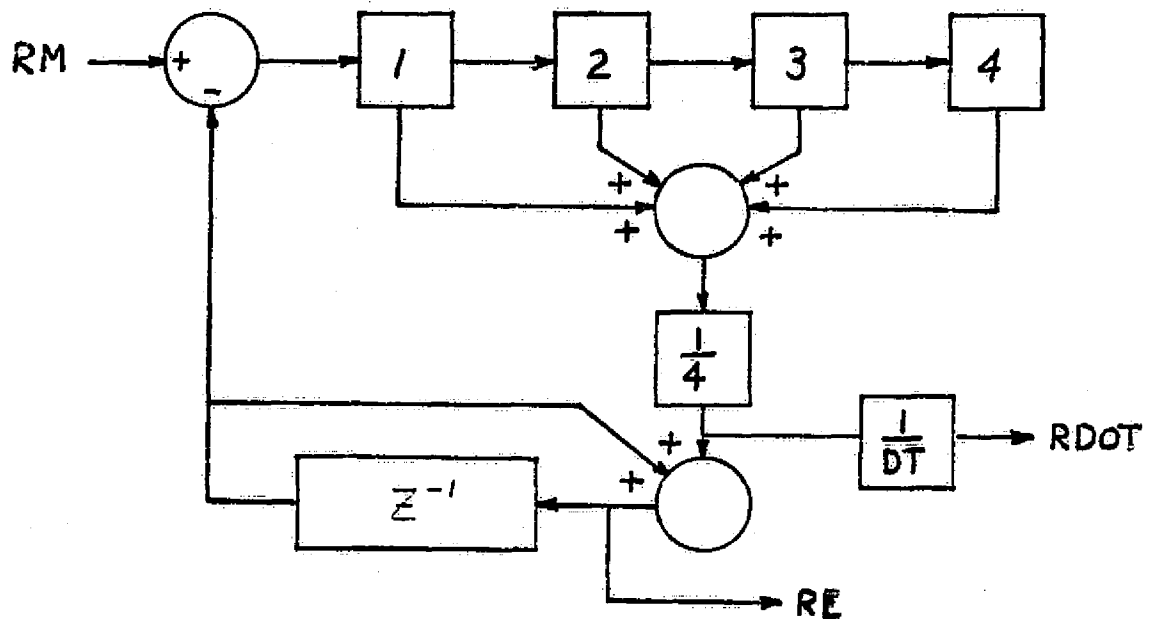


FIGURE 5 - Filter 4

In this design the values contained in the four storage registers now represent range error terms, rather than range terms, due to the

addition of feedback. An average of these four registers is taken resulting in an average range error term. This value is added to the previous estimated range to yield the new estimated range. The estimated range rate value is determined by dividing the average range error term by the sampling period DT . The estimated range is then fed back to the sampled range input to produce a new range error term.

Four is the maximum number of storage registers (samples) that will result in a stable system.⁴

E. Filter 5

Filter 5 is a tracking filter similar to Filter 4 except for the addition of skip sampling. In this design, although the sixteen storage registers are still updated every sample instance, the range feedback term is updated only every sixteen sample periods. Estimated range and range rate are also available only at sixteen-sample-period intervals. A block diagram of Filter 5 is provided in Figure 6.

Every sixteenth sample, two range values are determined: the estimated range at that sample instant and the feedback range which lags the estimated range by 7.5 sample periods. The feedback range is subtracted from each of the next sixteen range inputs so that the value in the sixteenth storage register represents the change in range over 8.5 sample periods, and the value in the fifteenth register represents the change in range over 9.5 sample periods. This same pattern continues through the first storage register which contains the change in range over 23.5 sample periods. An average taken of the sixteen storage registers then yields a value that represents the smoothed change in range over sixteen sample periods. This value is added to

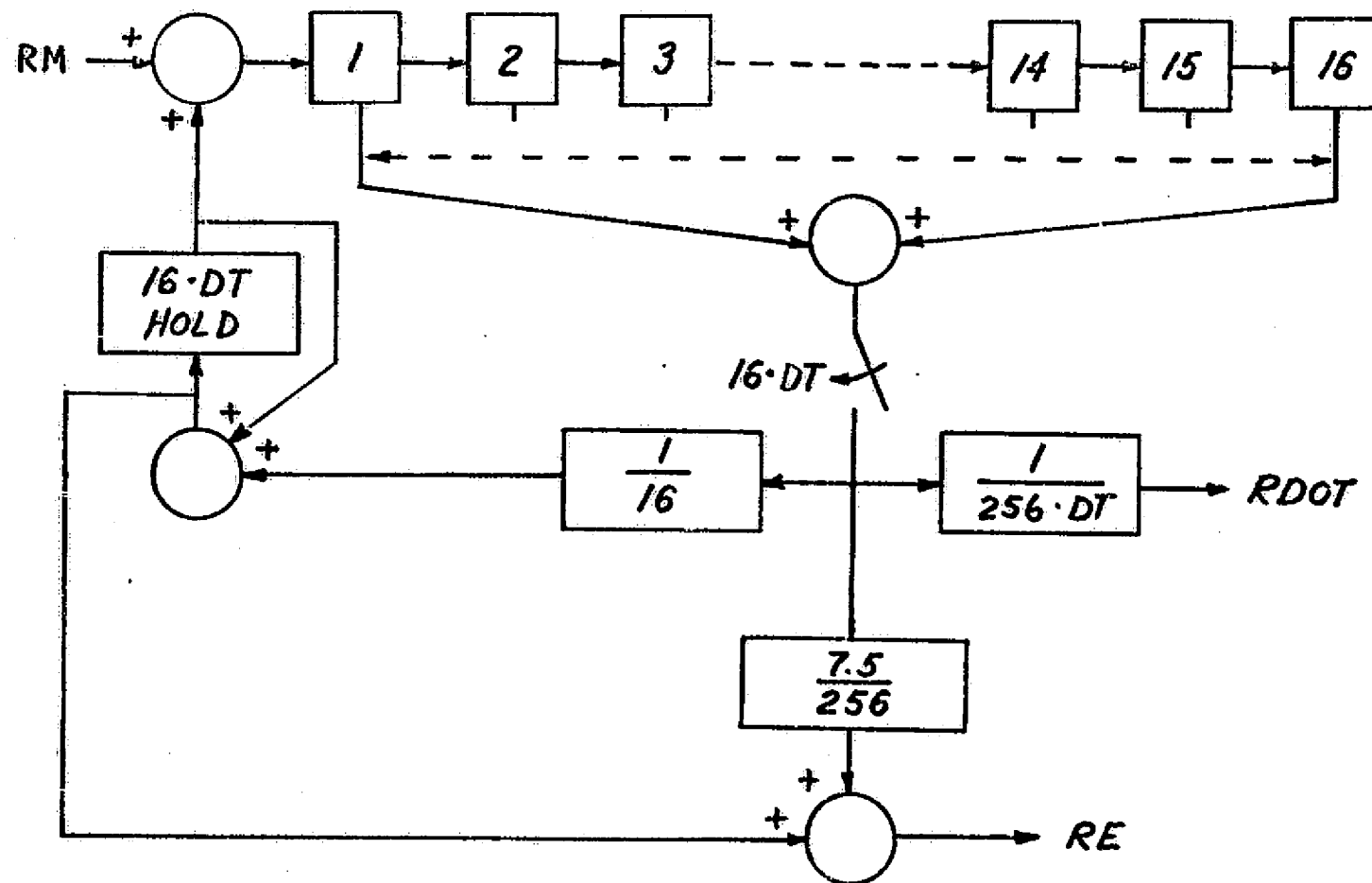


FIGURE 6 - Filter 5

the present feedback range to determine the feedback range to be used for the next set of sixteen range inputs. The smoothed change in range over sixteen sample periods is then multiplied by the ratio $7.5/16$, and the result is added to the new feedback range to yield the estimated range for the present sample. The estimated range rate can be derived by dividing the smoothed change in range by sixteen to determine the change in range over one sample period. Division by the sample period then yields the estimated range rate.

III. IBM 370 COMPUTER SIMULATIONS

A. Description of Simulating Program

Program FLT is the FORTRAN program used in the IBM 370 simulations. In this program an initial range value and range rate value are entered to create a range signal of constant slope. This ramp simulates an object moving with constant velocity and is termed the true or actual range. A random number generator is used to create an array of normally distributed noise values of specified mean and standard deviation. Each filter is simulated for three range inputs. Simulation RUN 1 tests the filter response to the ramp, RUN 2 tests with the noise and RUN 3 uses a range input that is the summation of the ramp and the noise.

Filter calculations are performed by subroutine CALC. This subroutine also determines the input range value for a given sample period. That value is entered into the filter algorithm, and an estimated range and range rate value for that sample period are determined. The value IDI in the program determines how many times this process is repeated before program control returns to FLT. CALC returns to FLT the value at the most recent sampling period of the actual range, the range input, the estimated range, the actual range rate and the estimated range rate. These values along with the range error and range rate error, as determined from the differences of the estimated and actual range and range rate values, are output in tabulated form

and saved in an array to be used in plotting. The value IDRI determines how many times subroutine CALC is called.

At the end of a simulation run, program FLT calculates statistics on the steady state values of various range and range rate signals. The subroutine BECORI is called from the International Mathematical and Statistical Library for this function. The mean and standard deviation are calculated for a set of sample values of the input range, estimated range and the estimated range rate. For all simulation runs at 2500 samples/second and 6.25 samples/second, the last 50 sample values of each of the three signals are saved in an array to be analyzed by BECORI. Since the last 50 samples are used, it is necessary that the simulation runs be of sufficient duration to allow initial transients to abate. This will insure that the samples represent steady state signals so that statistically accurate results can be obtained.

Program FLT contains two plotting routines for graphical representation of data. The first, PLOT B, implements the printer as a plotter. Although it provides limited resolution it is useful as a quick check of simulation results. The second routine, MCPLLOT, formats data for a drum type plotter. After simulation RUN 3, four plots are generated for a particular filter - two range plots and two range rate plots. The first plot of each pair contains the graph of the actual signal and the resulting estimated signal over the full duration of that simulation run. However, in some cases the magnitude of the initial transient causes the scale of the plot to be too large to display any information on the steady state signal. For this reason, the second plot, an after-transient graph of the signal,

is made. It displays the actual and estimated signals after a time delay to allow for decay of the transient. These four plots are generated for every filter being simulated. At the end of program FLT, after all filters have been simulated, four additional plots are generated. These contain the transient and after-transient graphs of the range error and range rate error signals. As an example, if two filters have been simulated, each of the four final plots will contain the graph of two signals -- one for each filter. This allows for a comparison of filter results.

As a clarification, only one type filter can be simulated during a given execution of program FLT, where type of filter refers to Filter 1, Filter 2, etc. References made in the above text to different filters being simulated during the same execution of program FLT mean different parameter pairs such as K1 and K2 used in the same type filter.

A listing of program FLT, subroutine CALC for each of the five filter types, and the plotting routines, PLOT B and MCPLLOT, are given in Appendix A.

B. Test Conditions

As previously noted, three simulation runs are made to evaluate each filter. For the first run, the input signal is a ramp function representing actual range and is described by the equation:

$$r(t) = 60800 - 28t. \quad (17)$$

This is a ramp of negative slope corresponding to an initial range of 18.532 Km (60800 ft.) and a range rate of -8.5344 m/sec. (-28 ft./sec.). The purpose of this run is to evaluate the ability of the filter to accurately track an object moving at constant velocity. This input signal has been used in previous investigations of range rate filters.

The second simulation run determines how well the filter rejects random noise. The input signal is Gaussian noise with a mean (μ) of zero and a standard deviation (σ) of 30.48 m (100 ft.)

The purpose of the third run is to evaluate the response of the filter to noisy range data. The input signal is a summation of the inputs for the previous two runs. This input can be termed a noisy ramp. This run shows what effect the addition of noise has on the ability of the filter to track the actual range value.

In the first set of tests, all filters are run at a sampling rate of 2500 samples/second for a period of 75 seconds. Results are tabulated at one second intervals, and the steady state statistics, mean and standard deviation, are taken on the last 50 samples.

The second set of simulations are made at 6.25 samples/second for a duration of 192 seconds. Results are tabulated every 16 sample periods or 2.56 seconds. Steady state statistics for these simulations are taken on the last 50 samples.

C. Test Results

Data for each filter is divided into two sections: that obtained at a 2500 samples/second sampling rate and that obtained at

6.25 samples/second. Data for each filter is provided in tables and in many cases also presented in graphical form to aid in visualizing actual range and range rate signals. A summary of the filter's performance is given at the beginning of the data for that filter.

Four tables are presented for each filter. The first table gives the input and output range standard deviations for each of the three simulation runs. The second table gives the output range rate standard deviations for each of the three simulation runs. These values give a measure of a signal's variance from its mean. In the case of simulation RUN 1, where the range input is the ramp representing actual range, the desired range output signal is an identical ramp. Hence, the standard deviation of the input and output range signals should ideally be equal. Since the slope of the ramp is constant, it is expected that the estimated range rate signal produced by the filter should also be constant and hence its standard deviation will approach zero. During simulation RUN 2, the input range signal is Gaussian noise of zero mean and known standard deviation. If the noise is indeed attenuated by the filter, then the estimated range output will show a reduction in standard deviation from that of the input range signal. Since the input represents a constant range value (namely zero range) with noise added, the optimum estimated range rate signal returned by the filter would again have a standard deviation approaching zero. The range input provided to the filter in simulation RUN 3 is that of the Gaussian noise superimposed on the ramp. If both the ramp and the noise are independent, random signals and the filter is a linear system, then

$$\sigma_R^2 + \sigma_N^2 = \sigma_{R+N}^2 \quad (18)$$

where σ_R^2 is the variance of the estimated range output for the ramp input, σ_N^2 is the variance for the estimated range output for the noise input and σ_{R+N}^2 is the variance for the estimated range output for the ramp plus noise or noisy ramp input. In this case σ_R^2 , σ_N^2 and σ_{R+N}^2 would represent the square of the estimated range standard deviations for RUN 1, RUN 2 and RUN 3, respectively. However, although the ramp and noise are independent, the ramp is not a random signal so the above relation does not hold true. For this reason, RUN 3 is necessary to determine the results for a noisy ramp range input.

The third table contains the maximum steady state range and range rate errors for the three simulation runs. This information is presented to give another measure of the range over which an estimated range or range rate value will deviate from the actual value.

The fourth table lists the range variance reduction ratio, KR, and the range rate variance reduction ratio, KRR. These ratios are determined from the relations:

$$KR = \frac{\text{variance of range output}}{\text{variance of range input}} \quad (19)$$

and

$$KRR = \frac{\text{variance of range rate output}}{\text{variance of range input}} \quad (20)$$

The variance terms are the square of the standard deviations presented in the first and second tables under RUN 2. This is the simulation run conducted with Gaussian noise as the range input. Whereas the standard deviations provide information on specific signals, the reduction ratios offer a standardized method for comparison of filter results.

The fourth table also lists convergence time τ . This was arbitrarily defined as the time required for the estimated range to reach a steady state condition such that the magnitude of the range error does not exceed 3.048 m (10 ft.). The data used to determine τ is taken from simulation RUN 1, where the range input is the ramp.

1. Filter 1

Simulation data at 2500 samples/second for Filter 1 is presented in Tables I-IV for the constant gains filter and in Tables V-VIII for the bandlimited differentiator. Data at 6.25 samples/second is presented in Tables IX-XII for the constant gains filter and in Tables XIII-XVI for the bandlimited differentiator. A graphical display of the data at 2500 samples/second is shown for two sets of parameter pairs (K_1 , K_2) for the constant gains filter in Figures 7-12 and for the bandlimited differentiator in Figures 13-18. Data at 6.25 samples/second is shown for the constant gains filter in Figures 19-24 and for the bandlimited differentiator in Figures 25-30.

The data indicates that Filter 1 has higher noise rejection at the higher sampling rate. Since this filter is a digital approximation to a continuous system, this is expected.

The digital system more closely approximates a continuous system as the sampling rate increases.

For the bandlimited differentiator (BLD) the effect of b , the frequency scaling parameter, can be noted by observing the variance reduction ratios. In the bandlimited differentiator, the cutoff frequency is determined by the frequency scaling parameter. As the scaling parameter is reduced, the bandwidth of the system decreases. Hence, more lower frequency components of the noise are attenuated and the variance reduction ratios decrease. This, however, also reduces the system's responsiveness to higher frequency changes in range signals which results in a reduction in maneuver-following capabilities. This is shown by the increase in convergence time as the frequency scaling parameter is reduced.

A comparison of the variance reduction ratios shows a slight advantage for the bandlimited differentiator over the constant gains filter. While the constant gains filter produces good range and range rate data, there appears to be a trade off between the two. With the bandlimited differentiator, changing the frequency scaling parameter to improve range data also has the effect of improving range rate performance (at the expense of convergence time).

There are a variety of methods to derive a discretized approximation to a continuous system. The method used in Filter 1 produces a system that is a predictor, i.e., the estimated range returned at the n^{th} sampling instant is a predicted value of the range at the $(n + 1)^{\text{th}}$ sampling instant. This effect can be observed in the maximum steady state range error for simulation

TABLE I

FILTER 1 RANGE STANDARD DEVIATIONS (FT.) AT 2500 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
K1 = 0.7071 K2 = 0.45	0.1634	0.1634	110.7364	0.8773×10^{-1}	110.7339	0.2463
K1 = 0.75 K2 = 0.125	0.1634	0.1634	110.7364	0.9391×10^{-1}	110.7339	0.2523
K1 = 0.7 K2 = 0.3769	0.1634	0.1634	110.7364	0.8717×10^{-1}	110.7339	0.2459
K1 = 0.49 K2 = 0.98995	0.1634	0.1632	110.7364	0.5529×10^{-1}	110.7339	0.2142
K1 = 0.75 K2 = 0.45	0.1634	0.1634	110.7364	0.9325×10^{-1}	110.7339	0.2511

TABLE II

FILTER 1 RANGE RATE STANDARD DEVIATIONS (FT./SEC.) AT 2500 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 1 input = range OUTPUT RANGE RATE STND. DEV.	RUN 2 input = noise OUTPUT RANGE RATE STND. DEV.	RUN 3 input = ramp + noise OUTPUT RANGE RATE STND. DEV.
K1 = 0.7071 K2 = 0.45	0.0	0.5652×10^{-1}	0.5652×10^{-1}
K1 = 0.75 K2 = 0.125	0.0	0.1575×10^{-1}	0.1575×10^{-1}
K1 = 0.7 K2 = 0.3769	0.0	0.4740×10^{-1}	0.4740×10^{-1}
K1 = 0.49 K2 = 0.98995	0.0	0.1238	0.1238
K1 = 0.75 K2 = 0.45	0.0	0.5662×10^{-1}	0.5661×10^{-1}

TABLE III

FILTER 1 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
K1 = 0.7071 K2 = 0.45	0.112×10^{-1}	0.140×10^{-6}	0.278	0.323	0.290	0.323
K1 = 0.75 K2 = 0.125	0.108×10^{-1}	0.221×10^{-3}	0.223	0.138	0.234	0.139
K1 = 0.7 K2 = 0.3769	0.112×10^{-1}	0.225×10^{-7}	0.247	0.264	0.258	0.264
K1 = 0.49 K2 = 0.98995	0.106×10^{-1}	0.464×10^{-4}	0.322	1.395	0.332	1.395
K1 = 0.75 K2 = 0.45	0.112×10^{-1}	0.241×10^{-7}	0.250	0.329	0.261	0.329

TABLE IV

FILTER 1 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 2 input = noise		T CONVERGENCE TIME (SEC.)
	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	
K1 = 0.7071 K2 = 0.45	6.276×10^{-7}	2.605×10^{-7}	23
K1 = 0.75 K2 = 0.125	7.192×10^{-7}	2.023×10^{-8}	35
K1 = 0.7 K2 = 0.3769	6.197×10^{-7}	1.832×10^{-7}	25
K1 = 0.49 K2 = 0.98995	2.493×10^{-7}	1.250×10^{-6}	36
K1 = 0.75 K2 = 0.45	7.091×10^{-7}	2.614×10^{-7}	23

TABLE V

FILTER 1 RANGE STANDARD DEVIATIONS (FT.) AT 2500 SAMPLES/SEC.

FILTER 1 BLD	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
K1 = 0.7778 K2 = 0.3025 b = 0.55	0.1634	0.1634	110.7364	0.9729×10^{-1}	110.7339	0.2251
K1 = 0.7071 K2 = 0.25 b = 0.50	0.1634	0.1634	110.7364	0.8841×10^{-1}	110.7339	0.2468
K1 = 0.6364 K2 = 0.2025 b = 0.45	0.1634	0.1634	110.7364	0.7951×10^{-1}	110.7339	0.2381
K1 = 0.5657 K2 = 0.160 b = 0.40	0.1634	0.1632	110.7364	0.7056×10^{-1}	110.7339	0.2296
K1 = 0.4950 K2 = 0.1225 b = 0.35	0.1634	0.1633	110.7364	0.6157×10^{-1}	110.7339	0.2211

TABLE VI

FILTER 1 RANGE RATE STANDARD DEVIATIONS (FT./SEC.) AT 2500 SAMPLES/SEC.

FILTER 1 BLD	RUN 1	RUN 2	RUN 3
	input = ramp OUTPUT RANGE RATE STND. DEV.	input = noise OUTPUT RANGE RATE STND. DEV.	input = ramp + noise OUTPUT RANGE RATE STND. DEV.
K1 = 0.7778 K2 = 0.3025 b = 0.55	0.0	0.3818×10^{-1}	0.3818×10^{-1}
K1 = 0.7071 K2 = 0.25 b = 0.50	0.0	0.3155×10^{-1}	0.3155×10^{-1}
K1 = 0.6364 K2 = 0.2025 b = 0.45	0.0	0.2556×10^{-1}	0.2556×10^{-1}
K1 = 0.5657 K2 = 0.160 b = 0.40	0.0	0.2019×10^{-1}	0.2019×10^{-1}
K1 = 0.4950 K2 = 0.1225 b = 0.35	0.0	0.1544×10^{-1}	0.1544×10^{-1}

TABLE VII

FILTER 1 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

FILTER 1 BLD	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
K1 = 0.7778 K2 = 0.3025 b = 0.55	0.112	0.901×10^{-8}	0.208	0.246	0.197	0.246
K1 = 0.7071 K2 = 0.25 b = 0.50	0.112	0.128×10^{-6}	0.187	0.223	0.178	0.223
K1 = 0.6364 K2 = 0.2025 b = 0.45	0.112×10^{-1}	0.161×10^{-5}	0.172	0.208	0.161	0.208
K1 = 0.5657 K2 = 0.160 b = 0.40	0.111×10^{-1}	0.149×10^{-4}	0.152	0.199	0.142	0.199
K1 = 0.4950 K2 = 0.1225 b = 0.35	0.119×10^{-1}	0.757×10^{-4}	0.143	0.188	0.155	0.188

TABLE VIII

FILTER 1 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

FILTER 1 BLD	RUN 2		τ CONVERGENCE TIME (SEC.)
	input = noise		
	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	
K1 = 0.7778 K2 = 0.3025 b = 0.55	7.719 x 10 ⁻⁷	1.189 x 10 ⁻⁷	24
K1 = 0.7071 K2 = 0.25 b = 0.50	6.374 x 10 ⁻⁷	8.117 x 10 ⁻⁸	26
K1 = 0.6364 K2 = 0.2025 b = 0.45	5.155 x 10 ⁻⁷	5.328 x 10 ⁻⁸	29
K1 = 0.5657 K2 = 0.160 b = 0.40	4.060 x 10 ⁻⁷	3.324 x 10 ⁻⁸	32
K1 = 0.4950 K2 = 0.1225 b = 0.35	3.091 x 10 ⁻⁷	1.944 x 10 ⁻⁸	37

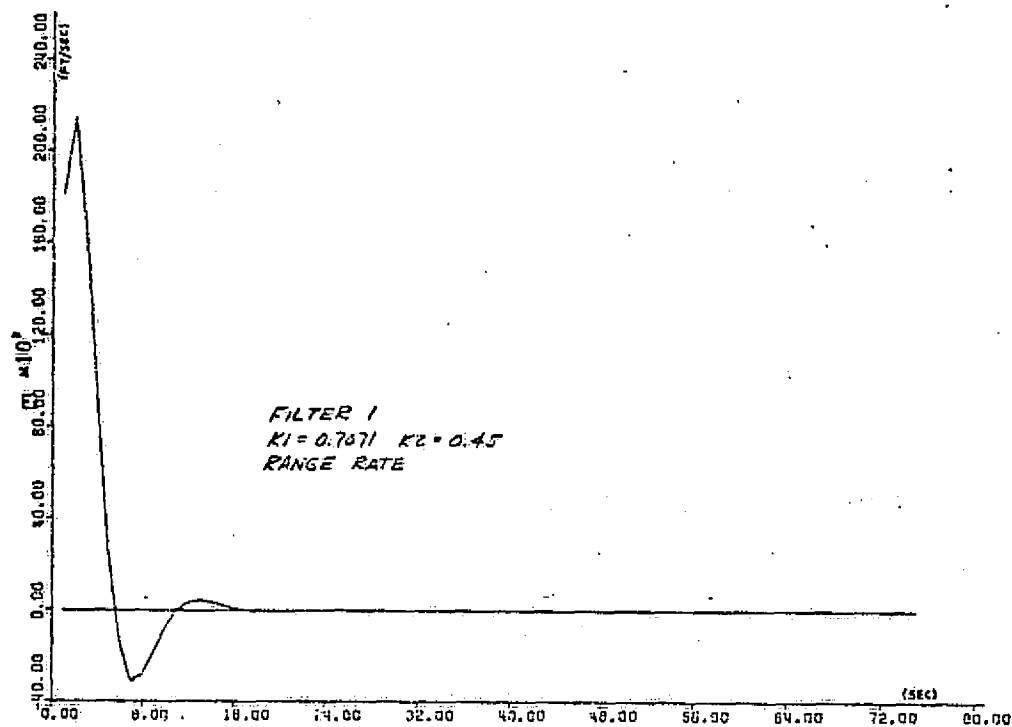
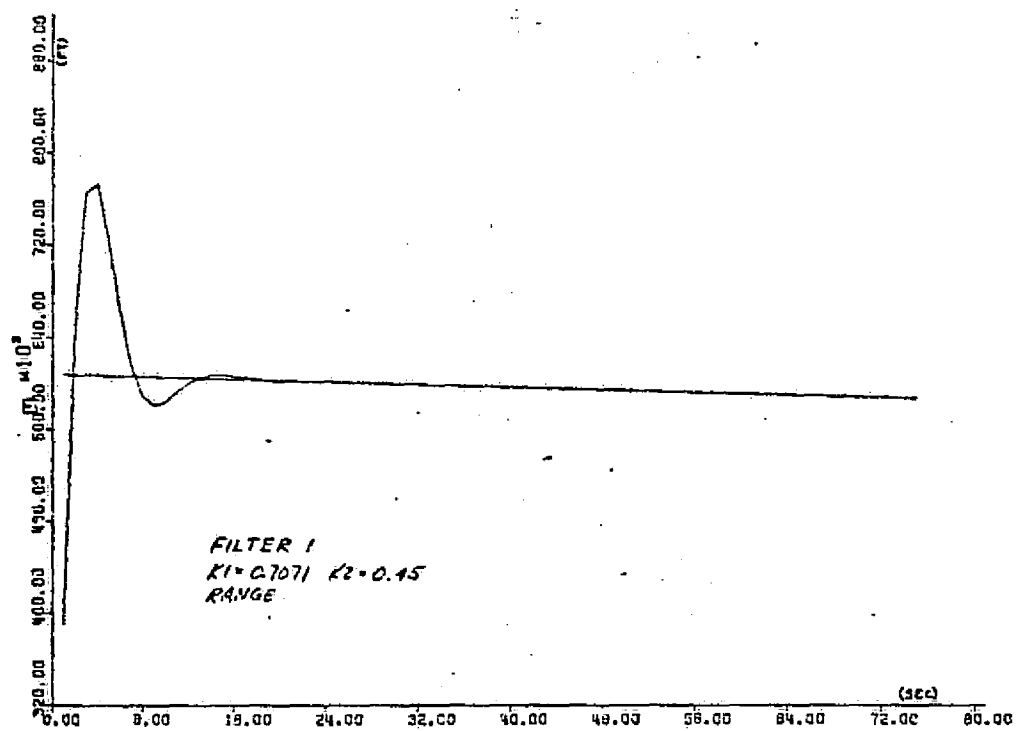


FIGURE 7 - Filter 1 (Constant Gains) at 2500 samples/sec.

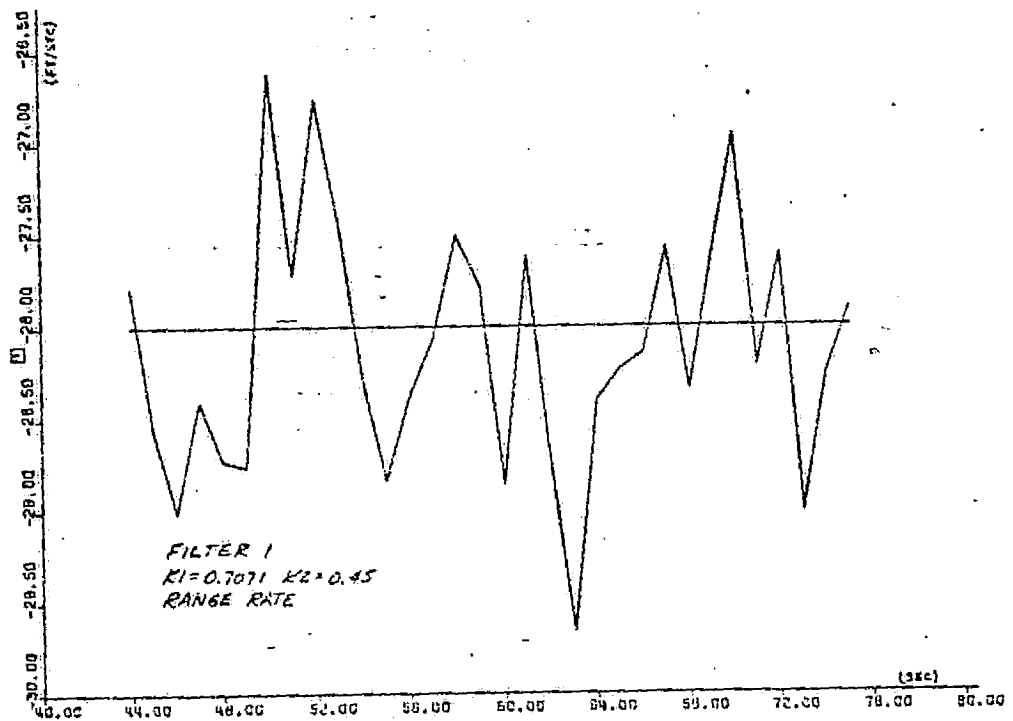
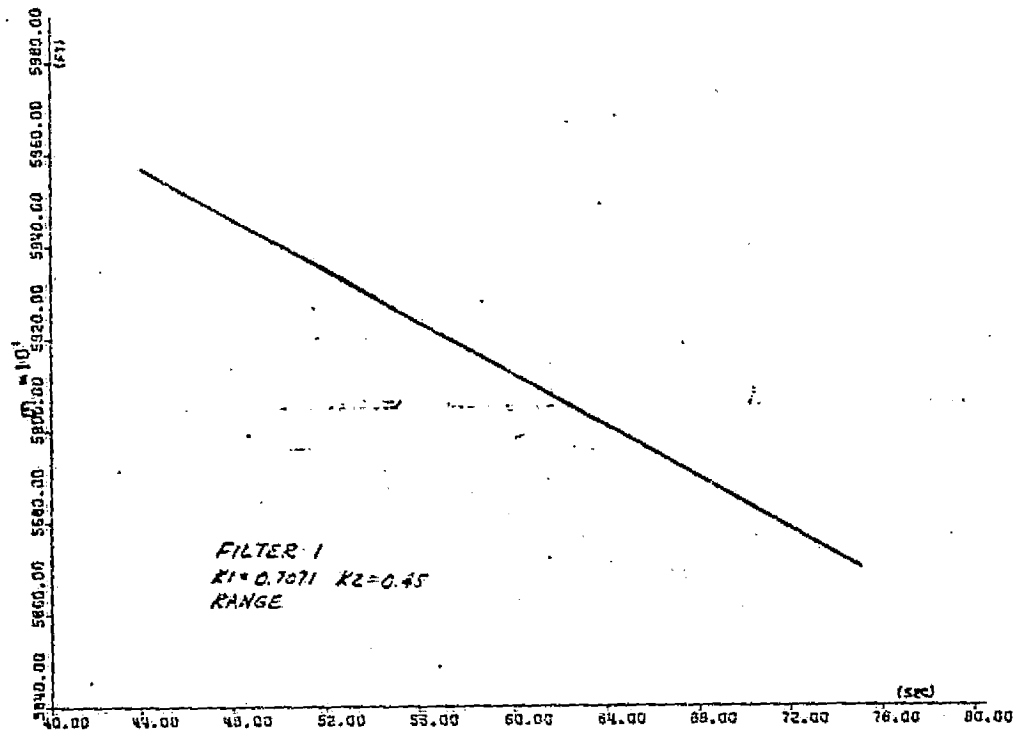


FIGURE 8 - Filter 1 (Constant Gains) at 2500 samples/sec.

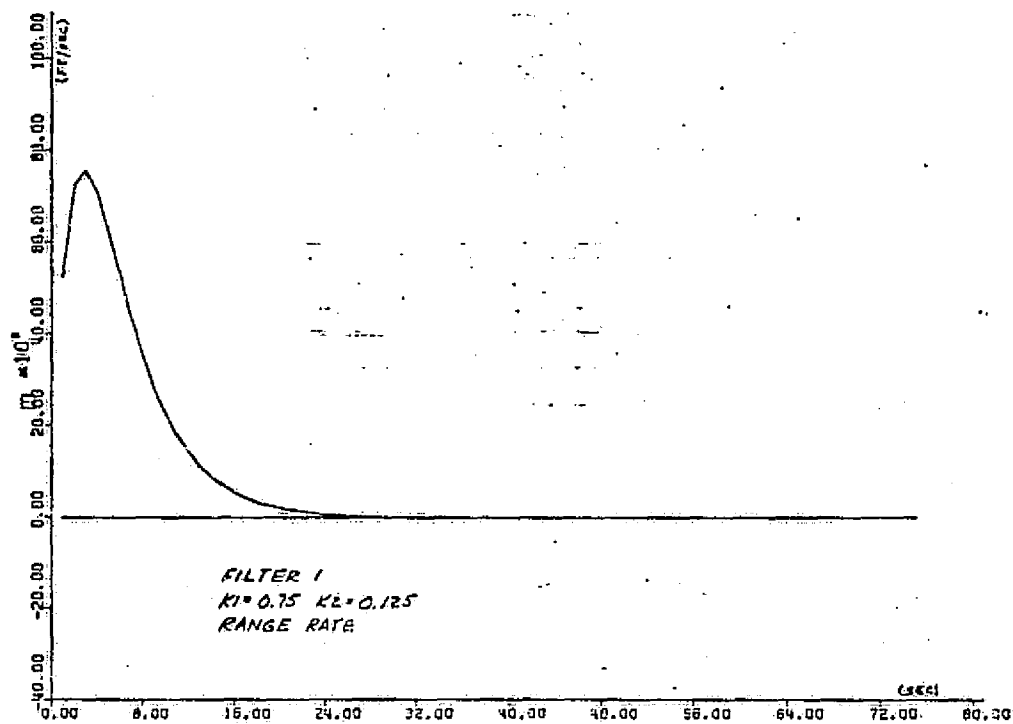
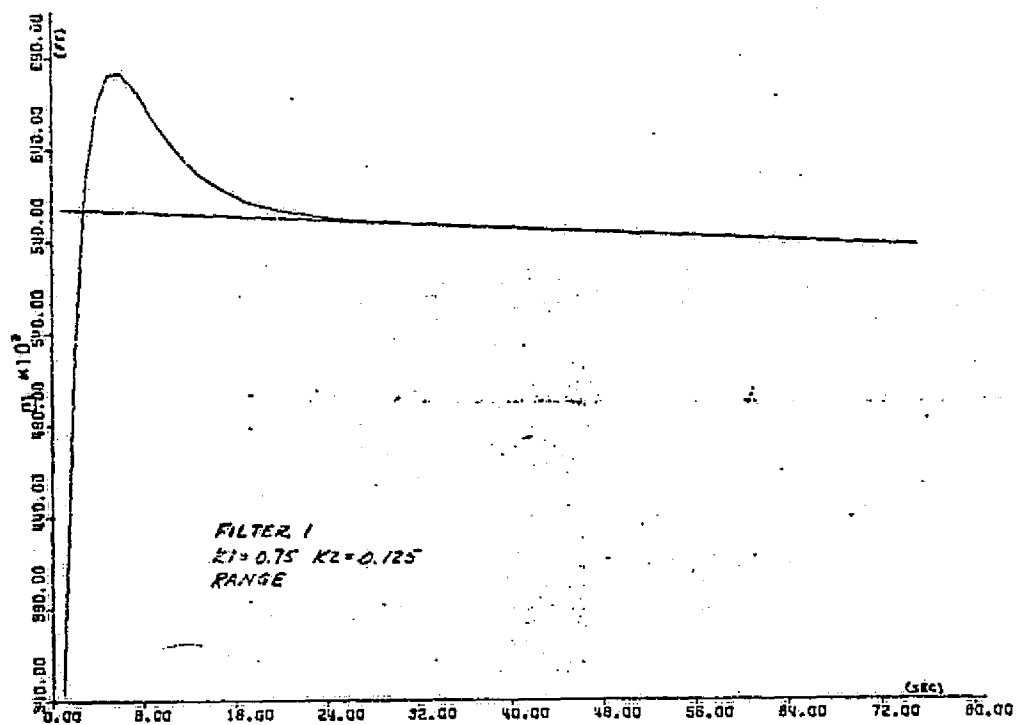


FIGURE 9 - Filter 1 (Constant Gains) at 2500 samples/sec.

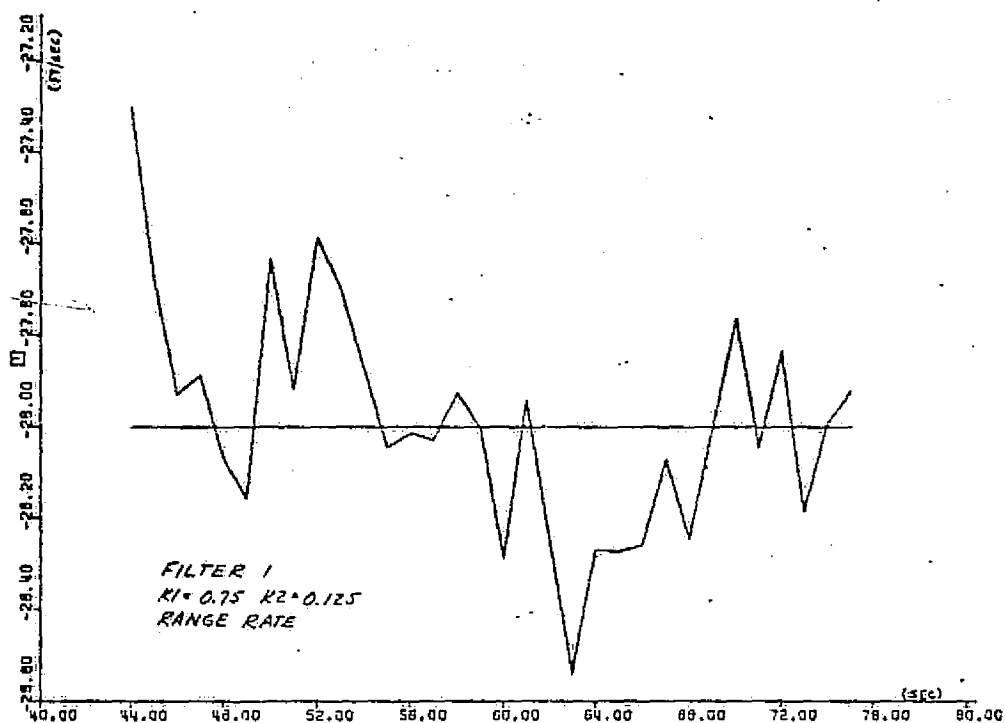
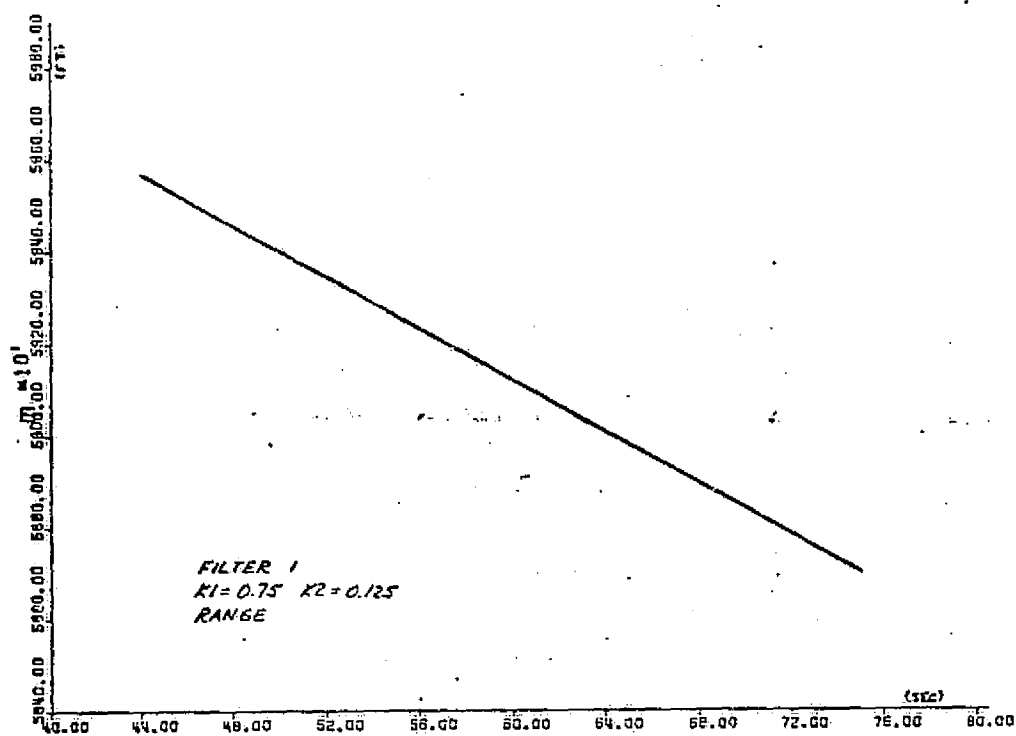


FIGURE 10 - Filter 1 (Constant Gains) at 2500 samples/sec.

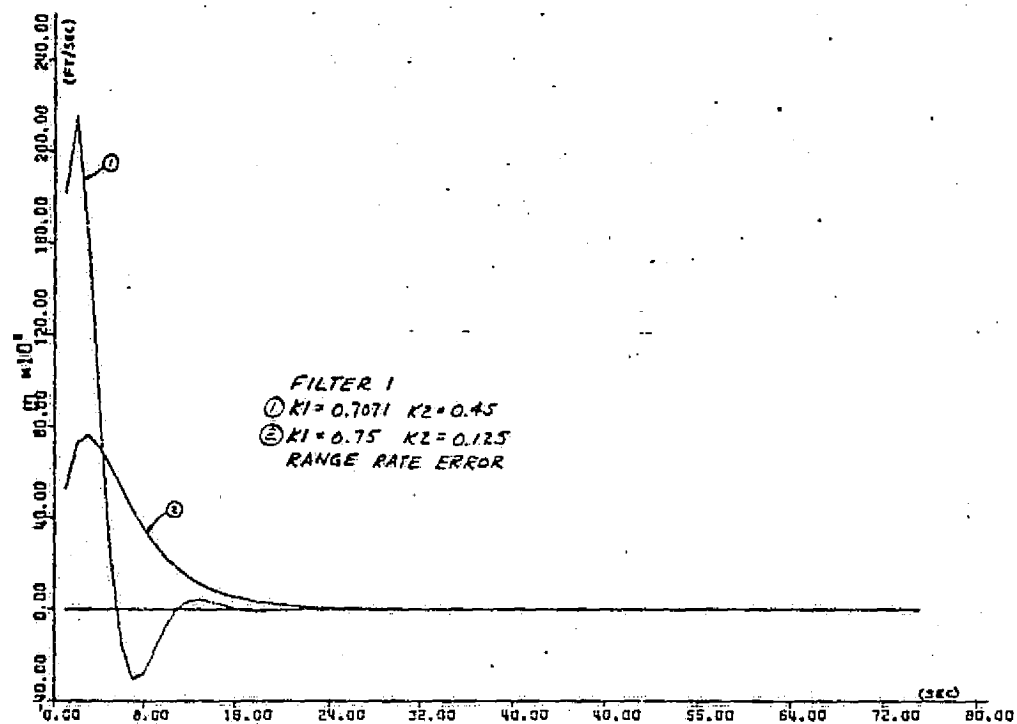
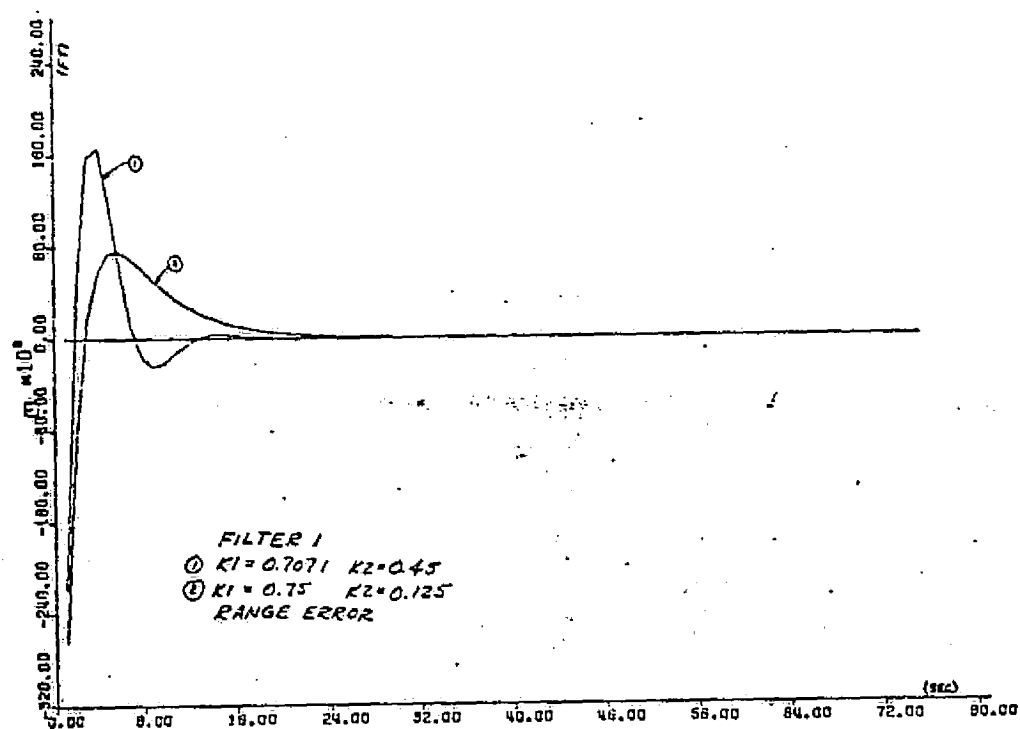


FIGURE 11 - Filter 1 (Constant Gains) at 2500 samples/sec.

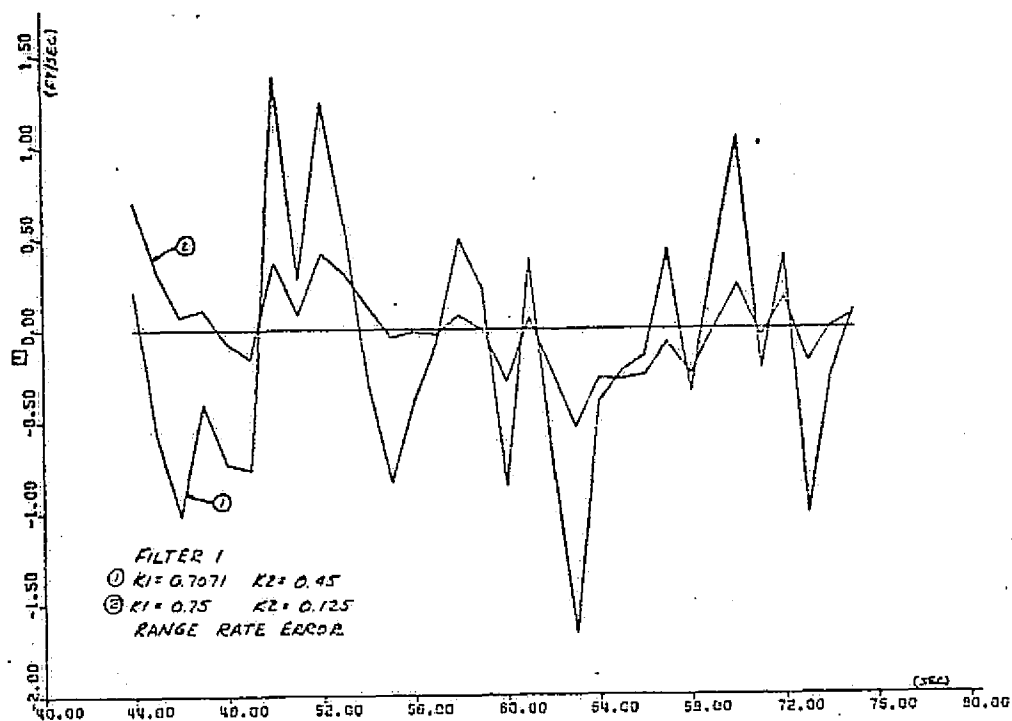
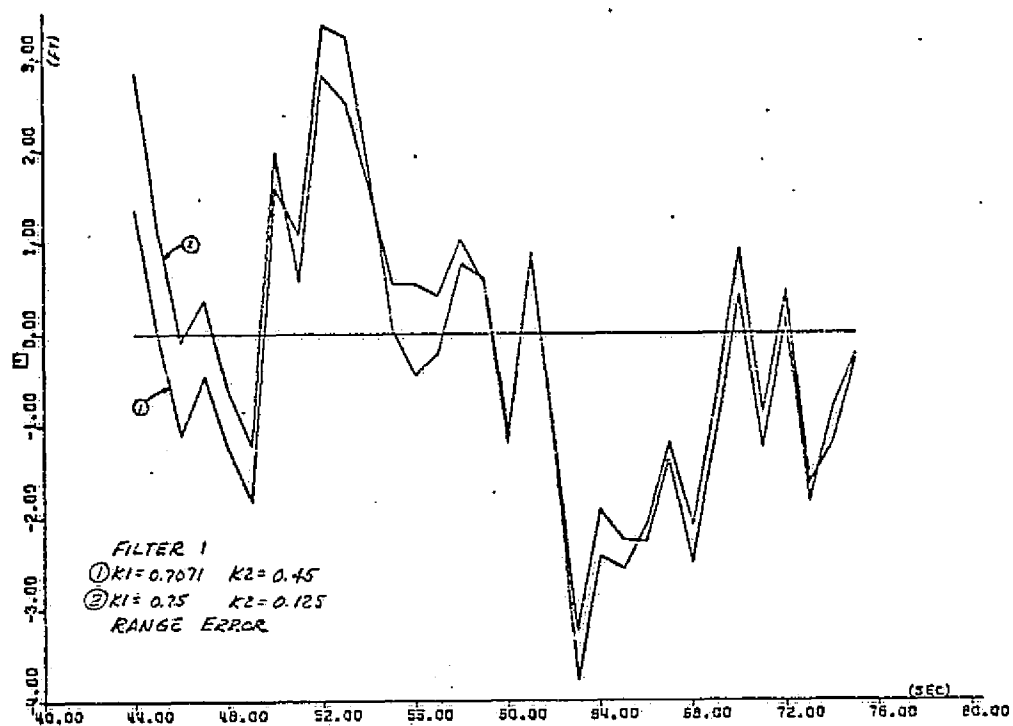


FIGURE 12 - Filter 1 (Constant Gains) at 2500 samples/sec.

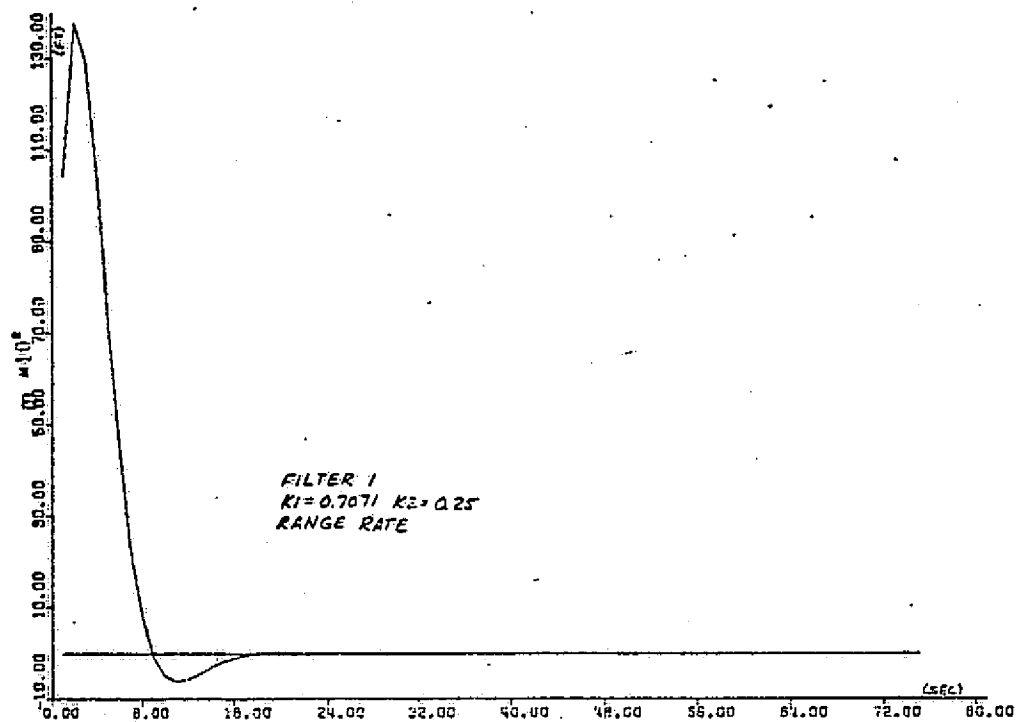
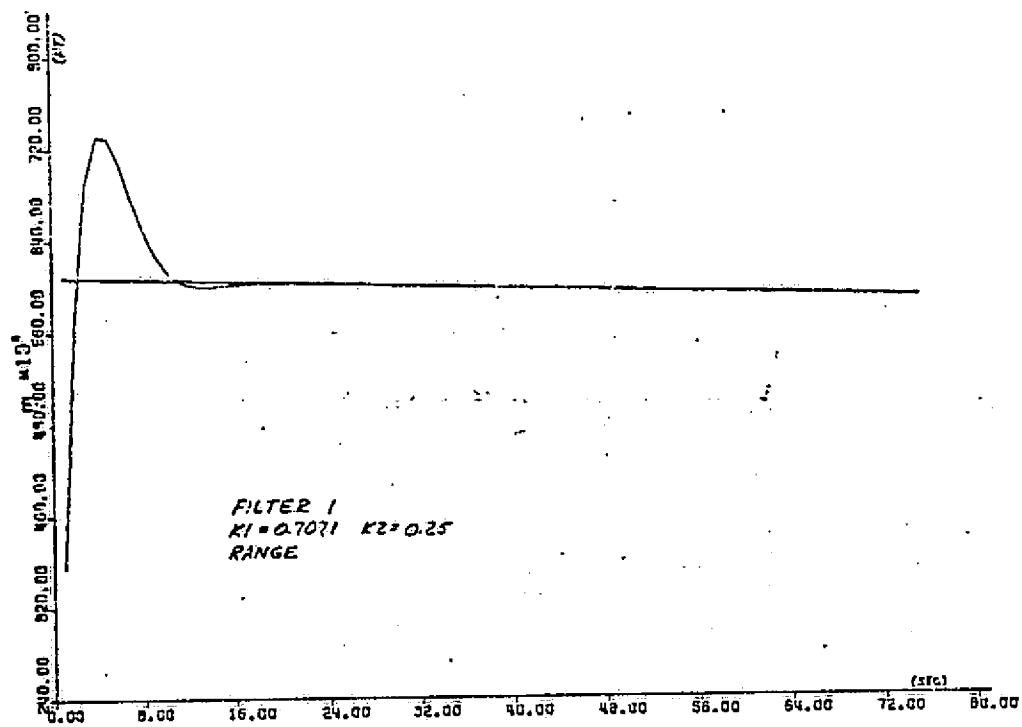


FIGURE 13 - Filter 1 (BLD) at 2500 samples/sec.

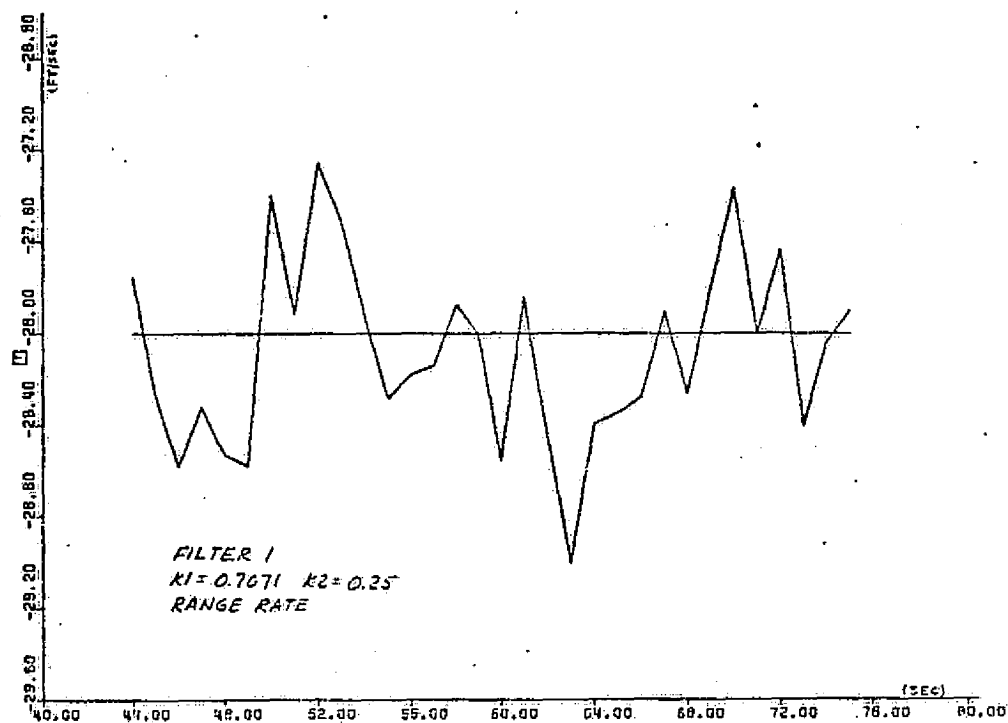
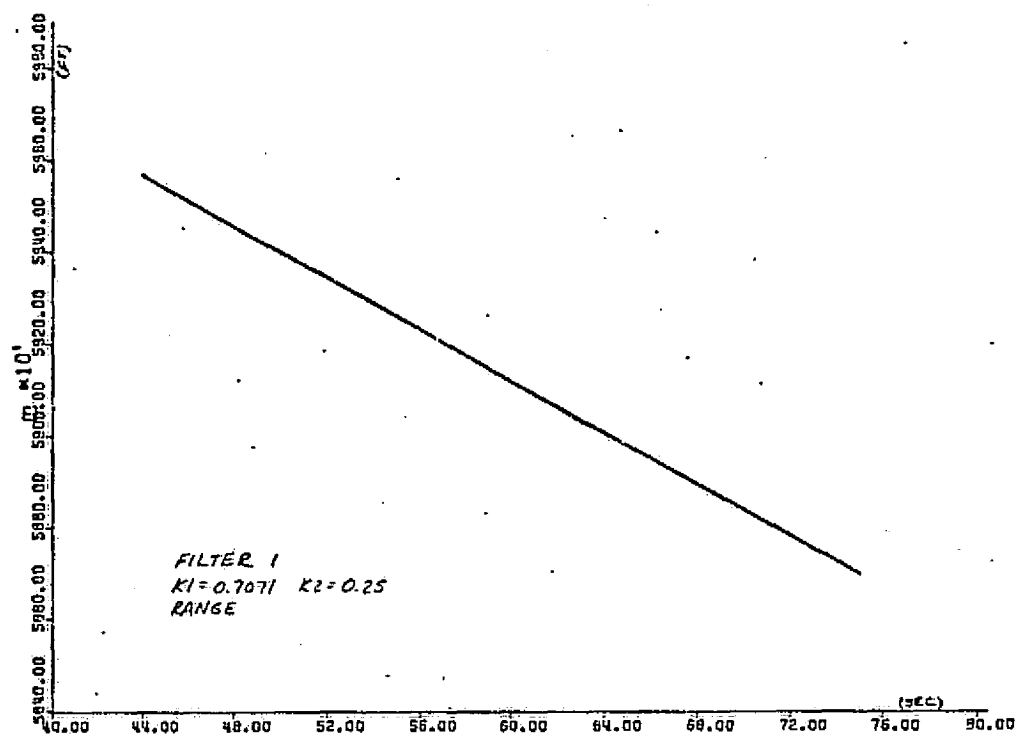


FIGURE 14 - Filter 1 (BLD) at 2500 samples/sec.

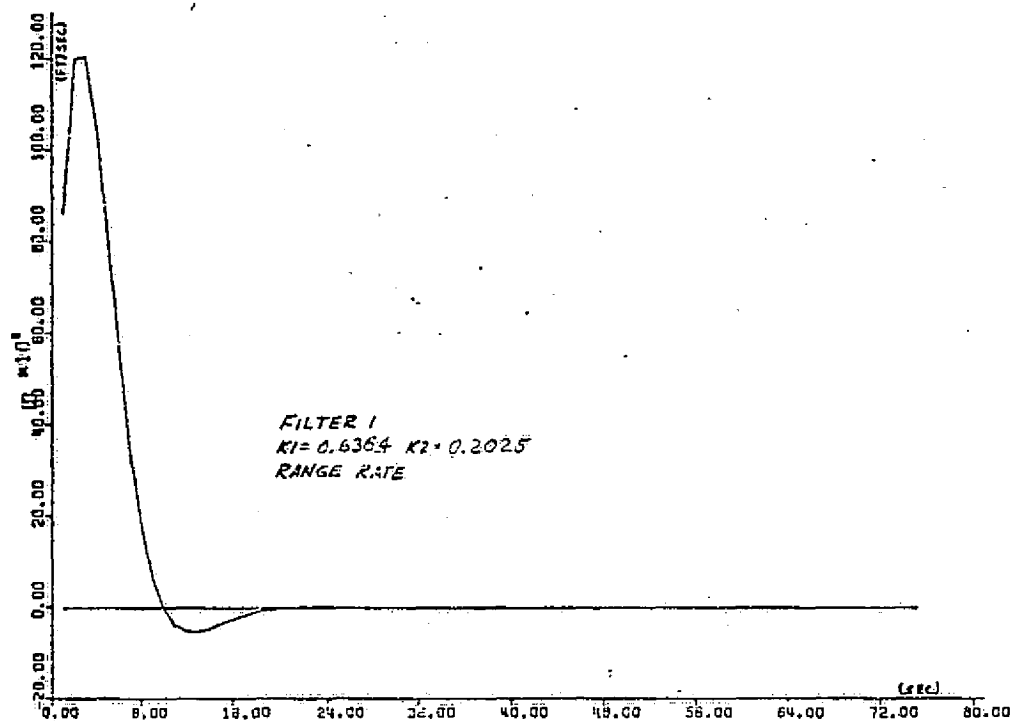
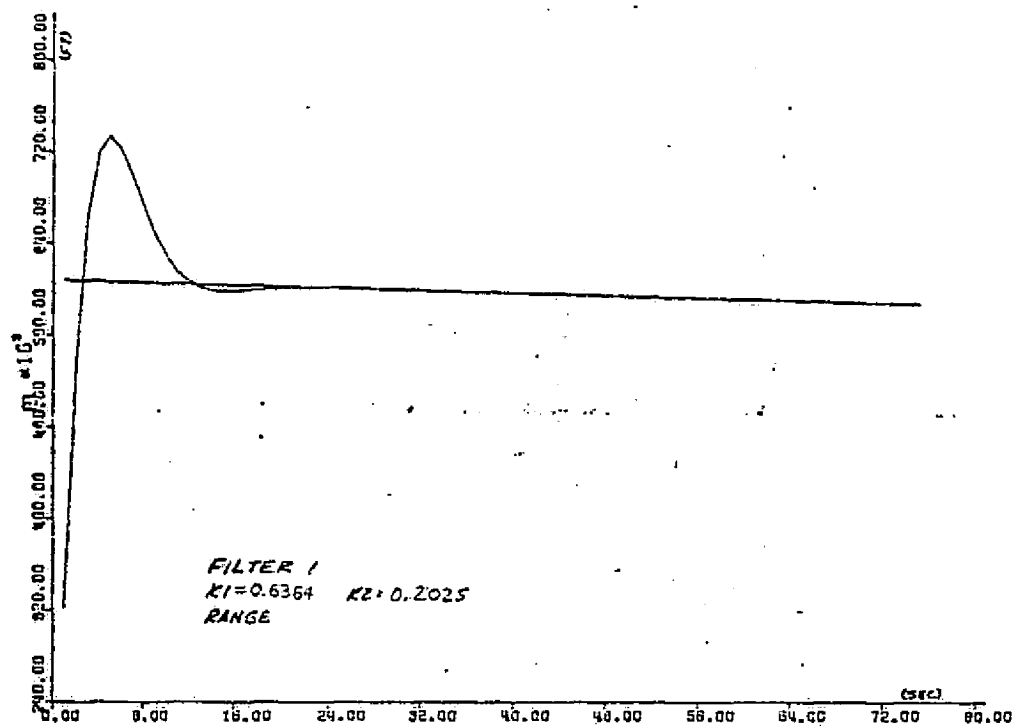


FIGURE 15 - Filter 1 (BLD) at 2500 samples/sec.

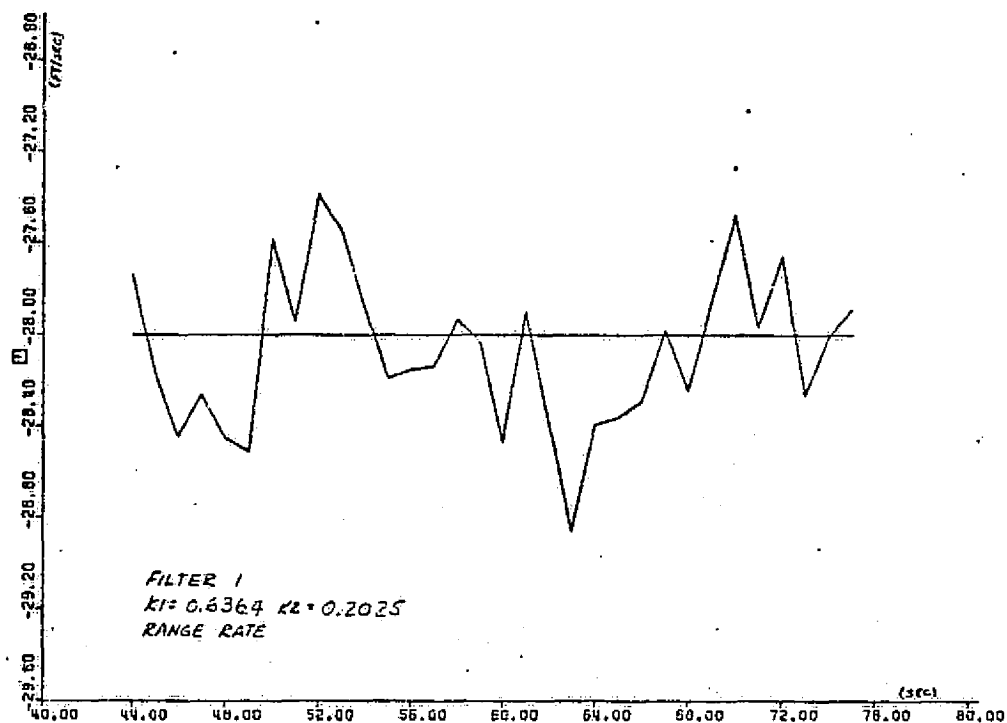
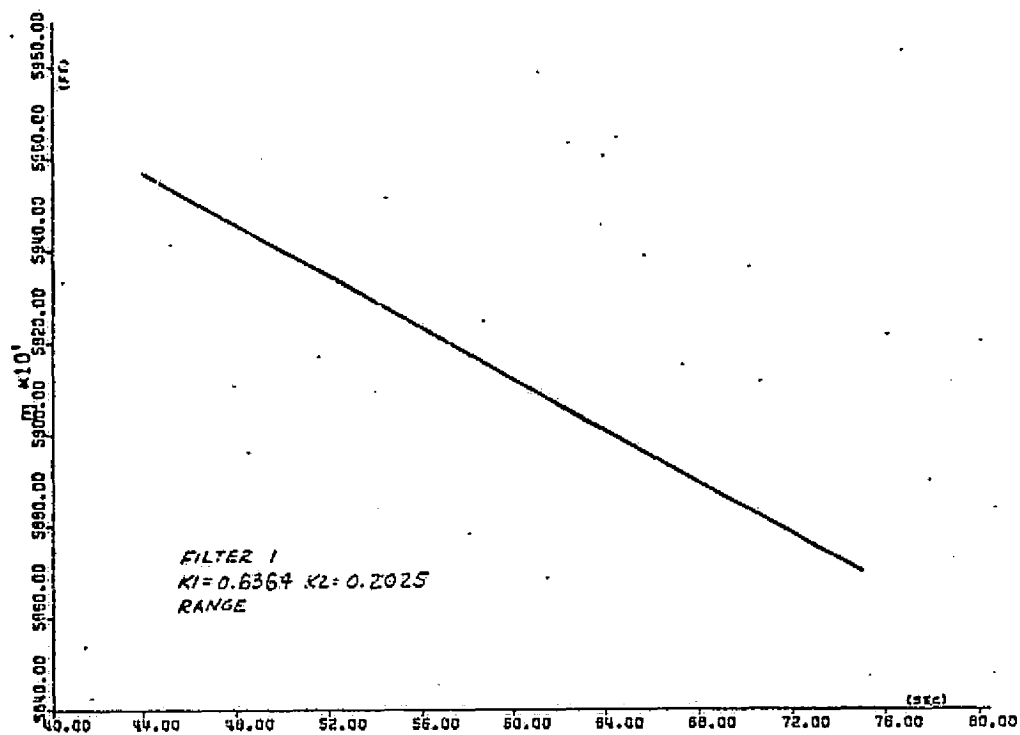


FIGURE 16 - Filter 1 (BLD) at 2500 samples/sec.

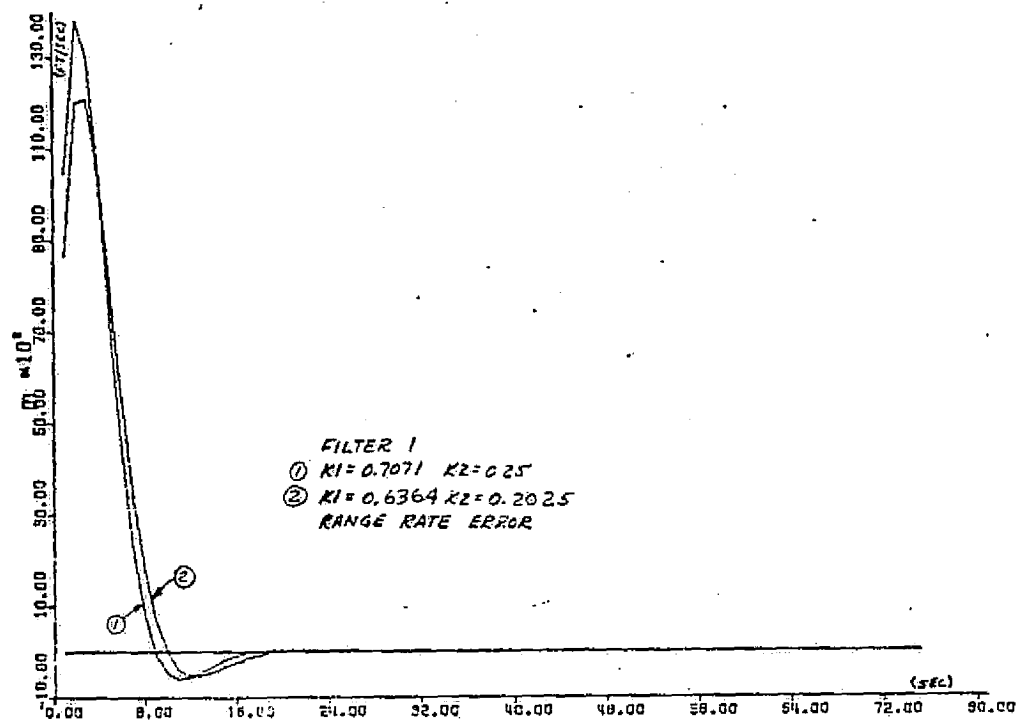
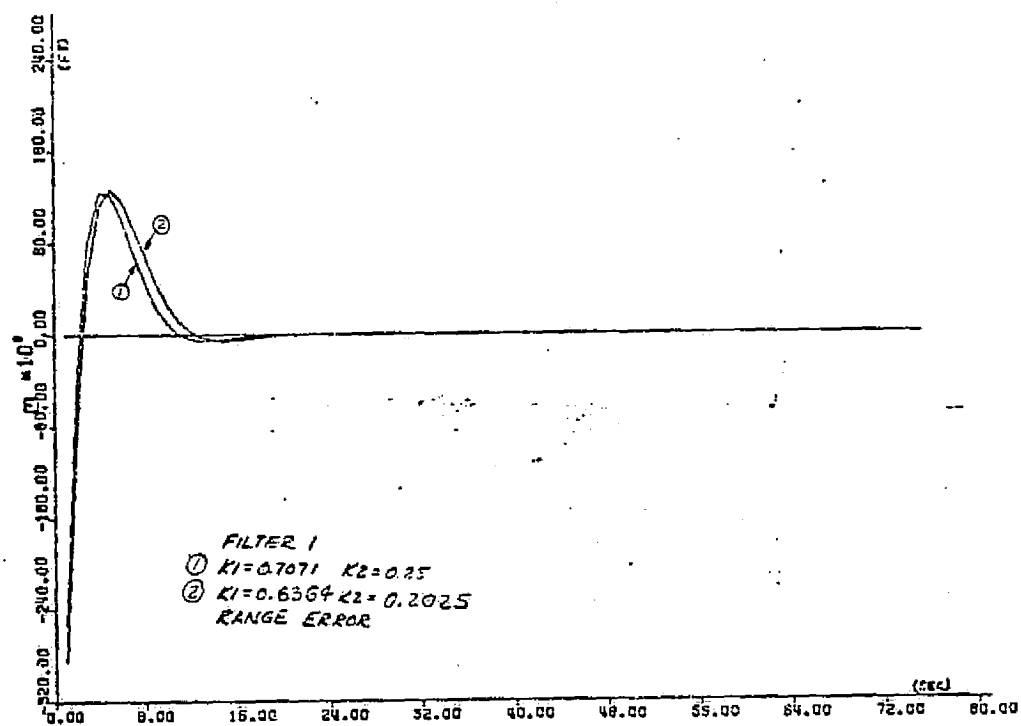


FIGURE 17 - Filter 1 (BLD) at 2500 samples/sec.

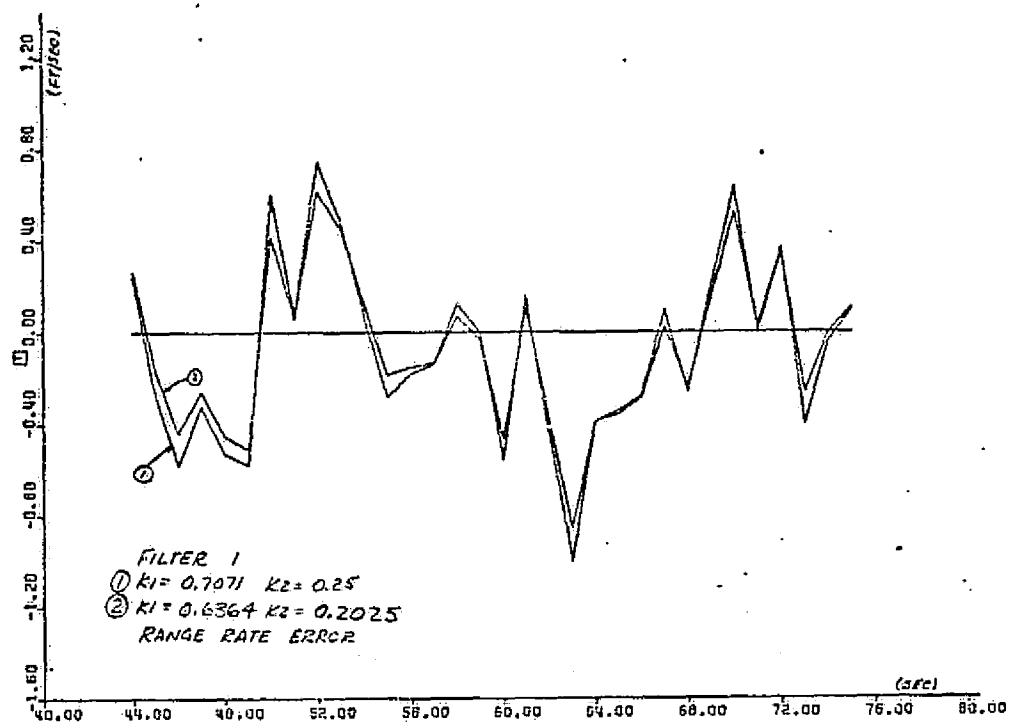
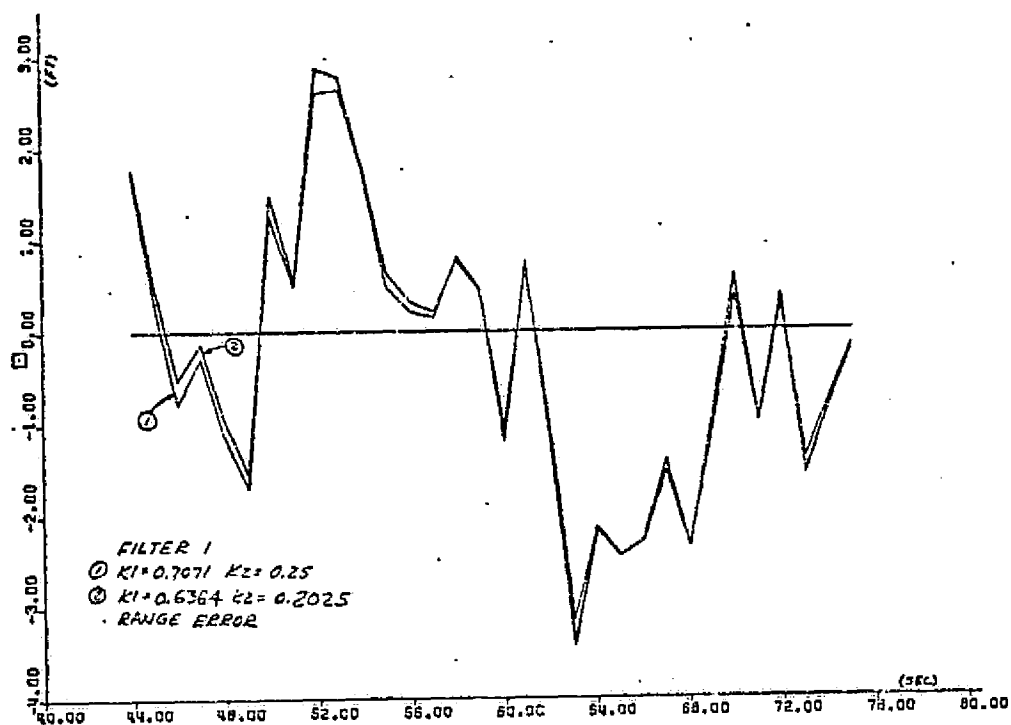


FIGURE 18 - Filter 1 (BLD) at 2500 samples/sec.

TABLE IX

FILTER 1 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
K1 = 0.7071 K2 = 0.45	65.3068	65.3067	91.3215	24.6606	115.2178	84.4420
K1 = 0.75 K2 = 0.125	65.3068	65.3067	91.3215	19.8742	115.2178	78.3689
K1 = 0.7 K2 = 0.3769	65.3068	65.3067	91.3125	23.6573	115.2178	83.4522
K1 = 0.49 K2 = 0.98995	65.3068	65.3067	91.3125	24.2674	115.2178	82.9028
K1 = 0.75 K2 = 0.45	65.3068	65.3067	91.3215	24.8739	115.2178	84.1560

TABLE X

FILTER 1 RANGE RATE STANDARD DEVIATIONS (FT./SEC.) AT 6.25 SAMPLES/SEC.

FILTER 1 CONSTANT GAINS	RUN 1 input = range OUTPUT RANGE RATE STND. DEV.	RUN 2 input = noise OUTPUT RANGE RATE STND. DEV.	RUN 3 input = ramp + noise OUTPUT RANGE RATE STND. DEV.
K1 = 0.7071 K2 = 0.45	0.0	11.4052	11.4052
K1 = 0.75 K2 = 0.125	0.0	3.2672	3.2671
K1 = 0.7 K2 = 0.3769	0.0	9.8217	9.8218
K1 = 0.49 K2 = 0.98995	0.0	24.8694	24.8694
K1 = 0.75 K2 = 0.45	0.0	11.2231	11.2231

TABLE XI

FILTER 1 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 6.25 SAMPLES /SEC.

FILTER 1 CONSTANT GAINS	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
K1 = 0.7071 K2 = 0.45	4.4800	0.102×10^{-11}	52.8718	28.0303	57.3518	28.0303
K1 = 0.75 K2 = 0.125	4.4800	0.126×10^{-11}	44.8687	5.7304	49.3487	5.7304
K1 = 0.7 K2 = 0.3769	4.4800	0.116×10^{-11}	51.5882	22.9018	56.0682	22.9018
K1 = 0.49 K2 = 0.98995	4.4800	0.483×10^{-12}	57.2146	63.0950	61.6946	63.0950
K1 = 0.75 K2 = 0.45	4.4800	0.112×10^{-11}	53.3341	27.4471	57.8141	27.4471

TABLE XII

FILTER 1 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SECOND

FILTER CONSTANT GAINS	RUN 2		T CONVERGENCE TIME (SEC.)
	input = noise		
	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	
K1 = 0.7071 K2 = 0.45	7.292 x 10 ⁻²	1.560 x 10 ⁻²	23
K1 = 0.75 K2 = 0.125	4.736 x 10 ⁻²	1.280 x 10 ⁻³	36
K1 = 0.7 K2 = 0.3769	6.711 x 10 ⁻²	1.157 x 10 ⁻²	25
K1 = 0.49 K2 = 0.98995	7.062 x 10 ⁻²	7.416 x 10 ⁻²	38
K1 = 0.75 K2 = 0.45	7.419 x 10 ⁻²	1.510 x 10 ⁻²	23

TABLE XIII

FILTER 1 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
FILTER 1	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
BLD						
K1 = 0.7778						
K2 = 0.3025	65.3068	65.3067	91.3215	23.1994	115.2178	82.0274
b = 0.55						
K1 = 0.7071						
K2 = 0.25	65.3068	65.3067	91.3215	21.6291	115.2178	81.0868
b = 0.50						
K1 = 0.6364						
K2 = 0.2025	65.3068	65.3067	91.3215	19.9024	115.2178	79.8818
b = 0.45						
K1 = 0.5657						
K2 = 0.160	65.3068	65.3067	91.3215	17.9741	115.2178	78.3595
b = 0.40						
K1 = 0.4950						
K2 = 0.1225	65.3068	65.3067	91.3215	15.7742	115.2178	76.4249
b = 0.35						

TABLE XIV

FILTER 1 RANGE RATE STANDARD DEVIATIONS (FT./SEC.) AT 6.25 SAMPLES/SEC.

FILTER 1 BLD	RUN 1	RUN 2	RUN 3
	input = ramp OUTPUT RANGE RATE STND. DEV.	input = noise OUTPUT RANGE RATE STND. DEV.	input = ramp + noise OUTPUT RANGE RATE STND. DEV.
K1 = 0.7778 K2 = 0.3025 b = 0.55	0.0	7.7822	7.7822
K1 = 0.7071 K2 = 0.25 b = 0.50	0.0	6.7482	6.7482
K1 = 0.6364 K2 = 0.2025 b = 0.45	0.0	5.7662	5.7662
K1 = 0.5657 K2 = 0.160 b = 0.40	0.0	4.8206	4.8206
K1 = 0.4950 K2 = 0.1225 b = 0.35	0.0	3.8874	3.8874

TABLE XV

FILTER 1 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 6.25 SAMPLES/SEC.

FILTER 1 BLD	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
K1 = 0.7778 K2 = 0.3025 b = 0.65	4.480	0.209×10^{-11}	50.988	16.971	55.468	16.971
K1 = 0.7071 K2 = 0.25 b = 0.50	4.480	0.245×10^{-11}	48.114	13.780	52.594	13.780
K1 = 0.6364 K2 = 0.2125 b = 0.45	4.480	0.270×10^{-11}	44.710	10.906	49.190	10.906
K1 = 0.5657 K2 = 0.160 b = 0.40	4.480	0.329×10^{-11}	40.688	8.570	45.168	8.570
K1 = 0.4950 K2 = 0.1225 b = 0.35	4.480	0.330×10^{-11}	36.010	6.730	40.490	6.730

TABLE XVI

FILTER 1 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

	RUN 2		
	input = noise		
FILTER 1	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	τ CONVERGENCE TIME (SEC.)
BLD			
K1 = 0.7778 K2 = 0.3025 b = 0.55	6.454×10^{-2}	7.262×10^{-3}	23
K1 = 0.7071 K2 = 0.25 b = 0.50	5.610×10^{-2}	5.460×10^{-3}	26
K1 = 0.6364 K2 = 0.2025 b = 0.45	4.750×10^{-2}	3.987×10^{-3}	28
K1 = 0.5657 K2 = 0.160 b = 0.40	3.874×10^{-2}	2.770×10^{-3}	31
K1 = 0.4950 K2 = 0.1225 b = 0.35	2.984×10^{-2}	1.812×10^{-3}	36

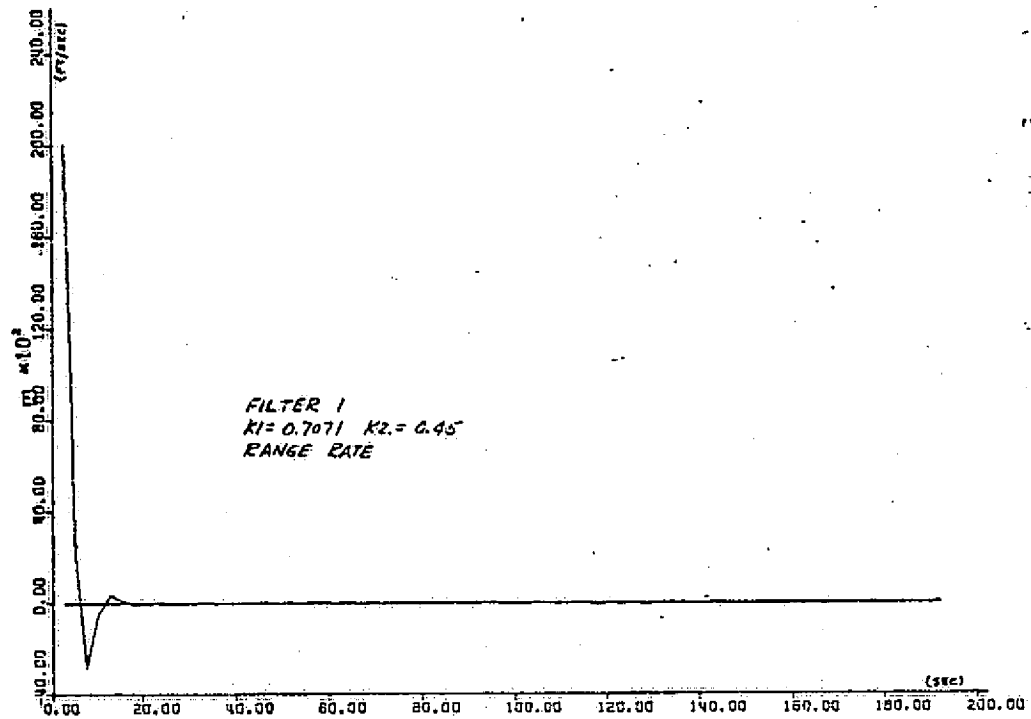
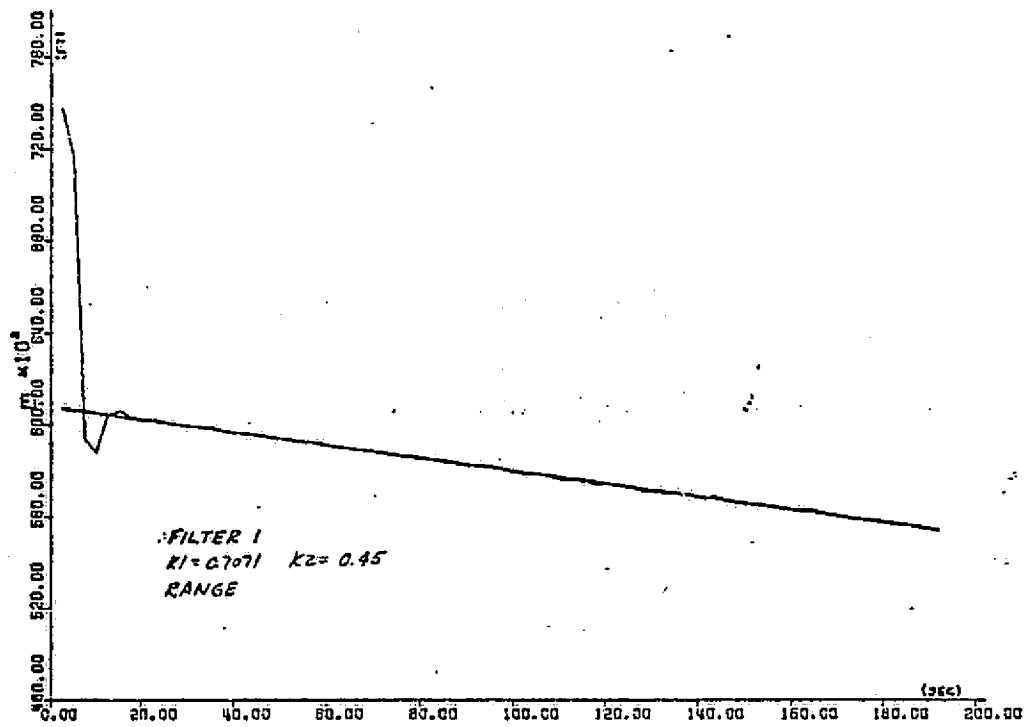


FIGURE 19 - Filter1 (Constant Gains) at 6.25 samples/sec.

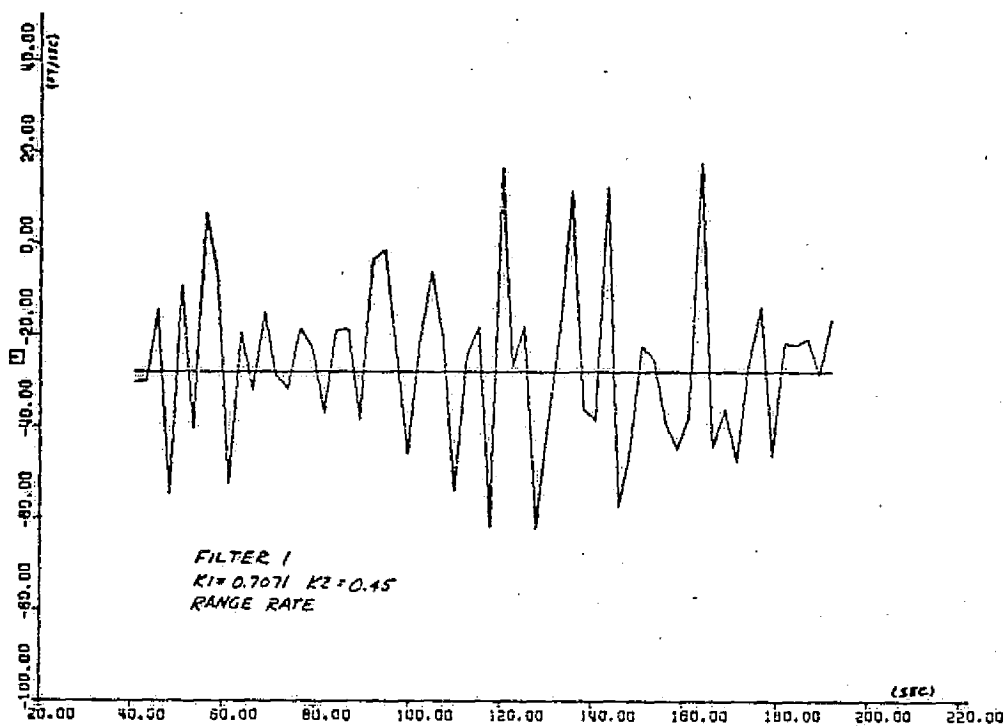
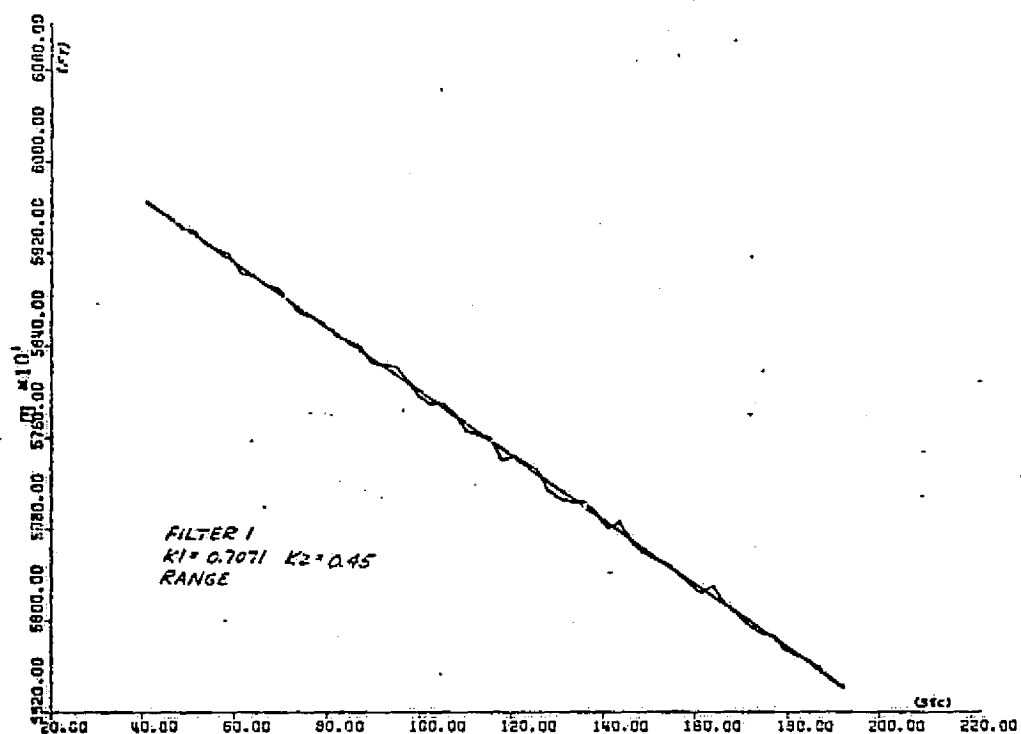


FIGURE 20 - Filter 1 (Constant Gains) at 6.25 samples/sec.

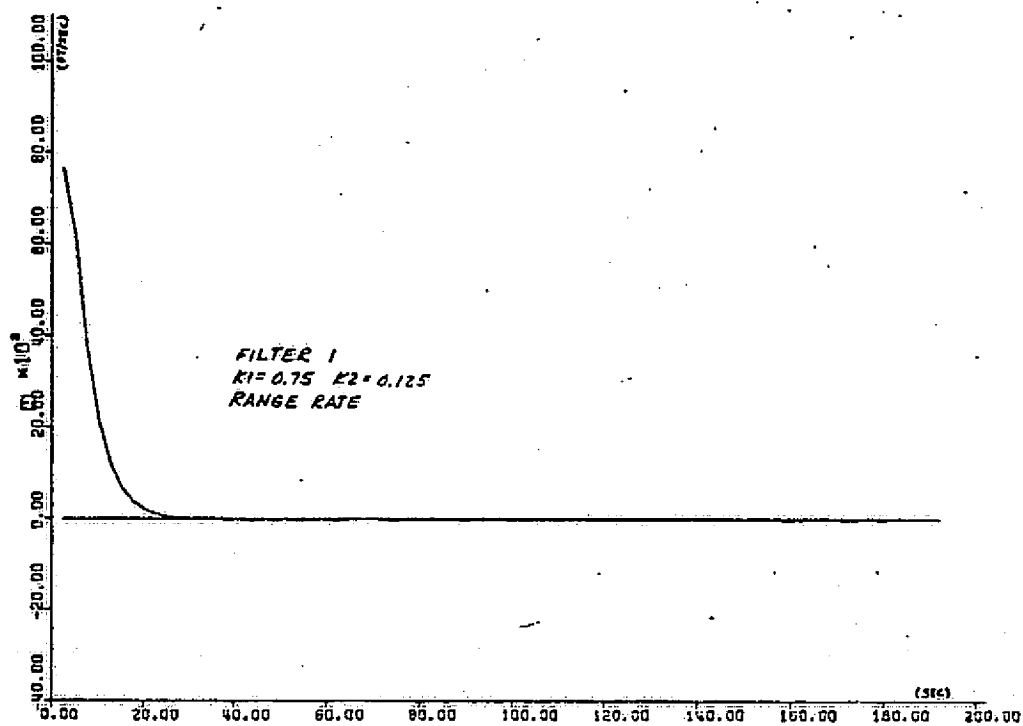
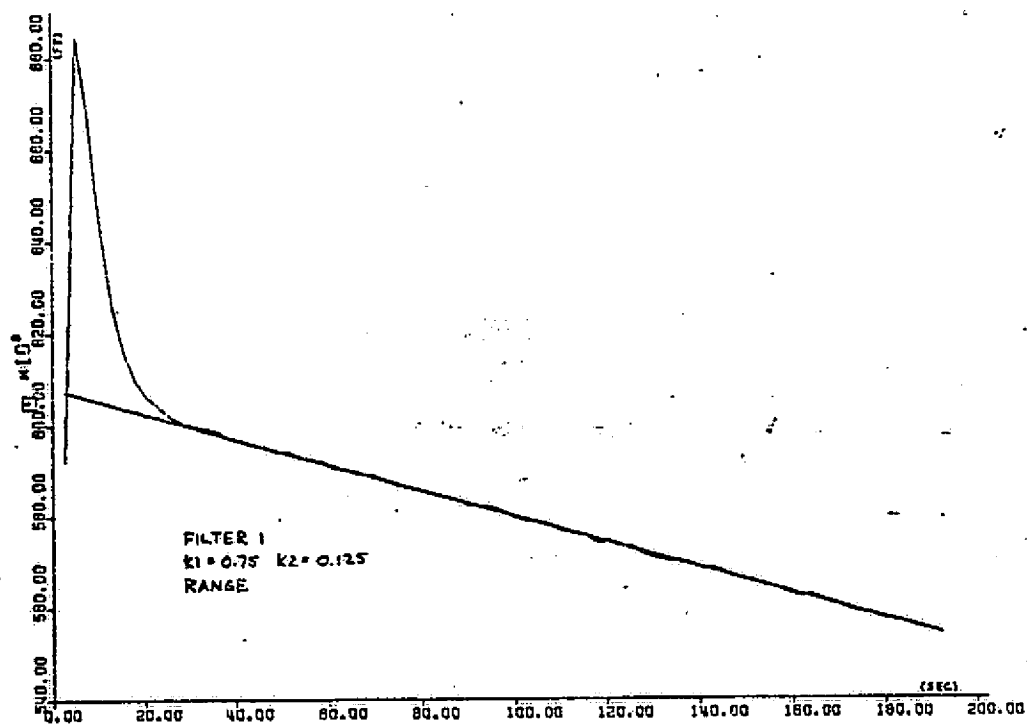


FIGURE 21 - Filter 1 (Constant Gains) at 6.25 samples/sec.

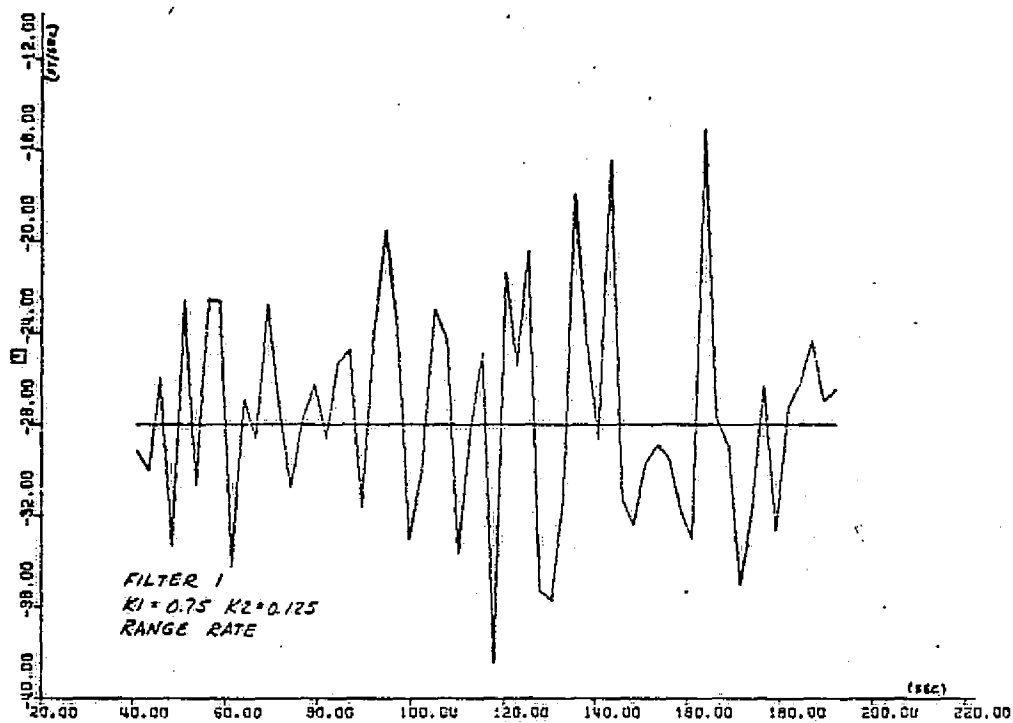
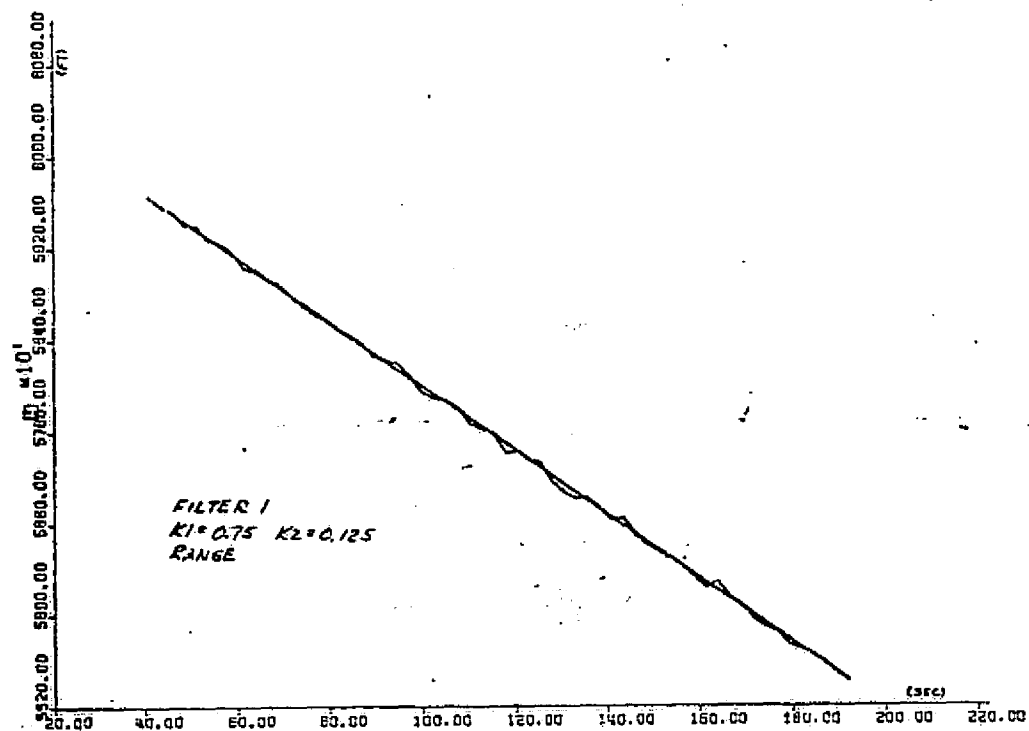


FIGURE 22 - Filter 1 (Constant Gains) at 6.25 samples/sec.

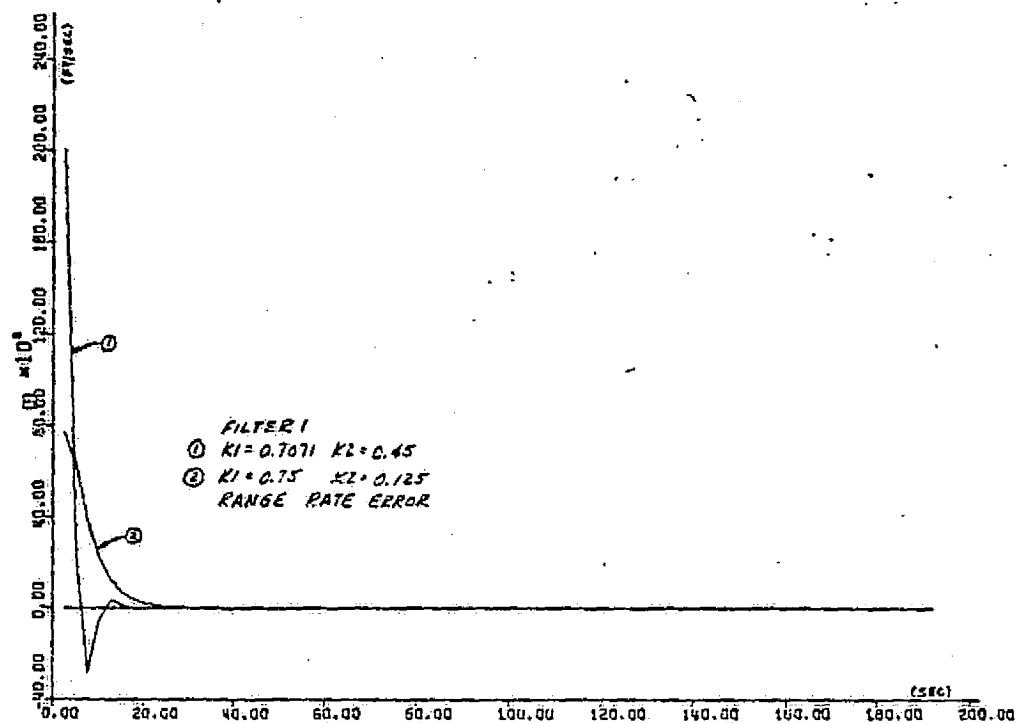
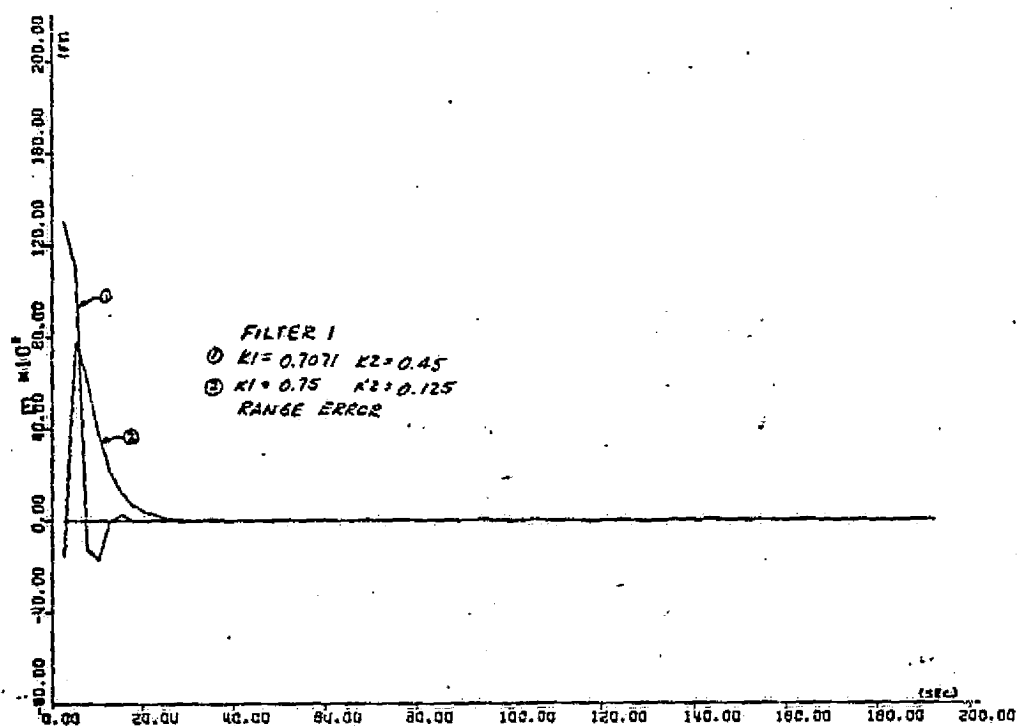


FIGURE 23 - Filter 1 (Constant Gains) at 6.25 samples/sec.

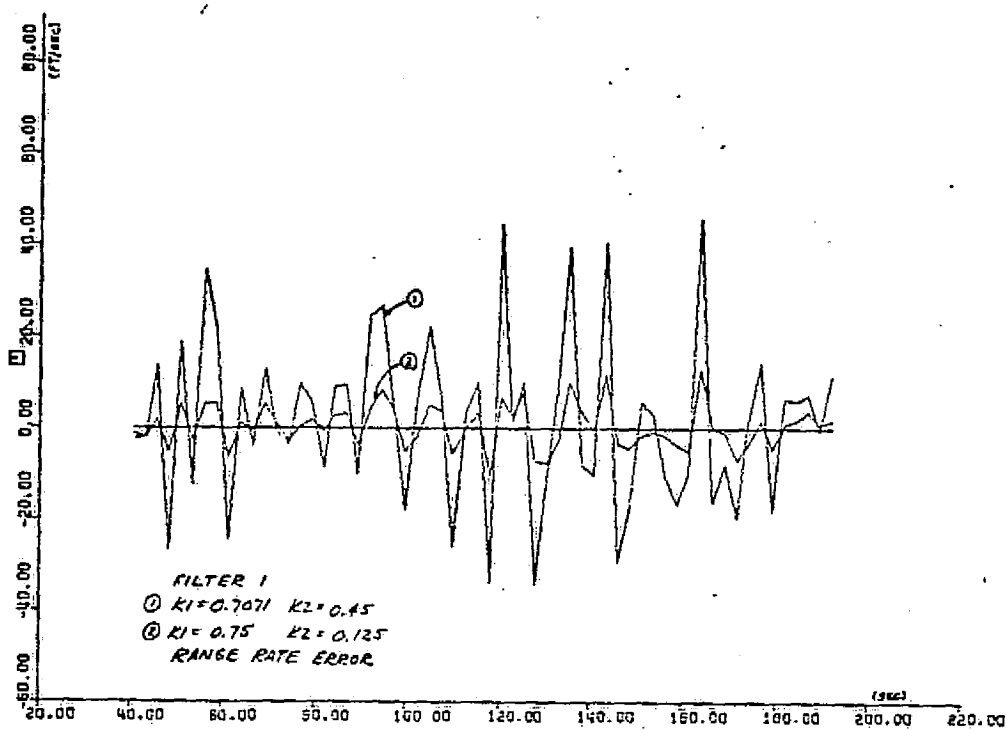
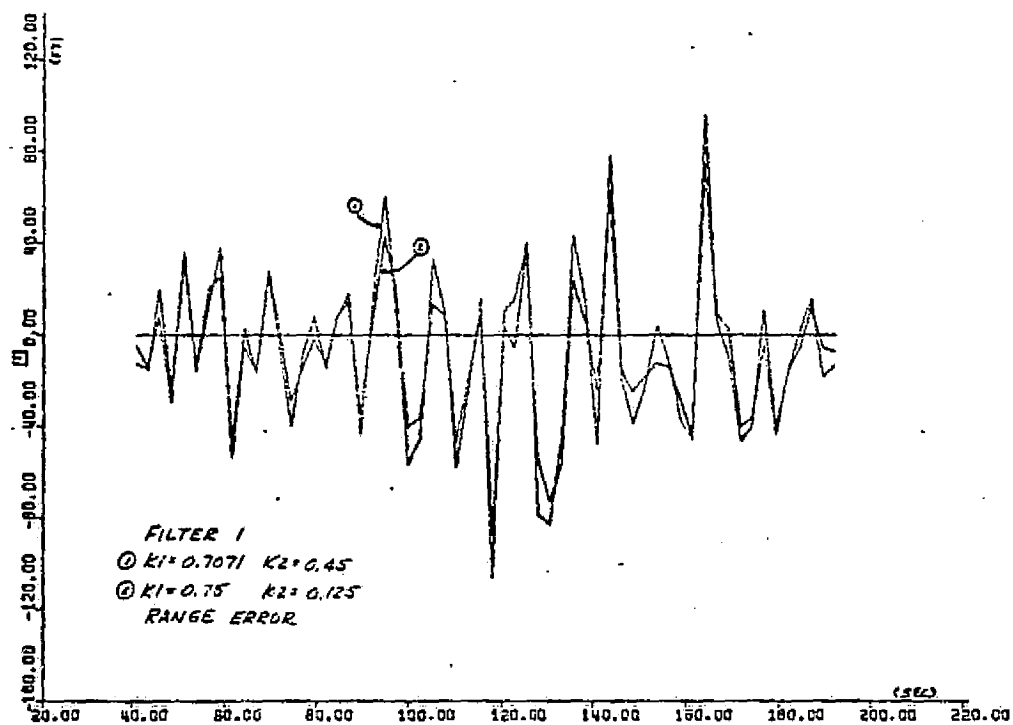


FIGURE 24 - Filter 1 (Constant Gains) at 6.25 samples/sec.

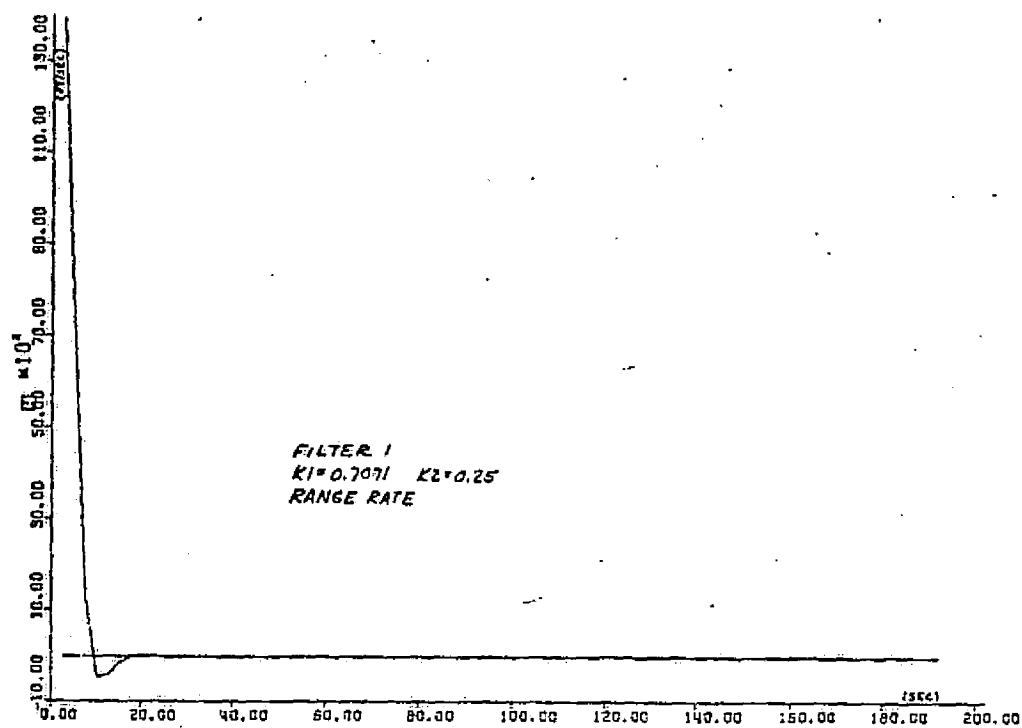
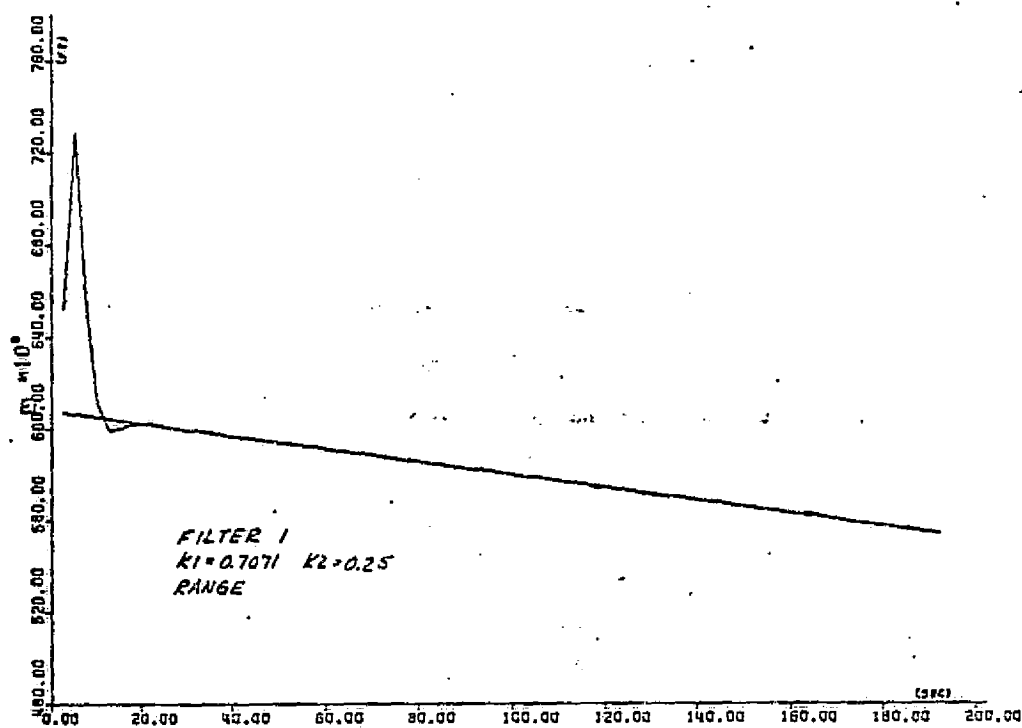


FIGURE 25 - Filter 1 (BLD) at 6.25 samples/sec.

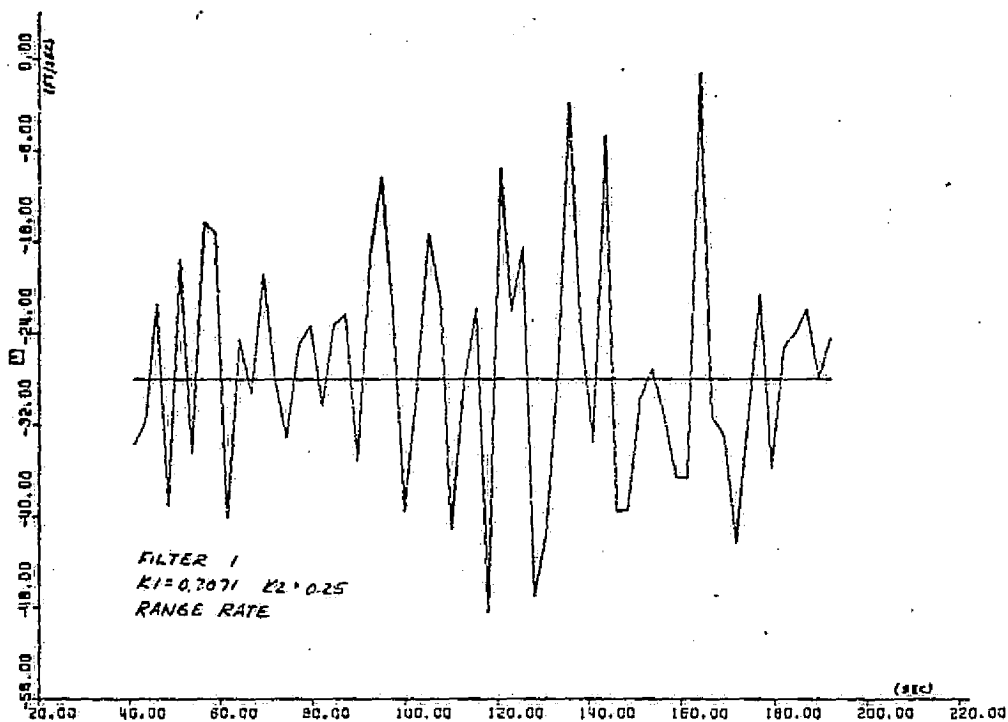
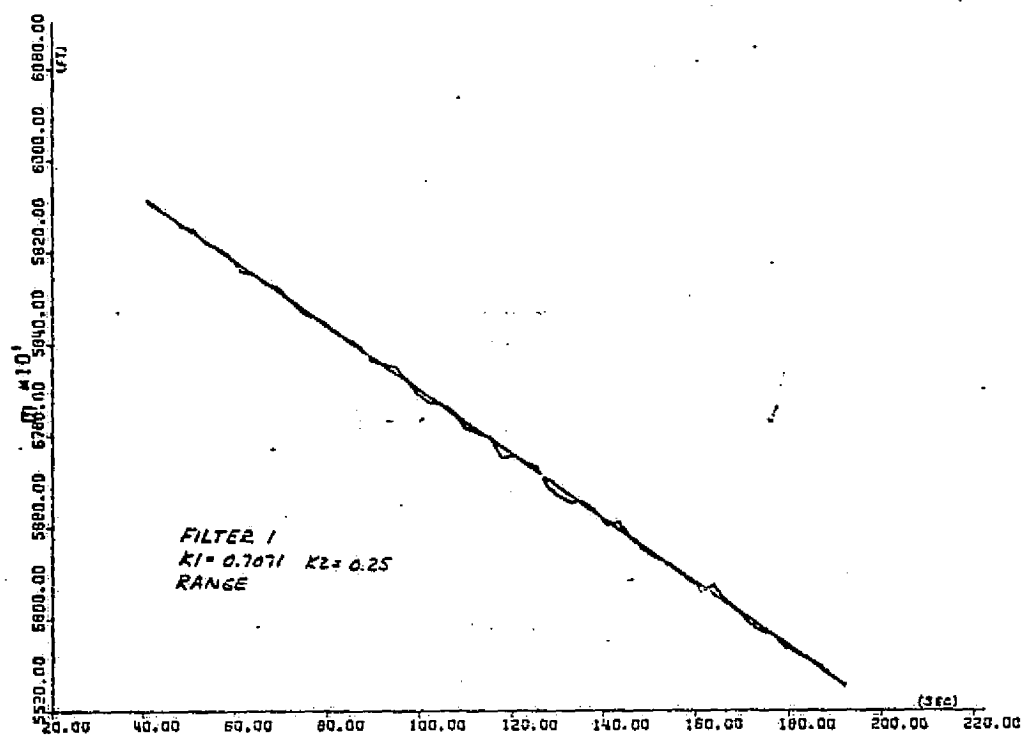


FIGURE 26 - Filter 1 (BLD) at 6.25 samples/sec.

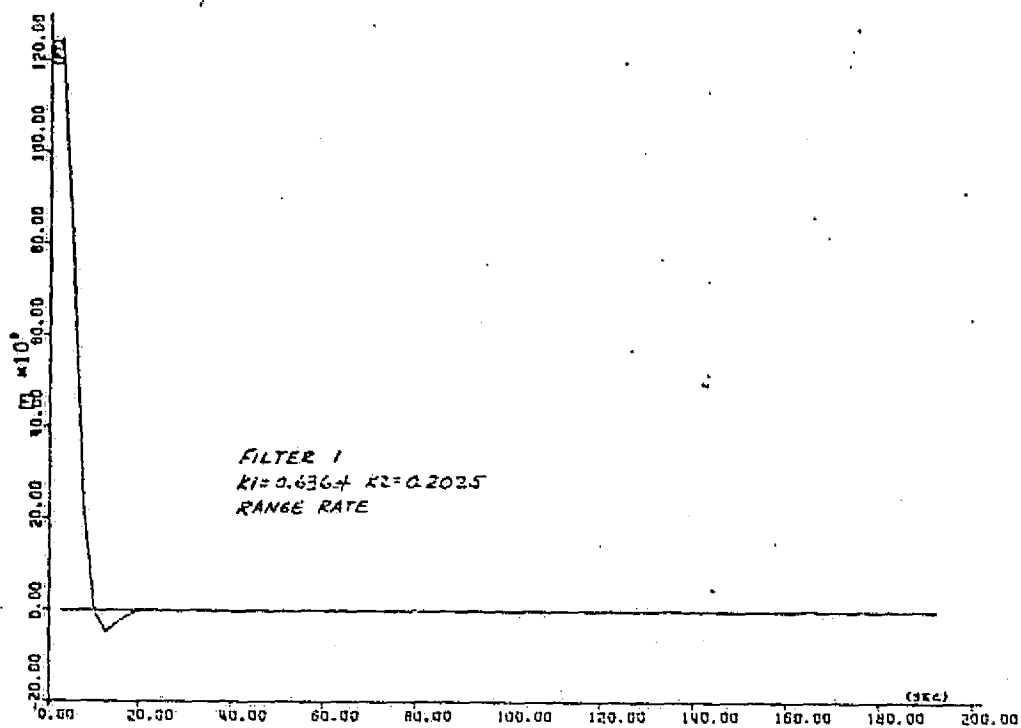
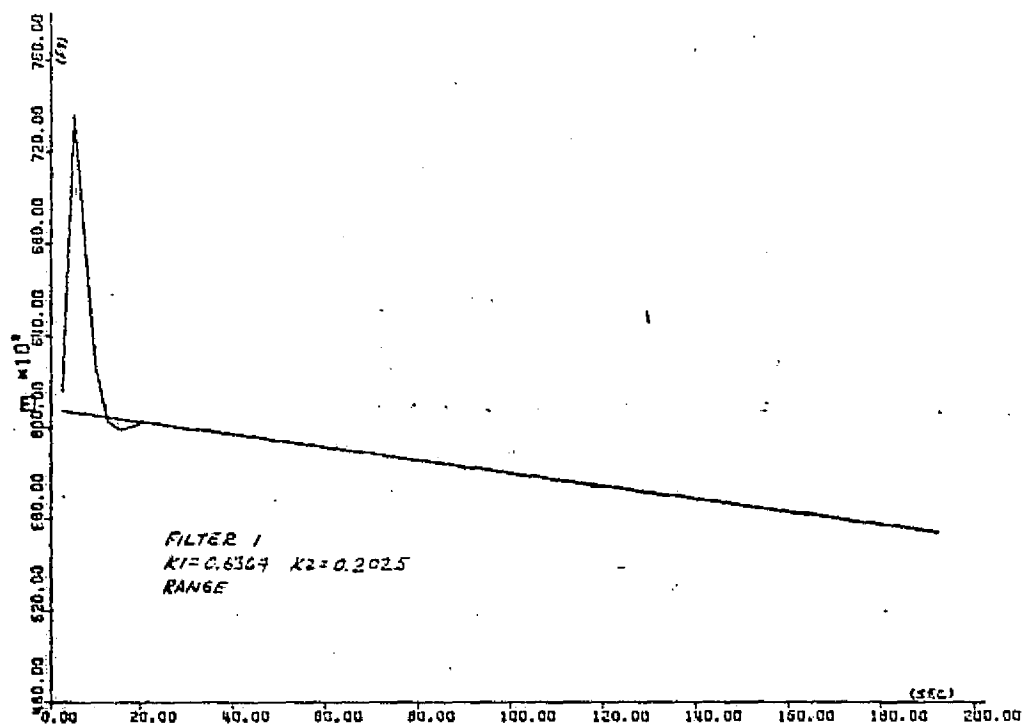


FIGURE 27 - Filter 1 (BLD) at 6.25 samples/sec.

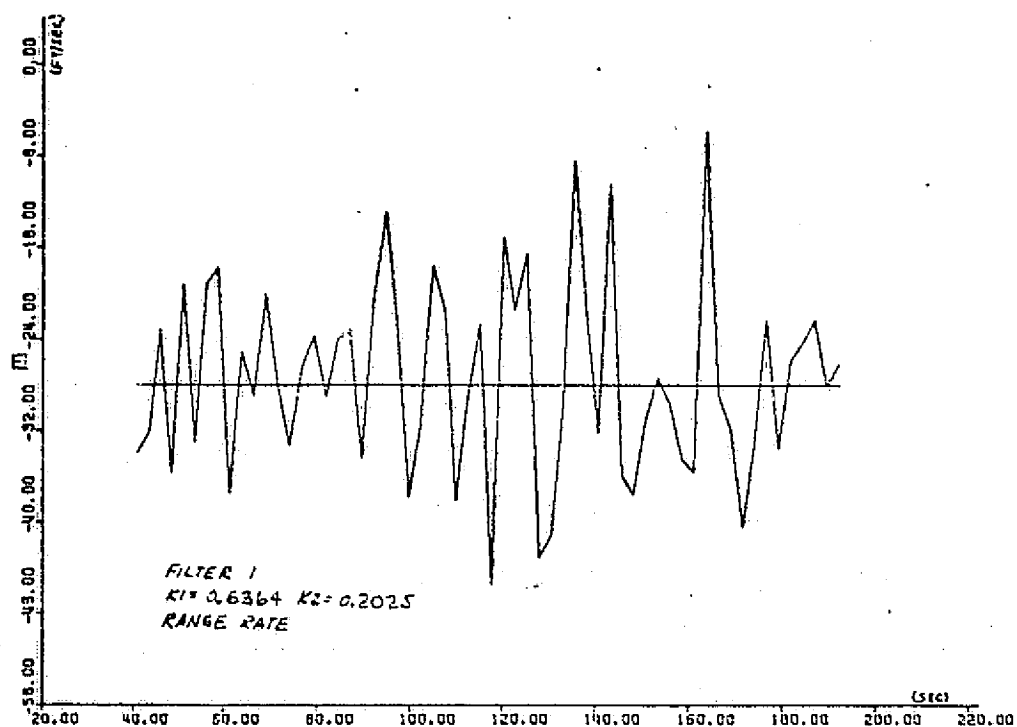
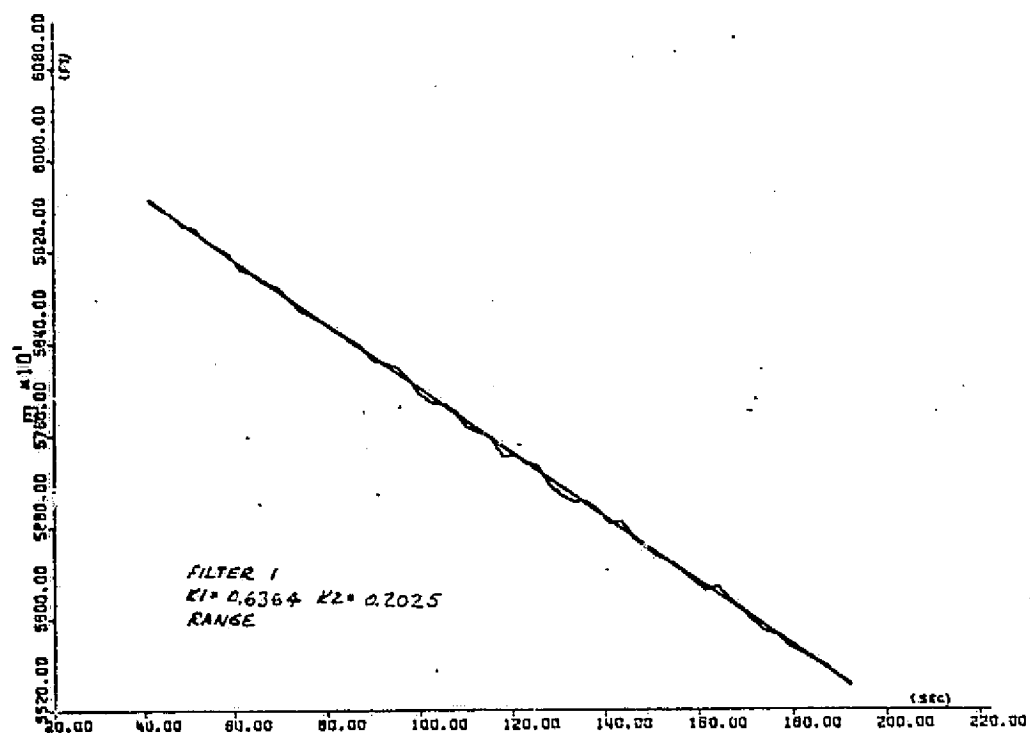


FIGURE 28 - Filter 1 (BLD) at 6.25 samples/sec.

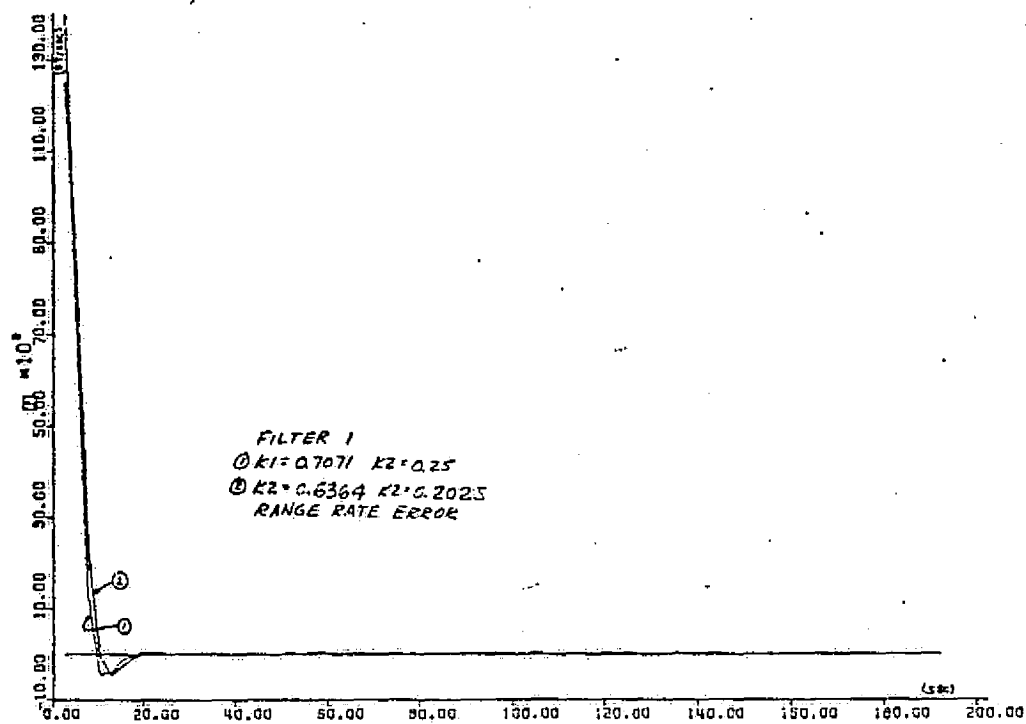
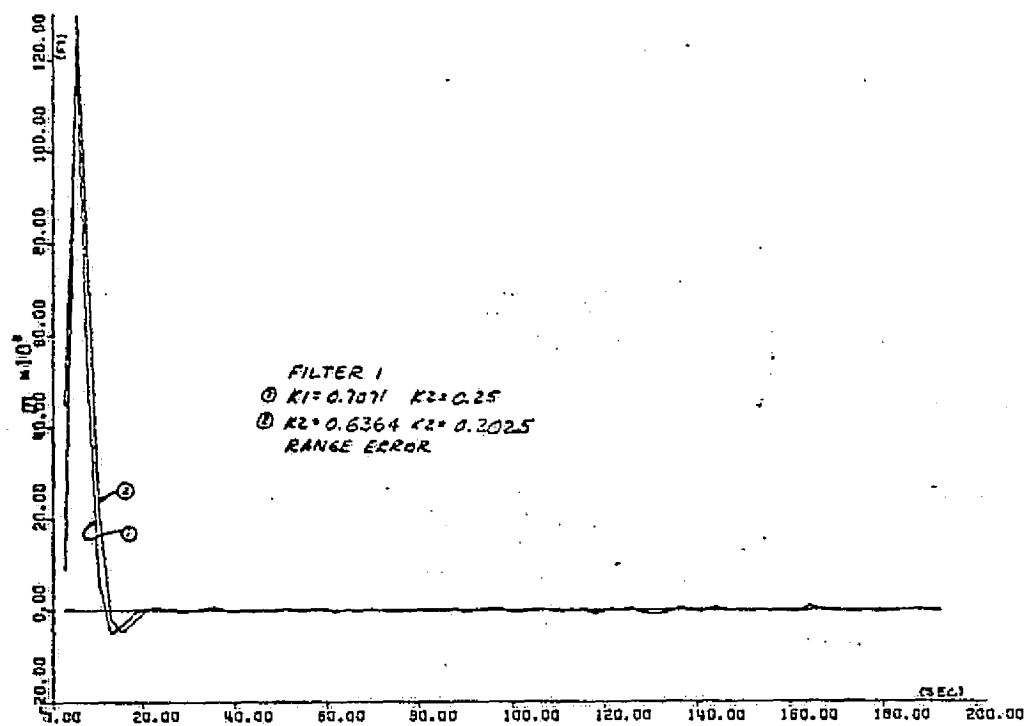


FIGURE 29 - Filter 1 (BLD) at 6.25 samples/sec.

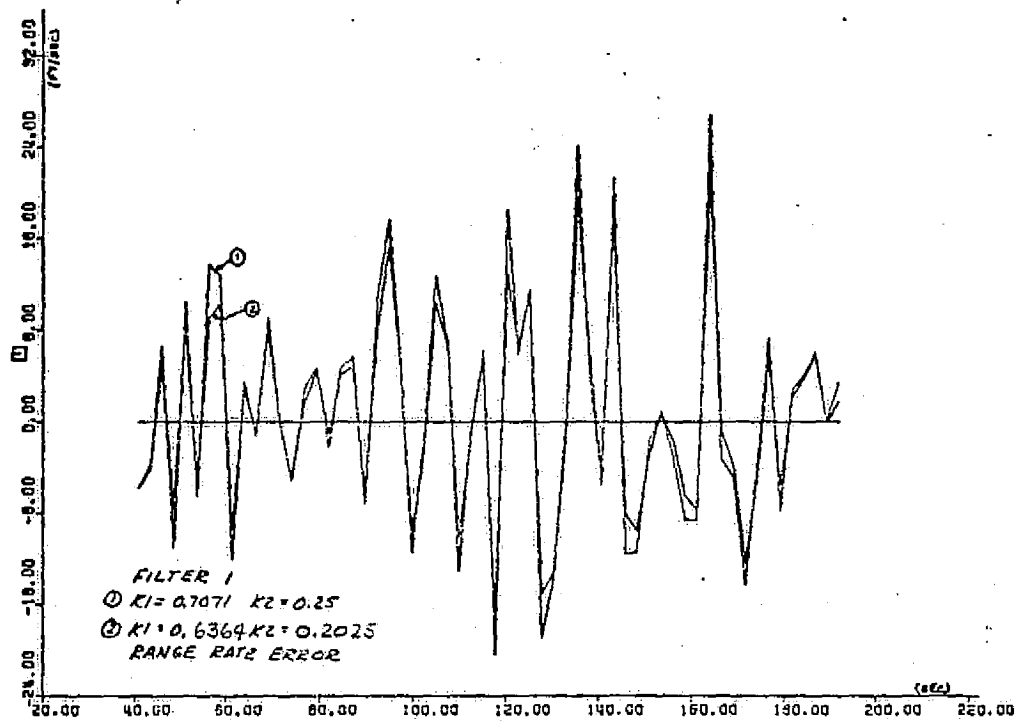
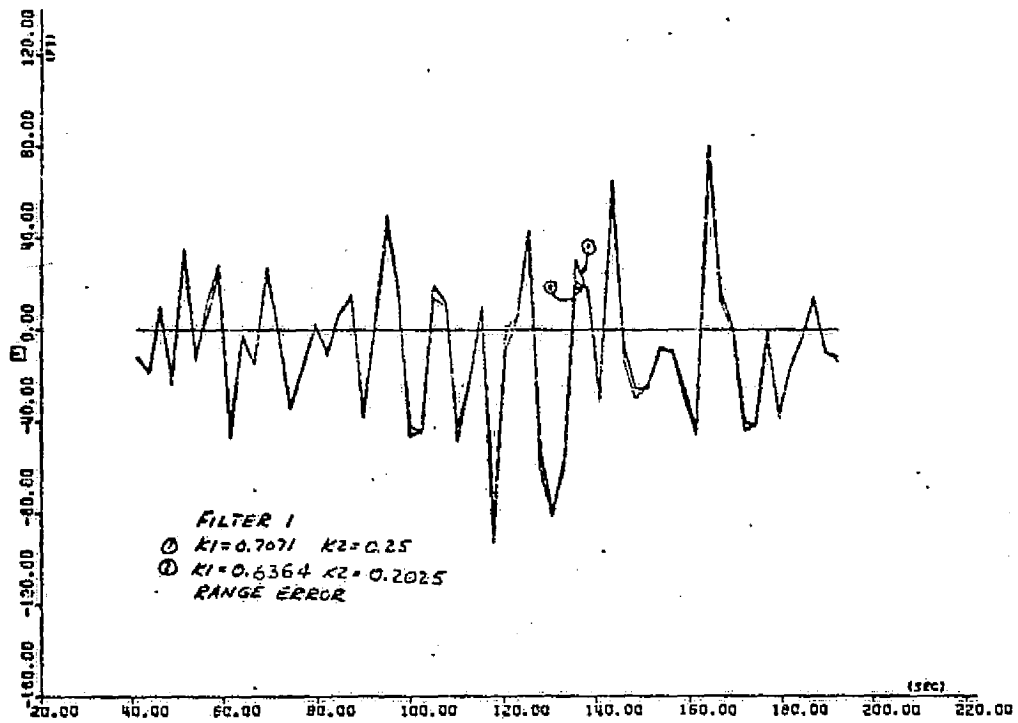


FIGURE 30 - Filter 1 (BLD) at 6.25 samples/sec.

RUN 1. With the ramp input, the recorded range error term represents the change in range over one sampling period. This can be corrected by multiplying the numerator of the transfer function H_R (equation (3)) by the delay term z^{-1} . This has the effect of shifting the system back in time one sampling period. The resulting transfer function is

$$H_R = \frac{RE}{RM} = \frac{z^{-1}(K_1DT + K_2DT^2) - z^{-2}(K_1DT)}{1 + z^{-1}(-2 + K_1DT + K_2DT^2) + z^{-2}(1 - K_1DT)}. \quad (21)$$

2. Filter 2

Simulation data for Filter 2 is presented in Tables XVII - XX for 2500 samples/second and in Tables XXI - XXIV for the 6.25 samples/second sampling rate. A graphical display of the data is shown for each set of parameter pairs α and β at 2500 samples/second in Figures 31-44. A graphical display of the data for the 6.25 samples/second sampling rate is shown in Figures 45-53. Graphs for two sets of parameter pairs are omitted because they yield systems whose convergence times exceed the duration of the simulation run.

The alpha-beta filter is simulated using six sets of α, β pairs that lie on the curve described by (10) within the region of stability defined by:

$$\alpha > 0, \beta > 0 \text{ and } 2\alpha + \beta < 4. \quad (22)$$

Near the upper end of the curve, at the point $\alpha = 1.0$ and $\beta = 1.0$, the filter is characterized by fast maneuver-following ability but poor noise rejection. Conversely, near the lower end of the curve, as α and β approach zero, the filter is characterized by improved noise rejection at the expense of maneuver-following ability.

For the 2500 samples/second tests, relative maneuver-following capabilities can be determined by examining the convergence time. This is the time required by the filter to recover from the transient at the start of the simulation. For larger values of α (such as $\alpha = 1.0$), convergence time is negligible. However, the range variance reduction ratio (KR) is one, meaning there is no noise attenuation. As α and β are decreased, convergence time increases indicating the inertia of the system is increasing and more time is required to recover from transients. While this presents a reduction in maneuver-following capabilities it also means the system is becoming more insensitive to the higher frequency components of noise. Hence, there is a corresponding decrease in range and range rate variance reduction ratios.

These observations are even more readily seen in the 6.25 samples/second simulation runs. The reduction in sampling rate causes an increase in convergence time. As an example, for $\alpha = 0.05$ and $\beta = 1.282 \times 10^{-2}$, the convergence time at 2500 samples/second is less than one second. However, at 6.25 samples/second that figure is approximately 56 seconds. In fact, the last three entries in the tables for the 6.25 samples/second tests all represent α, β pairs whose convergence times exceed the 192

TABLE XVII

FILTER 2 RANGE STANDARD DEVIATIONS (FT.) AT 2500 SAMPLES/SEC.

FILTER 2 alpha - beta	RUN 1 input = ramp		RUN 2 input = noise		RUN 2 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
$\alpha = 1.0$ $\beta = 1.0$	0.1634	0.1634	110.7364	110.7364	110.7339	110.7339
$\alpha = 0.2$ $\beta = 0.022222$	0.1634	0.1634	110.7364	29.0844	110.7339	29.0377
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	0.1634	0.1634	110.7364	7.5559	110.7339	7.6332
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	0.1634	0.1634	110.7364	0.3205	110.7339	0.4751
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	0.1634	0.1634	110.7364	0.9379×10^{-1}	110.7339	0.2518
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	0.1634	0.1632	110.7364	0.6538×10^{-1}	110.7339	0.225

TABLE XVIII

FILTER 2 RANGE RATE STANDARD DEVIATIONS (FT./SEC.) AT 2500 SAMPLES/SEC.

FILTER 2 alpha - beta	RUN 1 input = ramp	RUN 2 input = noise	RUN 3 input = ramp + noise
	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.
$\alpha = 1.0$ $\beta = 1.0$	0.0	412275.80	412275.81
$\alpha = 0.2$ $\beta = 0.0222222$	0.0	8741.49	8741.49
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	0.0	445.064	445.0637
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	0.0	0.3965	0.3965
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	0.0	0.3550×10^{-1}	0.355×10^{-1}
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	0.0	0.1739×10^{-1}	0.1739×10^{-1}

TABLE XIX

FILTER 2 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

FILTER 2 α - β	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
$\alpha = 1.0$ $\beta = 1.0$	0.0	0.578×10^{-9}	254.932	989103.5	254.932	989103.5
$\alpha = 0.2$ $\beta = 0.022222$	0.0	0.226×10^{-10}	96.237	21226.7	96.237	21226.7
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	0.909×10^{-12}	0.476×10^{-9}	37.824	1792.05	37.824	1792.05
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	0.546×10^{-10}	0.166×10^{-8}	0.780	1.409	0.780	1.409
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	0.425×10^{-7}	0.414×10^{-8}	0.199	0.236	0.199	0.236
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	0.134×10^{-4}	0.670×10^{-4}	0.133	0.194	0.133	0.194

TABLE XX

FILTER 2 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

FILTER 2 alpha - beta	RUN 2 input = noise		PREDICTED BY BENEDICT AND BORDNER		τ CONVERGENCE TIME (SEC.)
	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	KR	KRR	
$\alpha = 1.0$ $\beta = 1.0$	1.0	1.386×10^7	1.0	1.25×10^7	<1
$\alpha = 0.2$ $\beta = 0.022222$	6.898×10^{-2}	6.231×10^3	1.534×10^{-1}	8.521×10^3	<1
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	4.656×10^{-3}	1.615×10^1	3.782×10^{-2}	1.056×10^2	<1
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	8.377×10^{-6}	1.282×10^{-5}	7.501×10^{-4}	7.824×10^{-4}	8
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	7.174×10^{-7}	1.028×10^{-7}	2.250×10^{-4}	2.110×10^{-5}	24
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	3.486×10^{-7}	2.466×10^{-8}	1.575×10^{-4}	7.237×10^{-6}	35

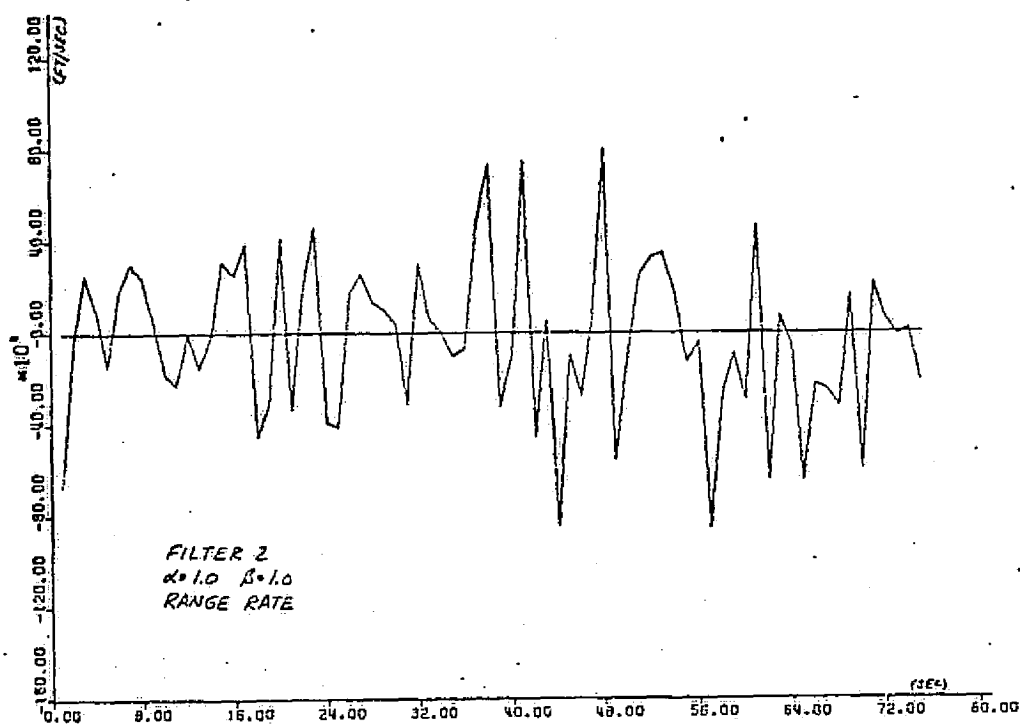
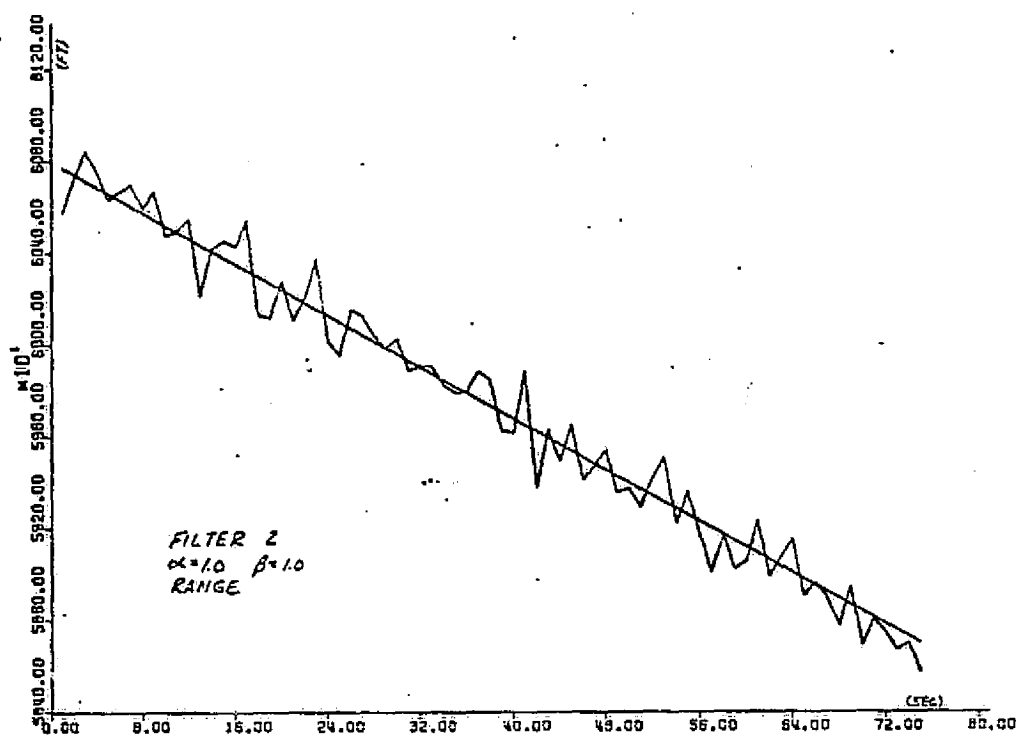


FIGURE 31 - Filter 2 at 2500 samples/sec.

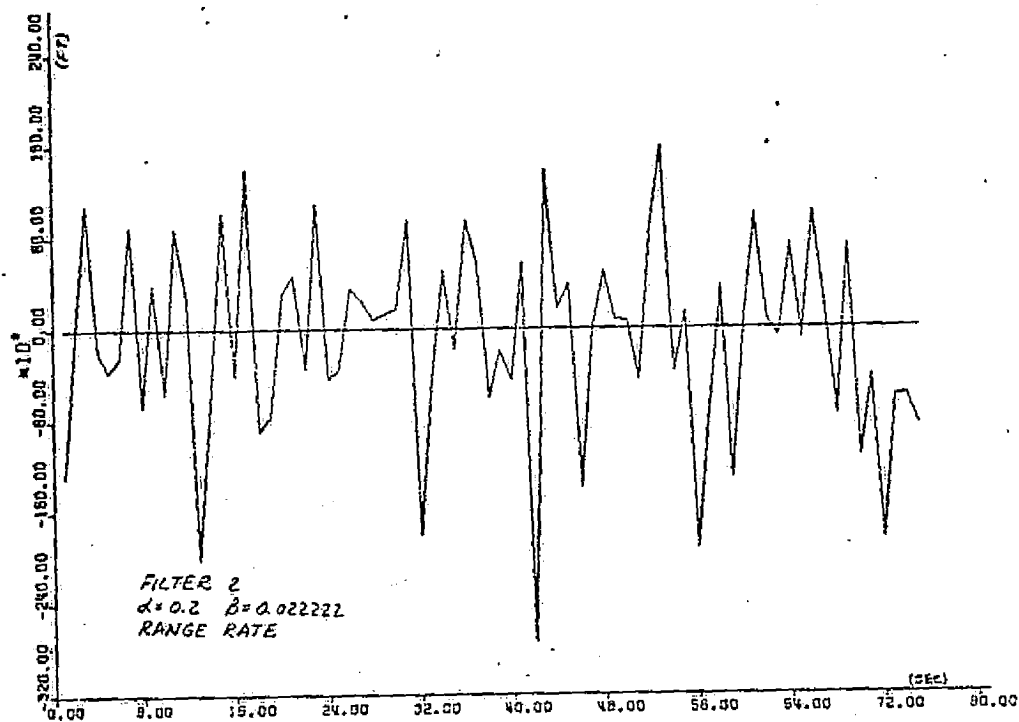
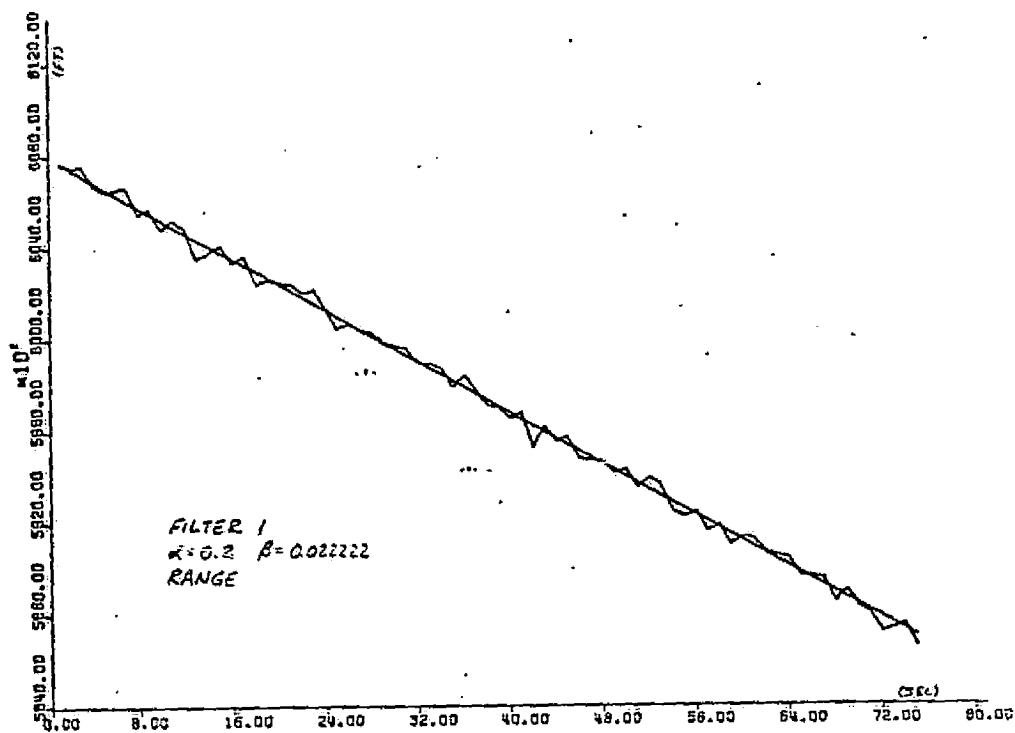


FIGURE 32 - Filter 2 at 2500 samples/sec.

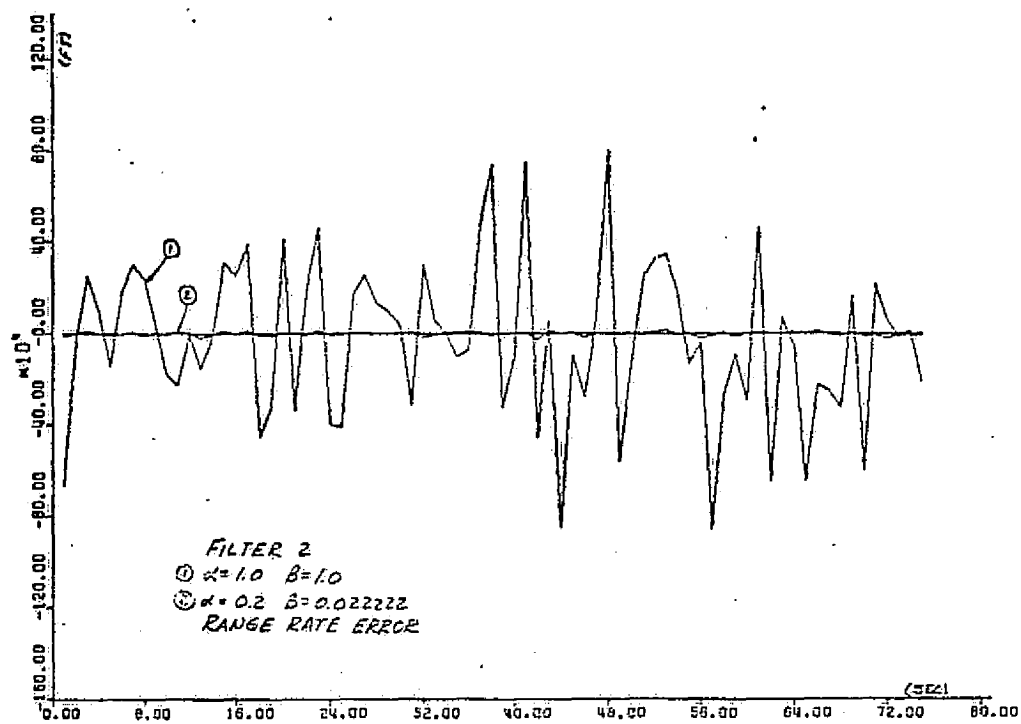
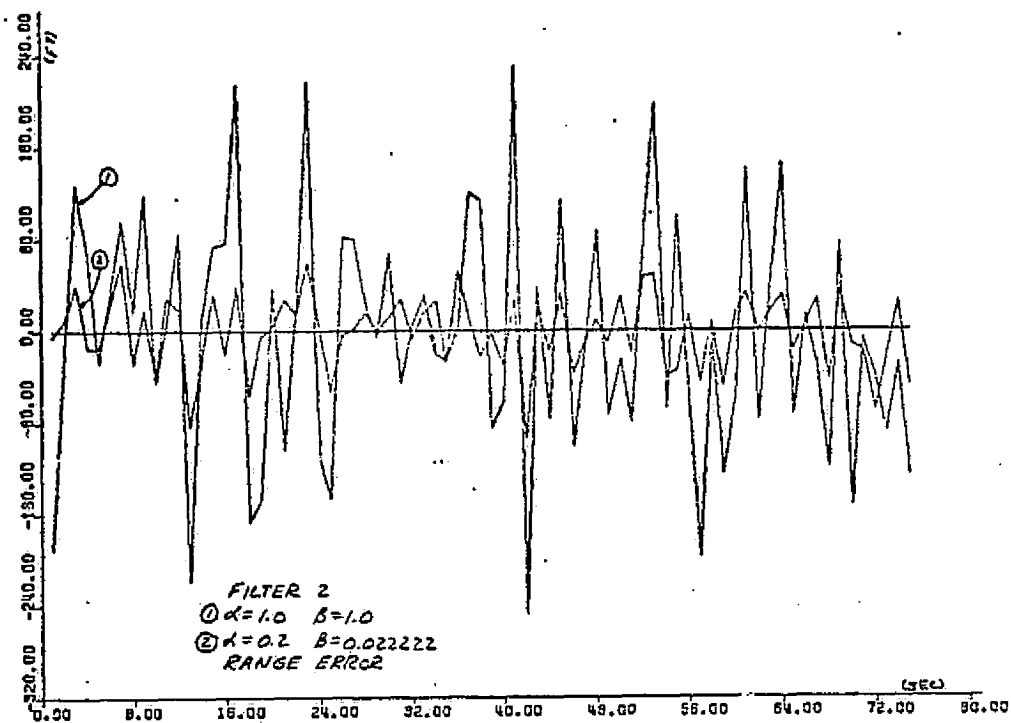


FIGURE 33 - Filter 2 at 2500 samples/sec.

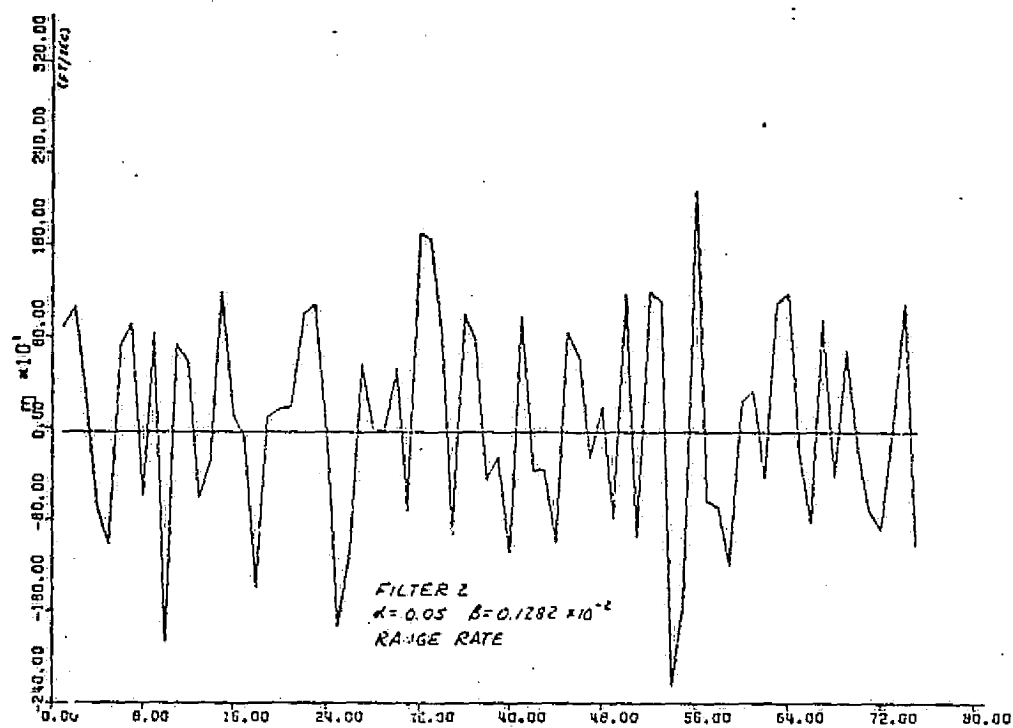
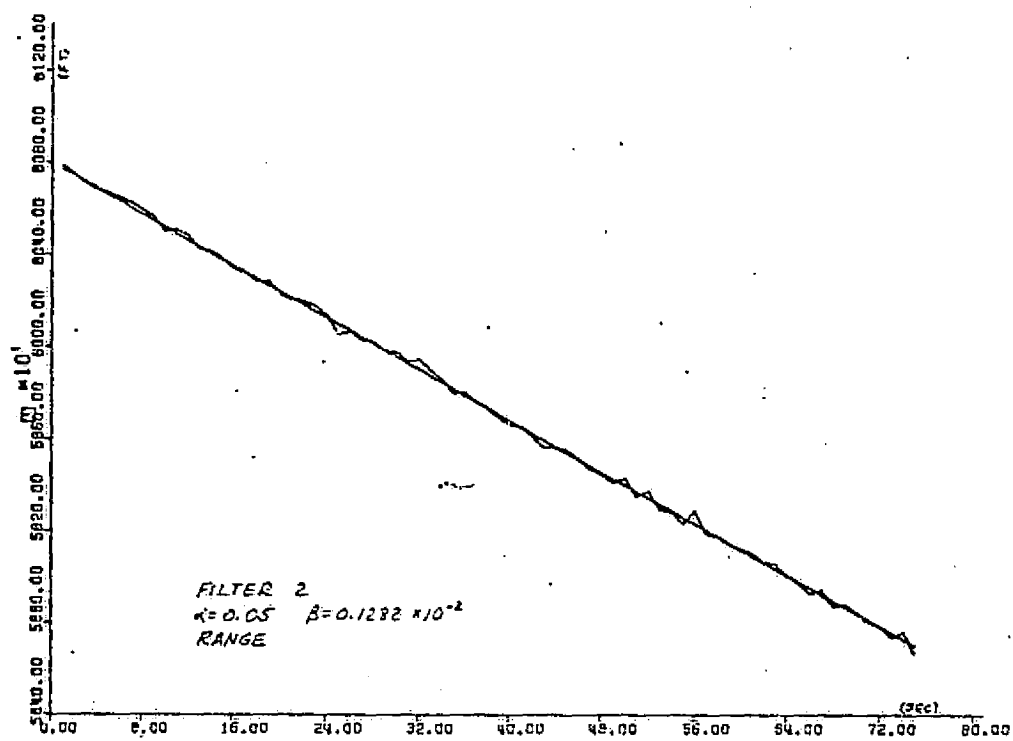


FIGURE 34 - Filter 2 at 2500 samples/sec.

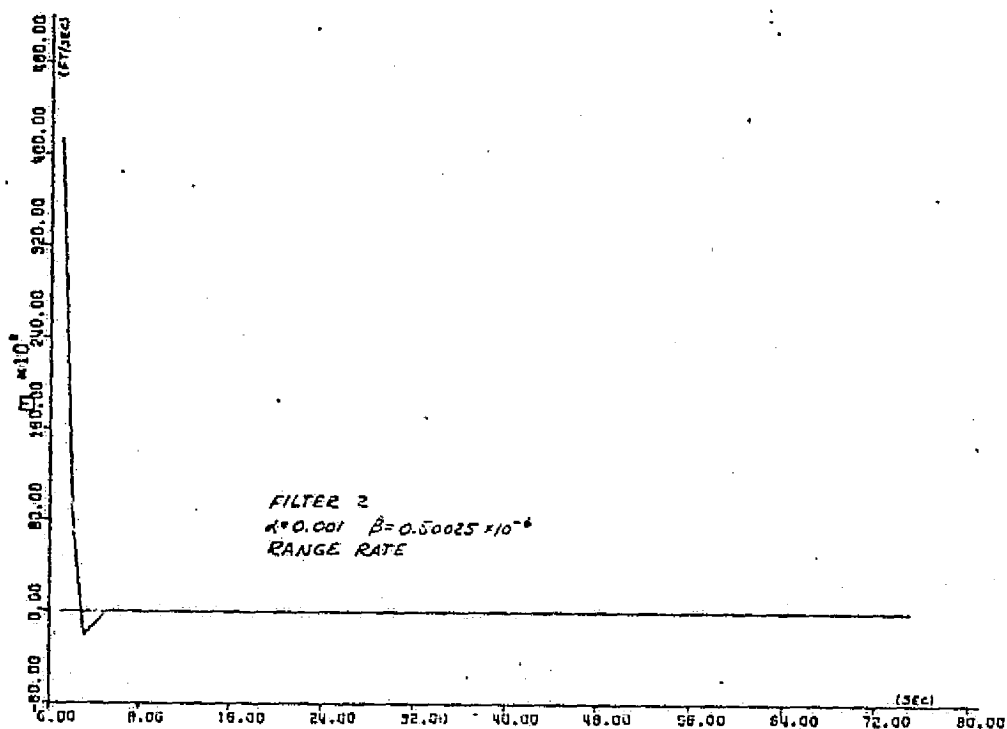
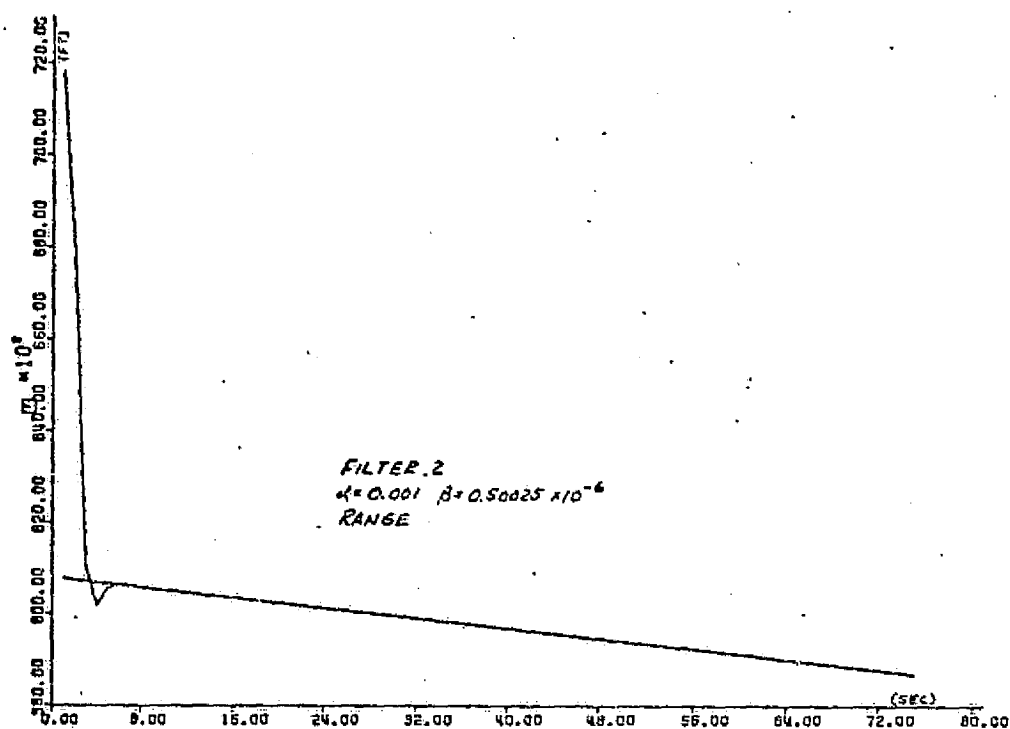


FIGURE 35 - Filter 2 at 2500 samples/sec.

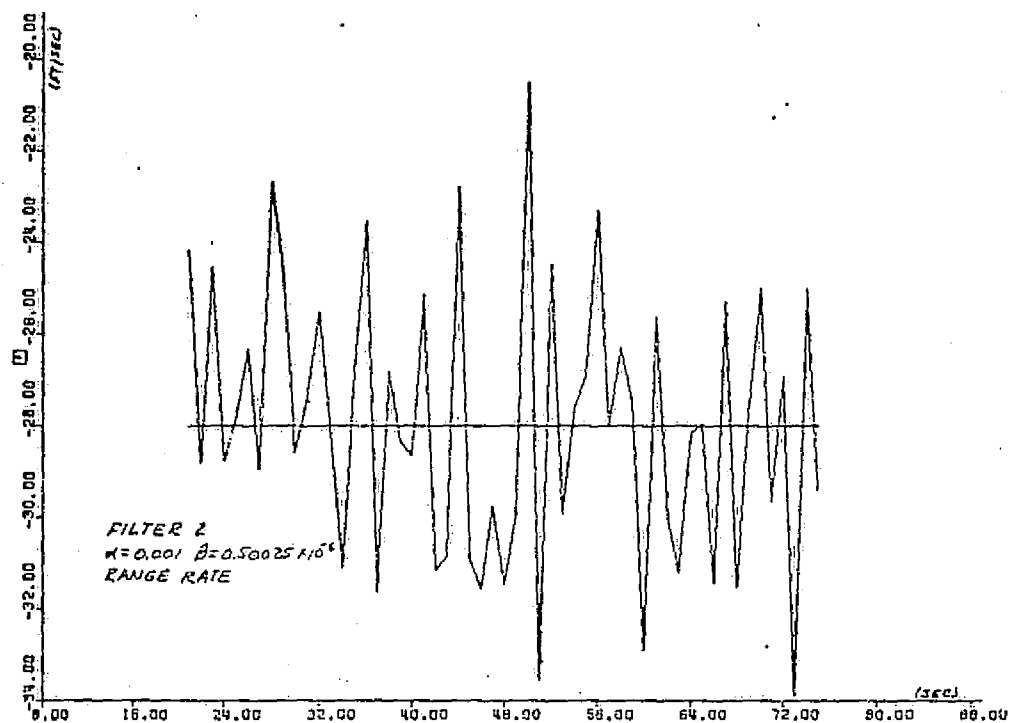
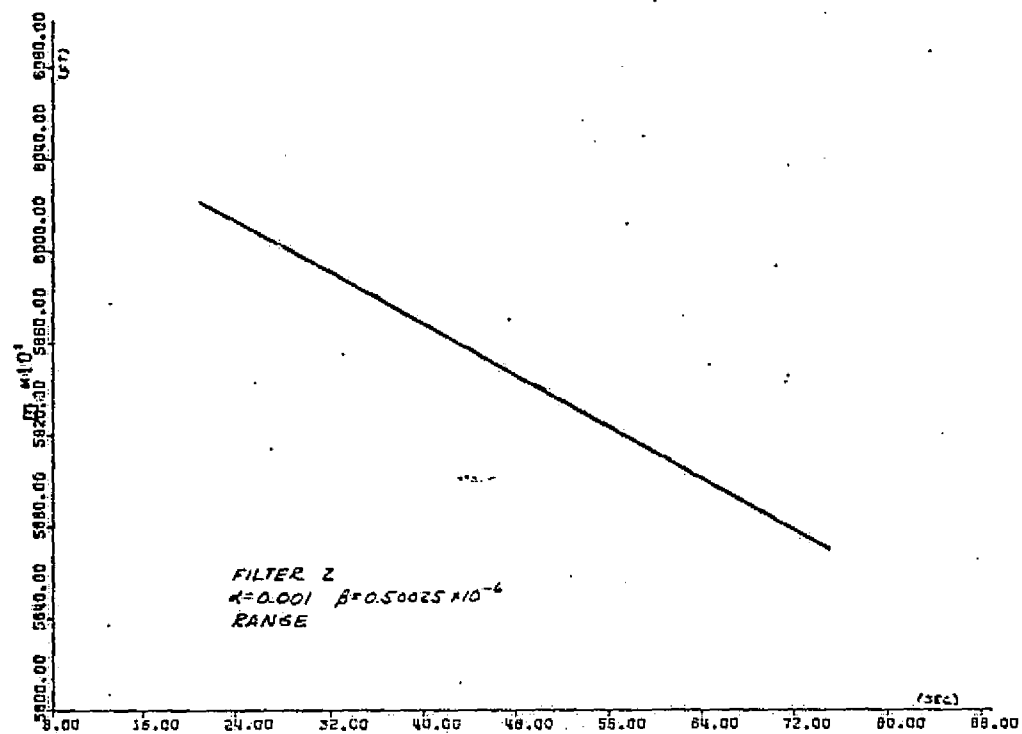


FIGURE 36 - Filter 2 at 2500 samples/sec.

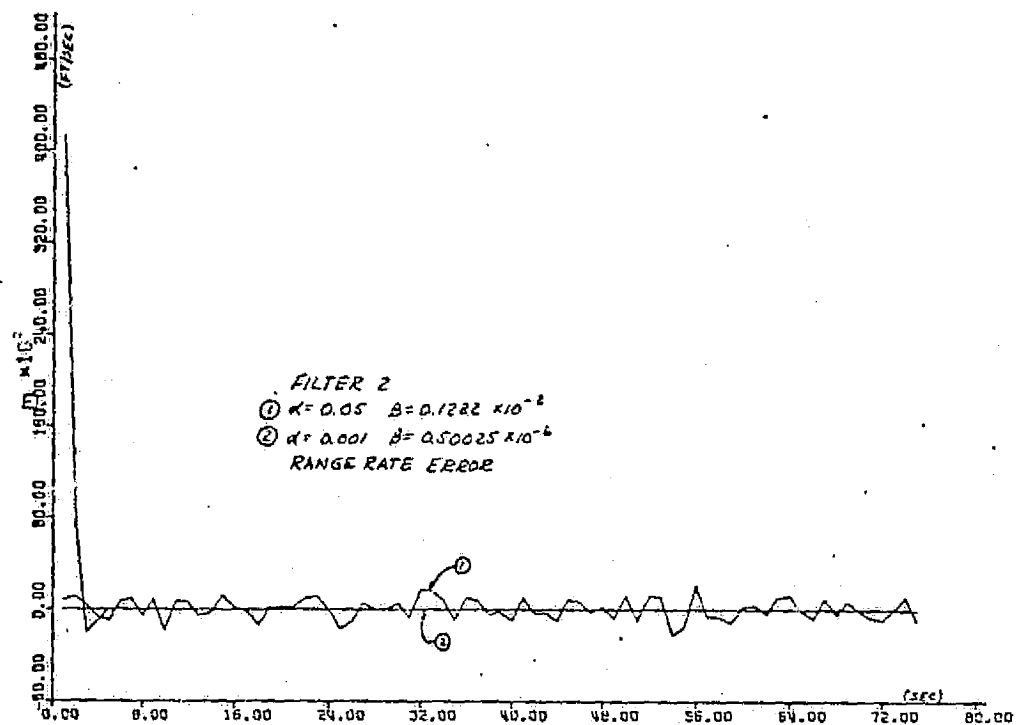
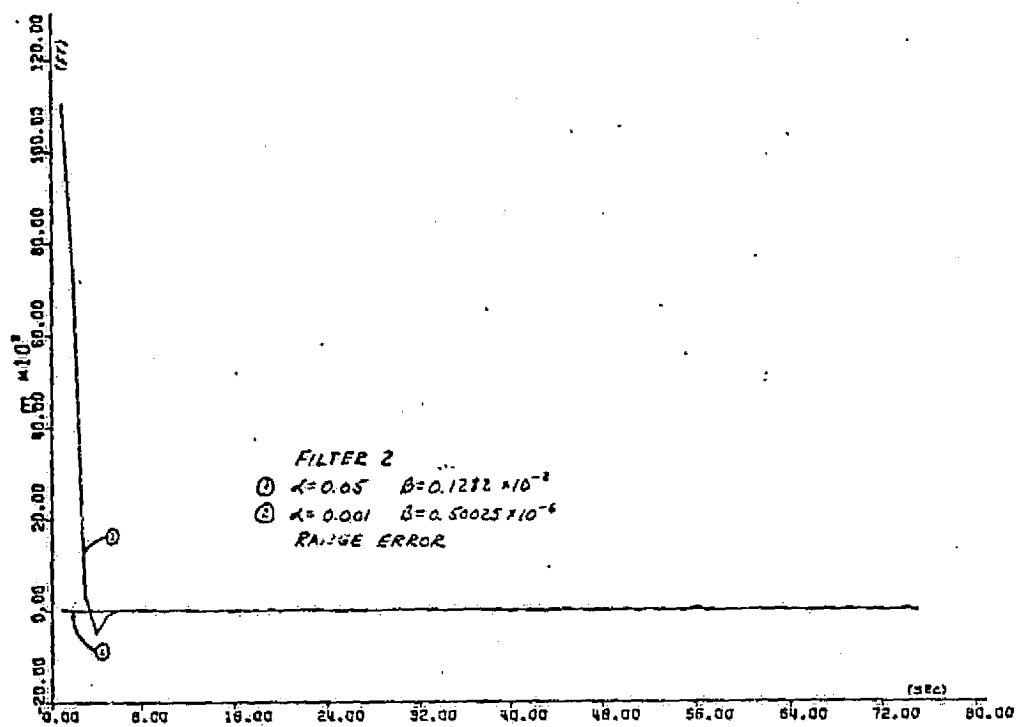


FIGURE 37 - Filter 2 at 2500 samples/sec.

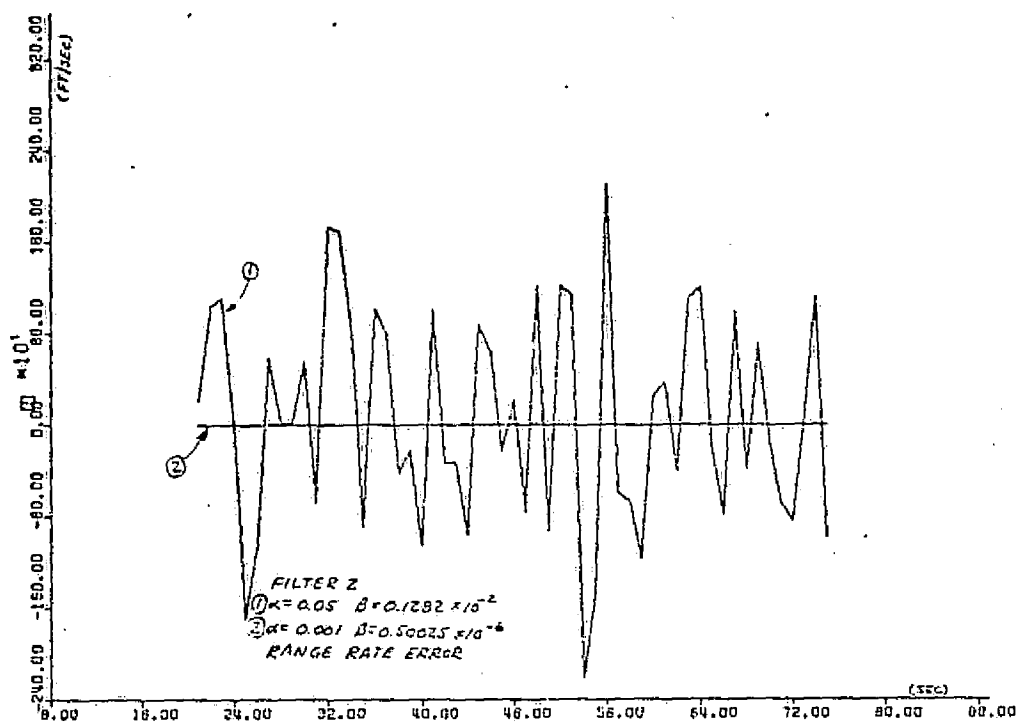
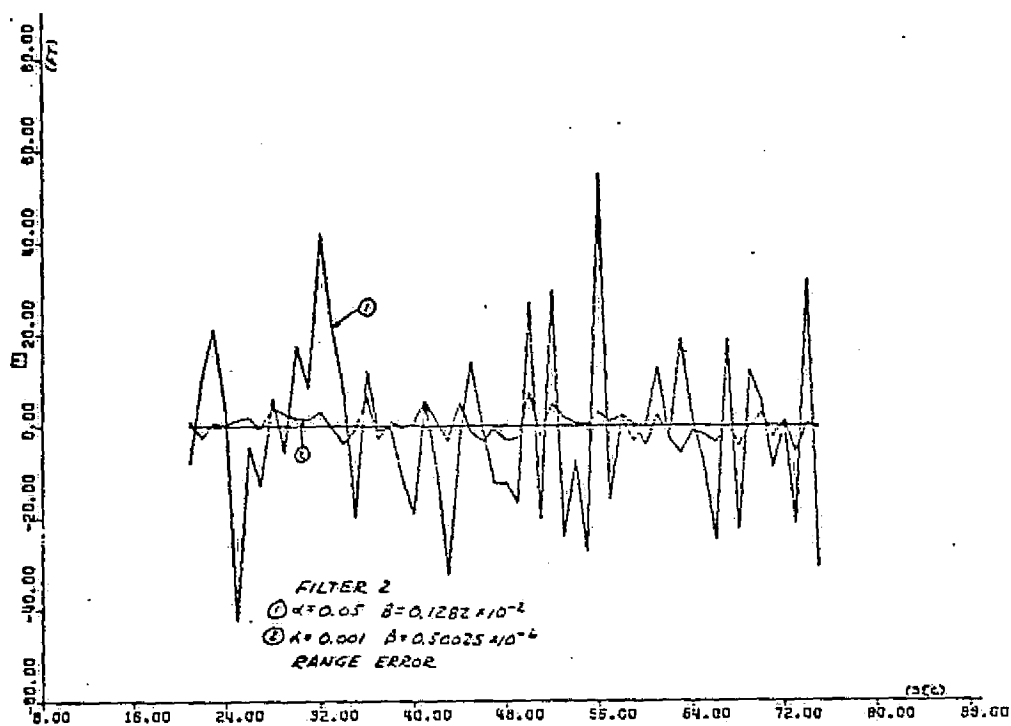


FIGURE 38 - Filter 2 at 2500 samples/sec.

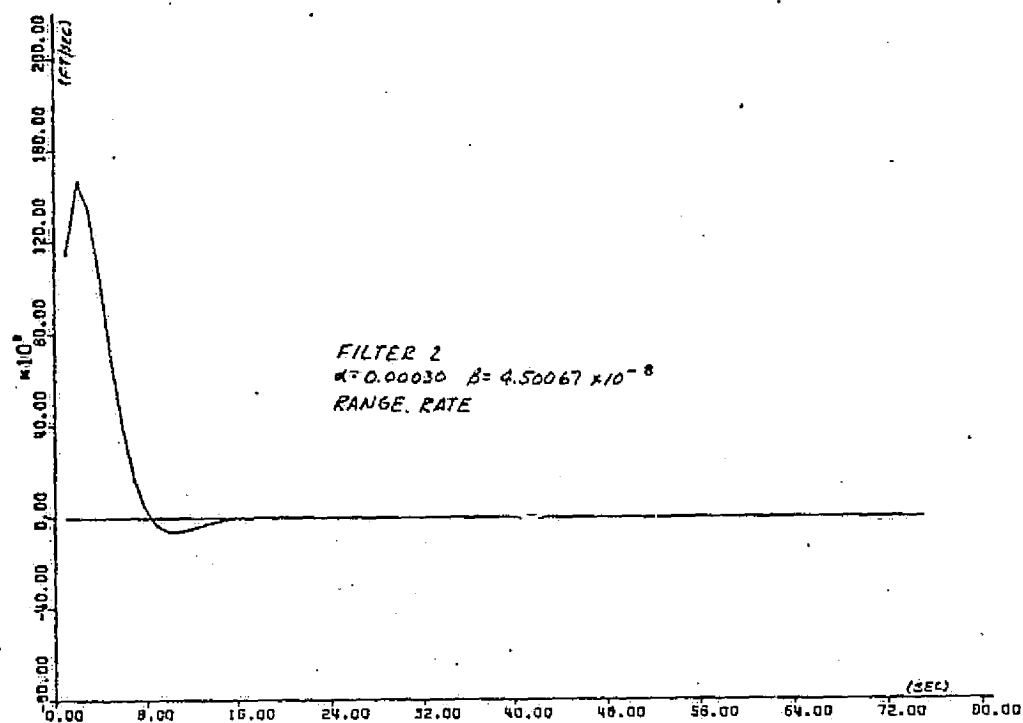
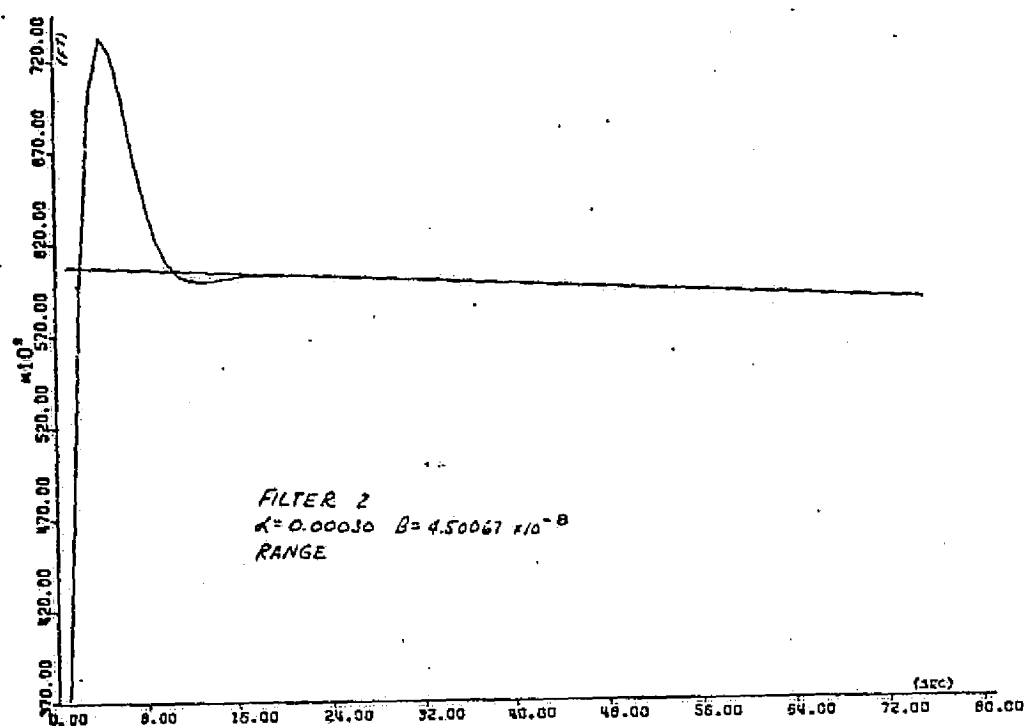


FIGURE 39 - Figure 2 at 2500 samples/sec.

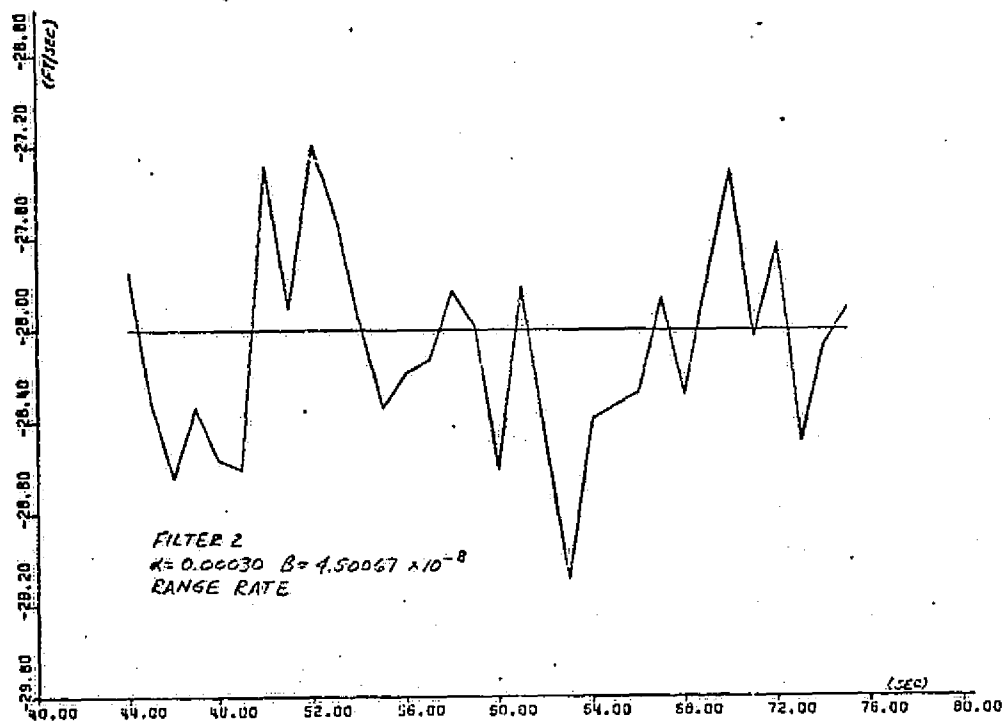
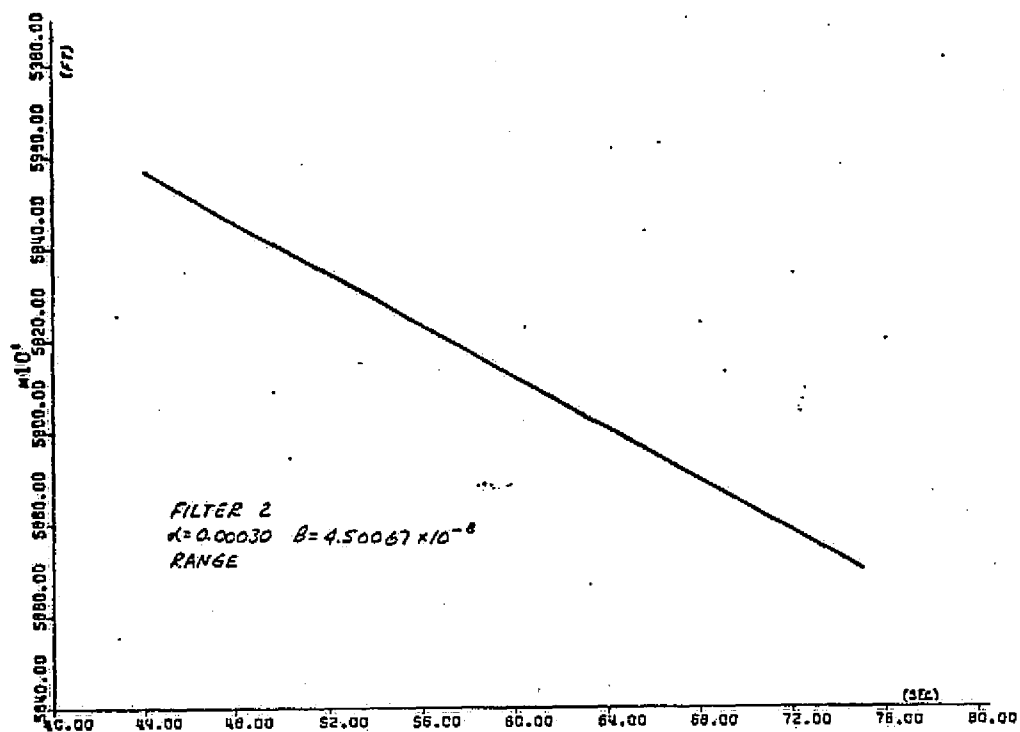


FIGURE 40 - Filter 2 at 2500 samples/sec.

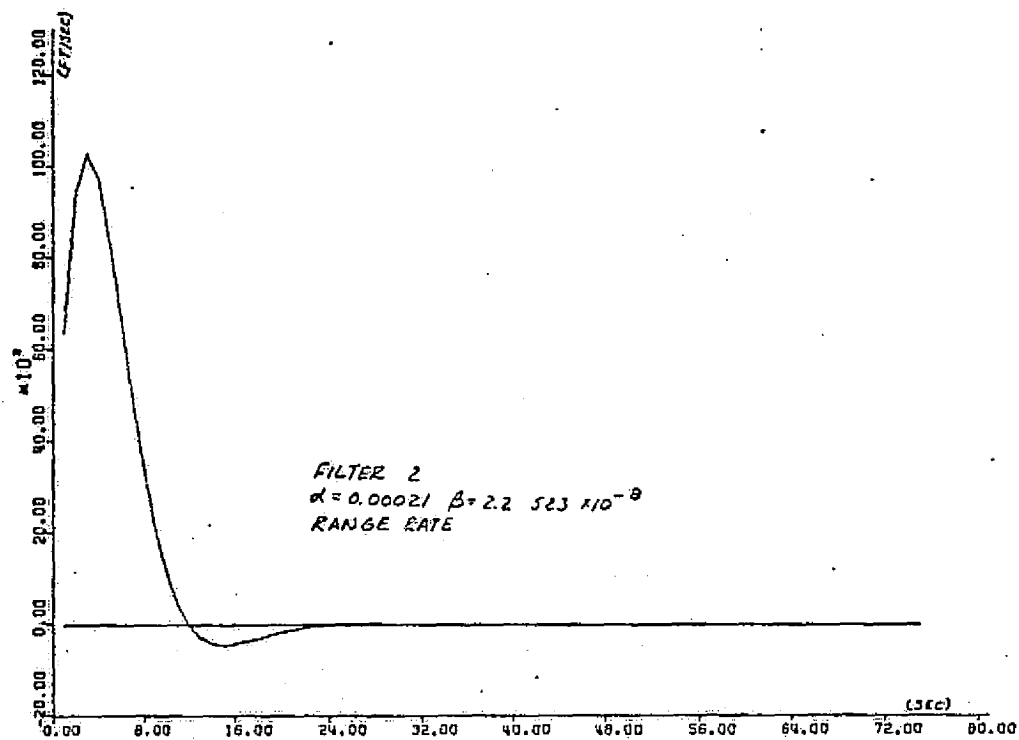
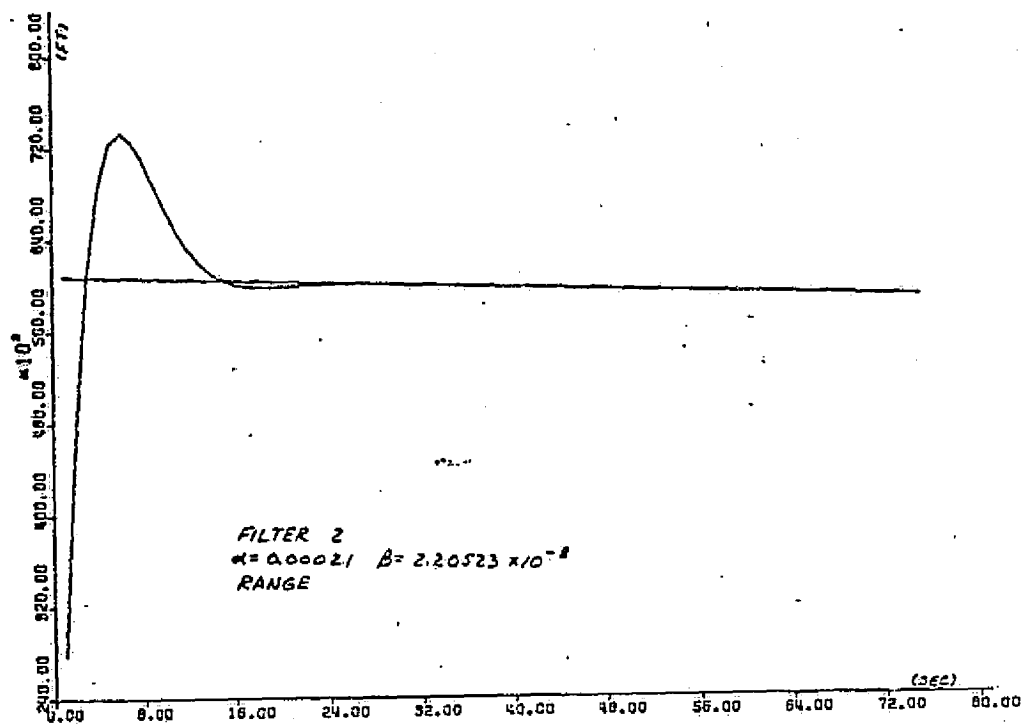


FIGURE 41 - Filter 2 at 2500 samples/sec.

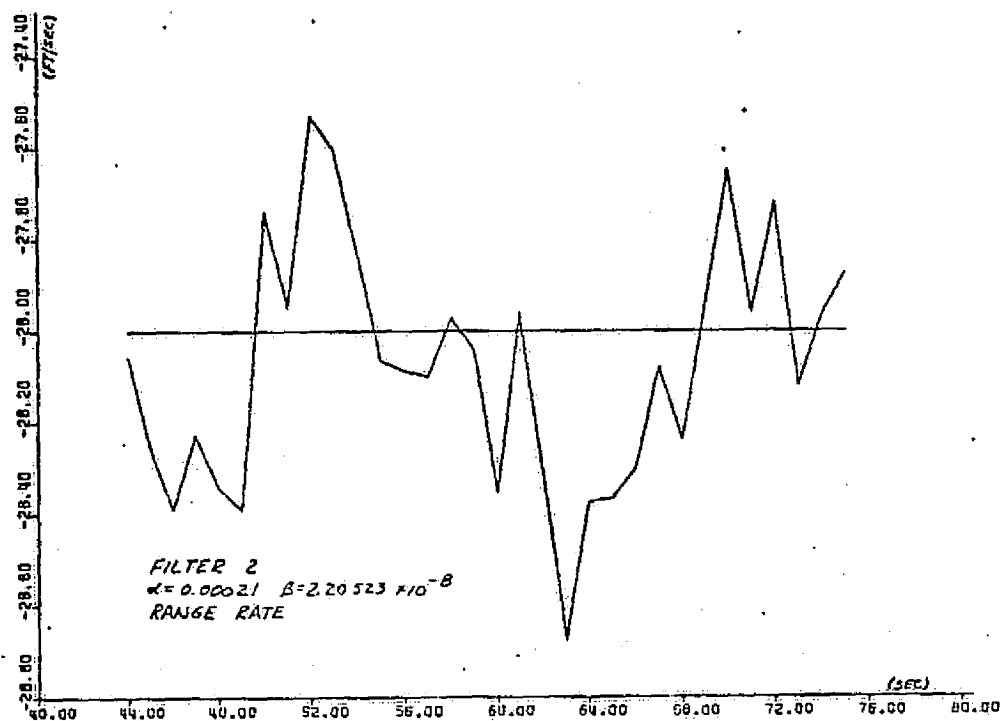
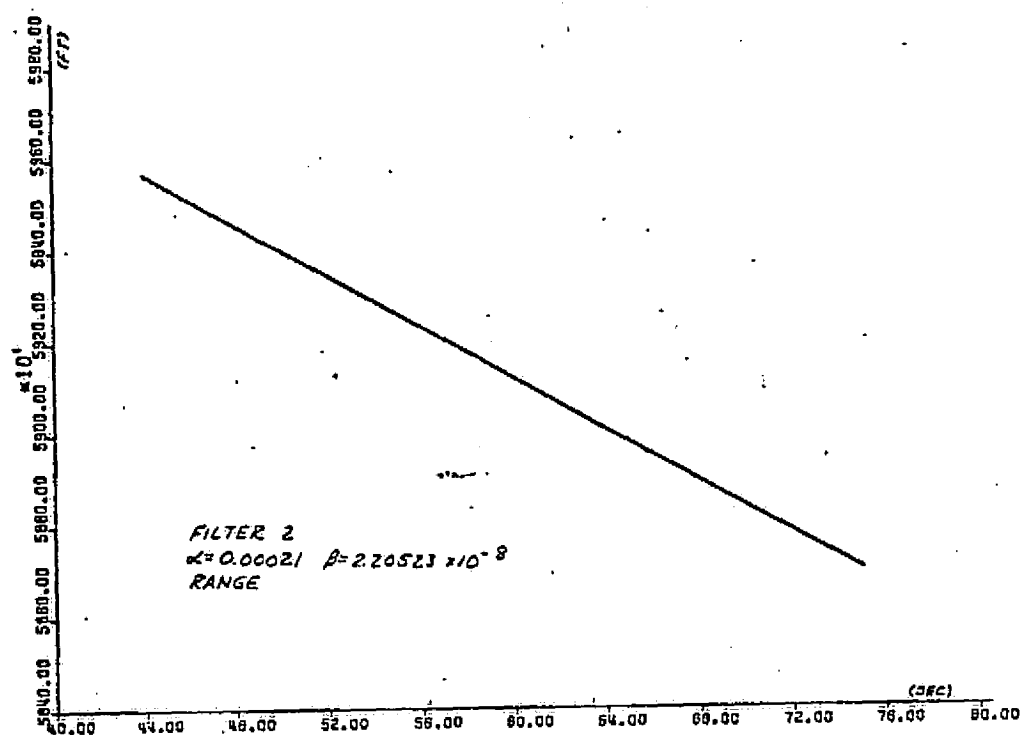


FIGURE 42 - Filter 2 at 2500 samples/sec.

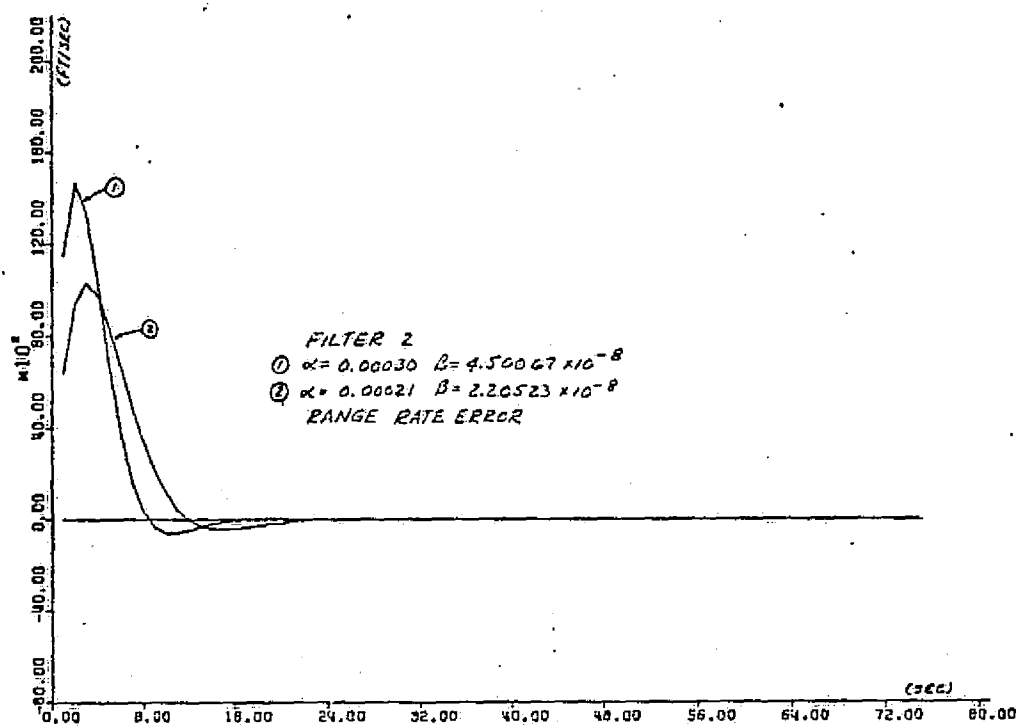
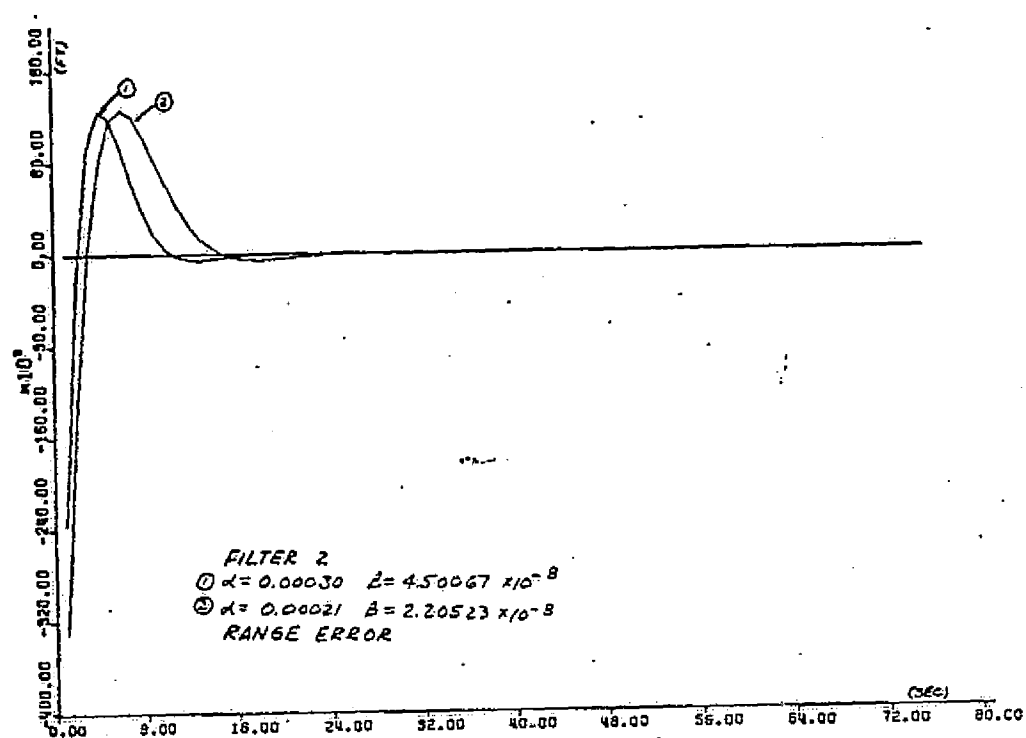


FIGURE 43 - Filter 2 at 2500 samples/sec.

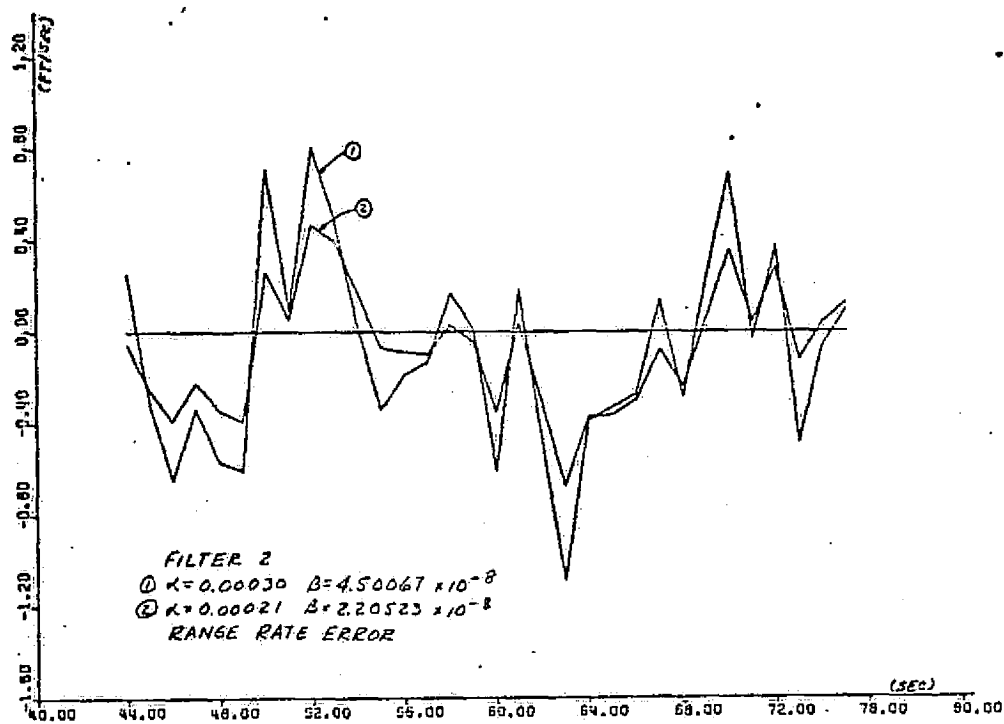
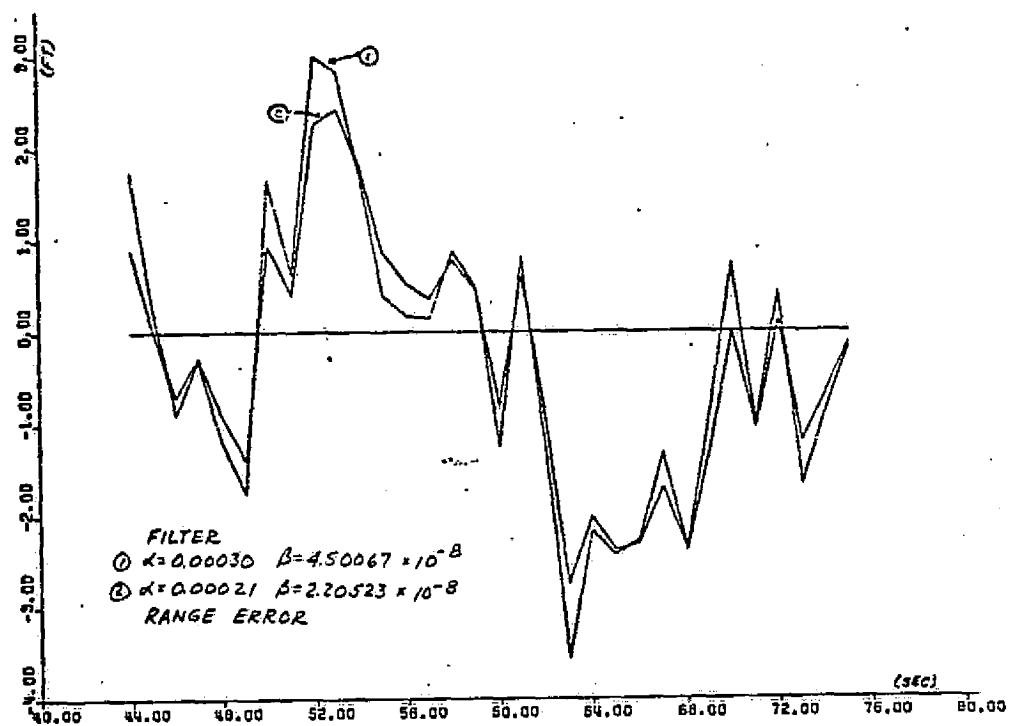


FIGURE 44 - Filter 2 at 2500 samples/sec.

TABLE XXI

FILTER 2 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES /SEC.

FILTER 2 alpha - beta	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
$\alpha = 1.0$ $\beta = 1.0$	65.3068	65.3068	91.3215	91.3215	115.2178	115.2178
$\alpha = 0.2$ $\beta = 0.022222$	65.3068	65.3068	91.3215	30.0503	115.2178	83.4223
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	65.3068	65.3068	91.3215	8.7326	115.2178	69.3696
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	65.3068	—	91.3215	0.2055	115.2178	—
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	65.3068	—	91.3215	0.6791×10^{-1}	115.2178	—
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	65.3068	—	91.3215	0.4817×10^{-1}	115.2178	—

THIS PAGE BLANK NOT REPRODUCED

TABLE XXIII

FILTER 2 MAXIMUM RANGE(FT.) AND RANGE RATE (FT./SEC.) ERRORS AT 6.25 SAMPLES /SEC.

FILTER 2 alpha - beta	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
$\alpha = 1.0$ $\beta = 1.0$	0.0	0.682×10^{-12}	230.565	1613.121	230.565	1613.121
$\alpha = 0.2$ $\beta = 0.022222$	0.0	0.233×10^{-11}	71.466	51.347	71.466	51.347
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	0.646×10^{-8}	0.279×10^{-8}	23.015	3.665	23.015	3.665
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	—	—	5.503	0.136×10^{-1}	—	—
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	—	—	1.964	0.172×10^{-2}	—	—
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	—	—	1.404	0.879×10^{-3}	—	—

TABLE XXIV

FILTER 2 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

FILTER 2 alpha - beta	RUN 2 input = noise		PREDICTED BY BENEDICT AND BORDNER		τ CONVERGENCE TIME (SEC.)
	KR RANGE VARIANCE REDUCTION RATIO	KRR RANGE RATE VARIANCE REDUCTION RATIO	KR	KRR	
$\alpha = 1.0$ $\beta = 1.0$	1.0	67.778	1.0	78.125	<1
$\alpha = 0.2$ $\beta = 0.022222$	1.083×10^{-1}	4.345×10^{-2}	1.554×10^{-1}	5.392×10^{-2}	15
$\alpha = 0.05$ $\beta = 0.1282 \times 10^{-2}$	9.144×10^{-3}	2.842×10^{-4}	3.782×10^{-2}	6.587×10^{-4}	56
$\alpha = 0.001$ $\beta = 5.0025 \times 10^{-7}$	5.064×10^{-6}	4.160×10^{-11}	7.501×10^{-4}	4.890×10^{-9}	>192
$\alpha = 0.00030$ $\beta = 4.50067 \times 10^{-8}$	5.530×10^{-7}	4.497×10^{-13}	2.250×10^{-4}	1.319×10^{-10}	>192
$\alpha = 0.00021$ $\beta = 2.20523 \times 10^{-8}$	2.782×10^{-7}	1.132×10^{-13}	1.575×10^{-4}	4.523×10^{-11}	>192

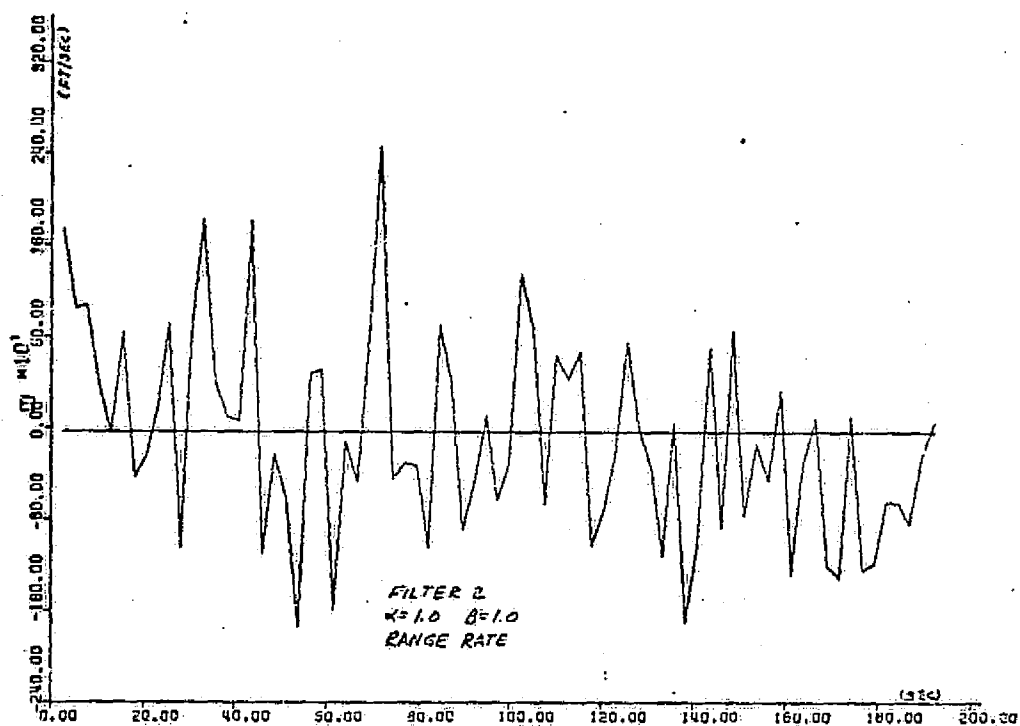
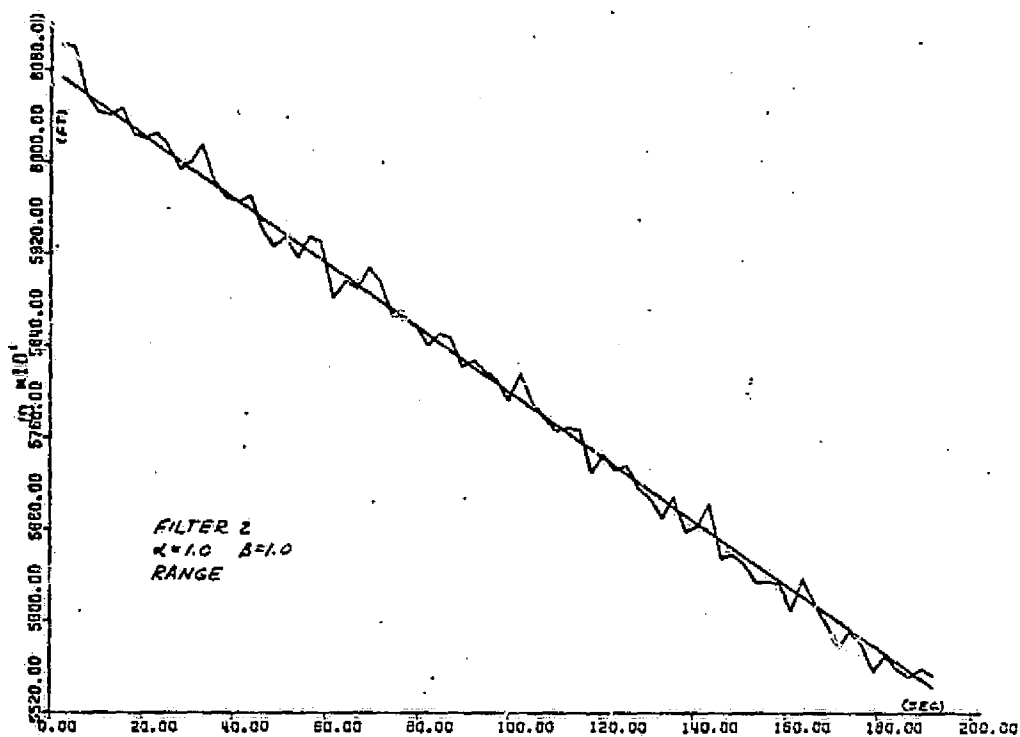


FIGURE 45 - Filter 2 at 6.25 samples/sec.

C2

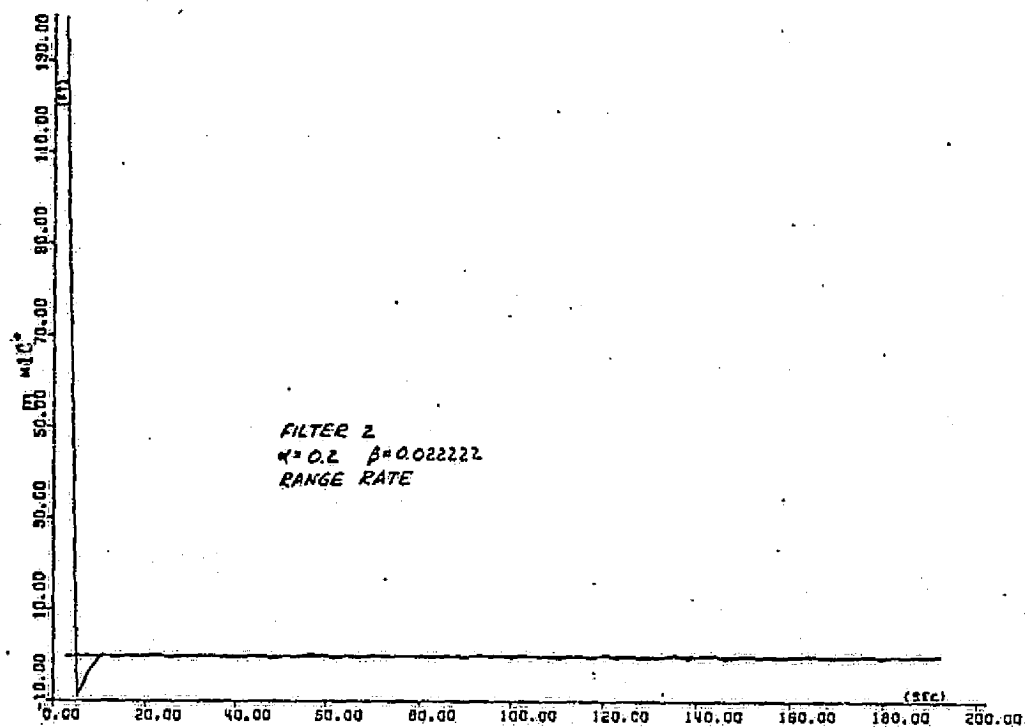
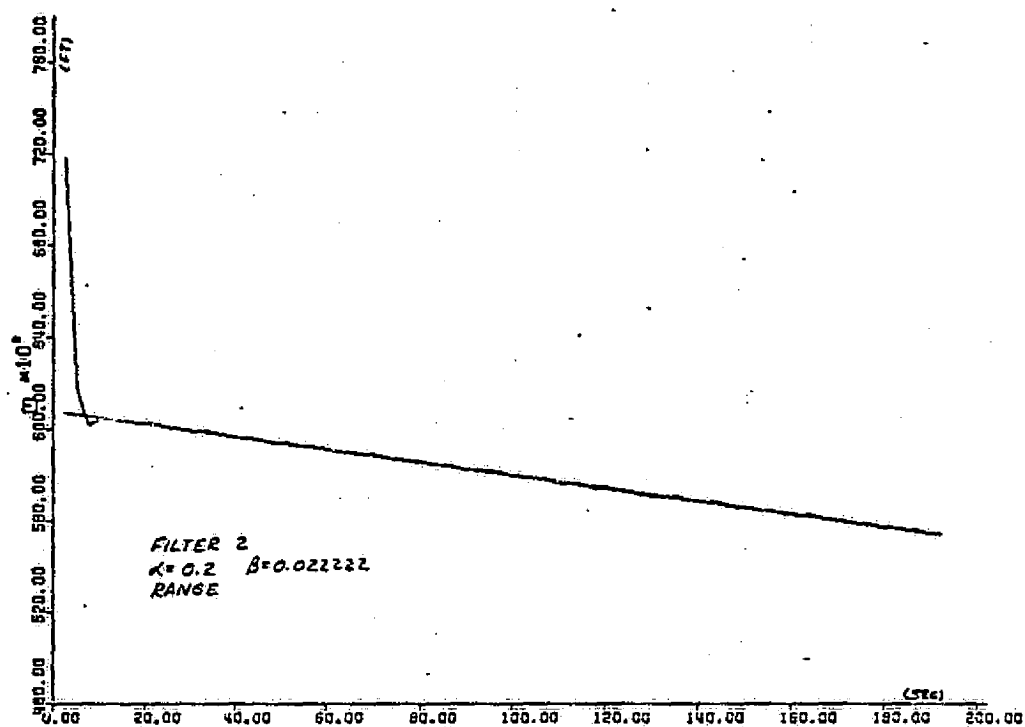


FIGURE 46 - Filter 2 at 6.25 samples/sec.

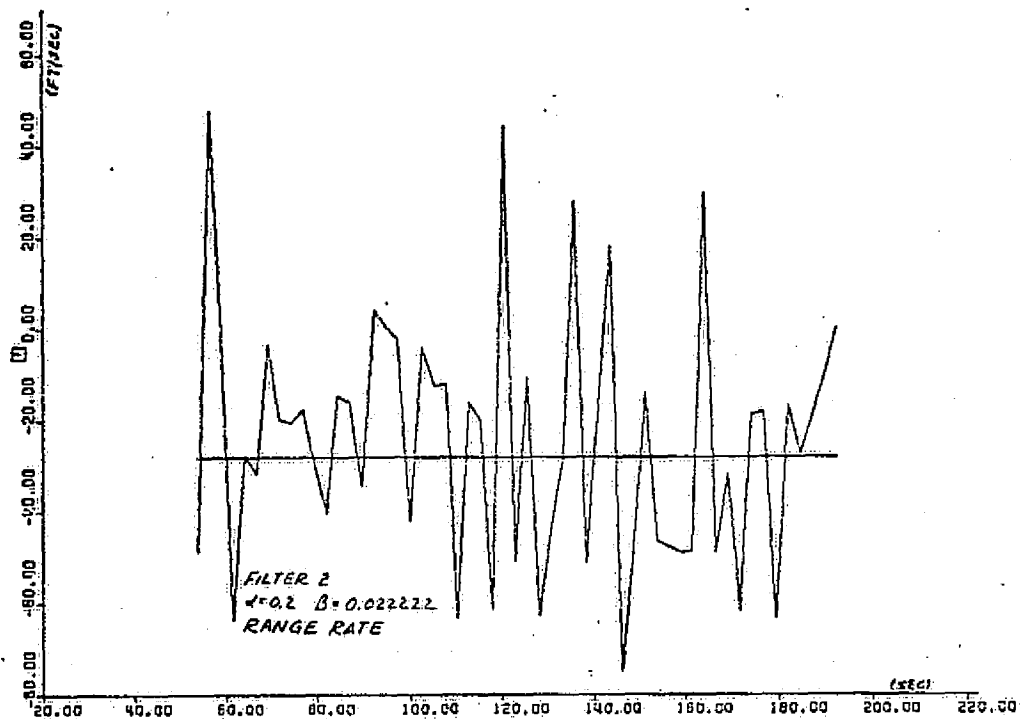
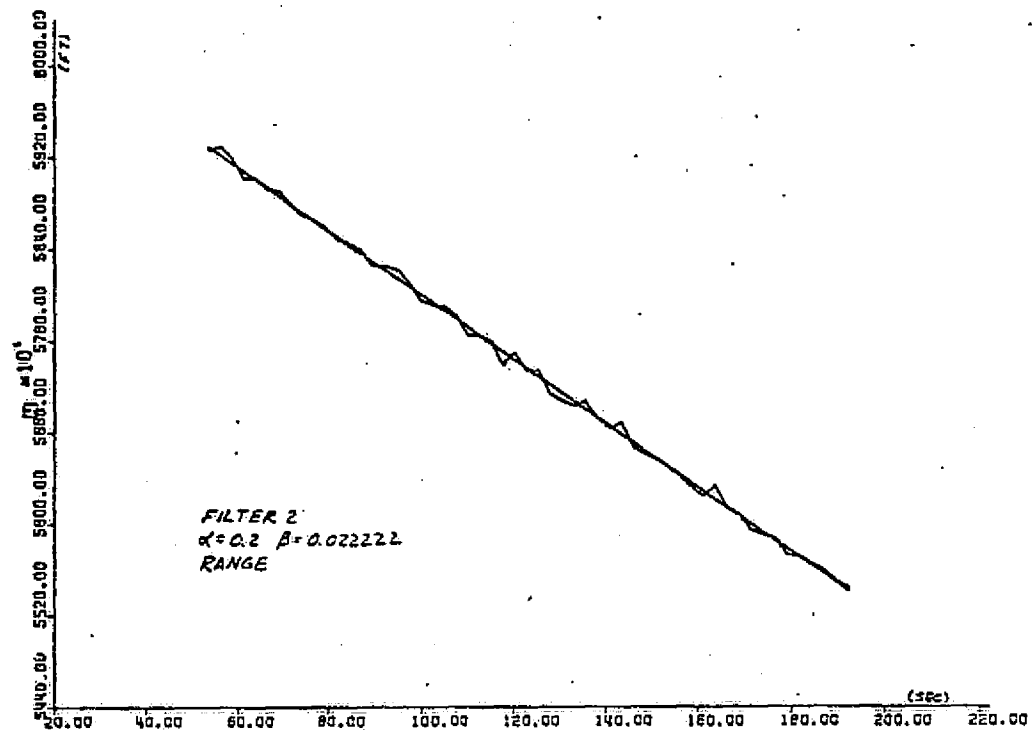


FIGURE 47 - Filter 2 at 6.25 samples/sec.

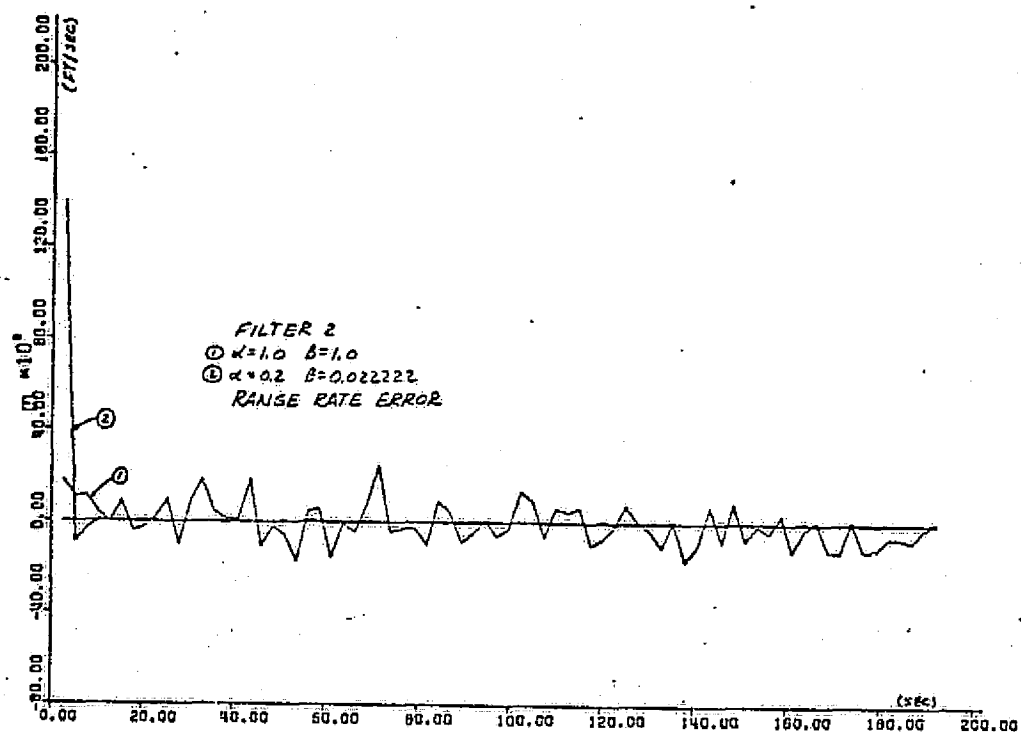
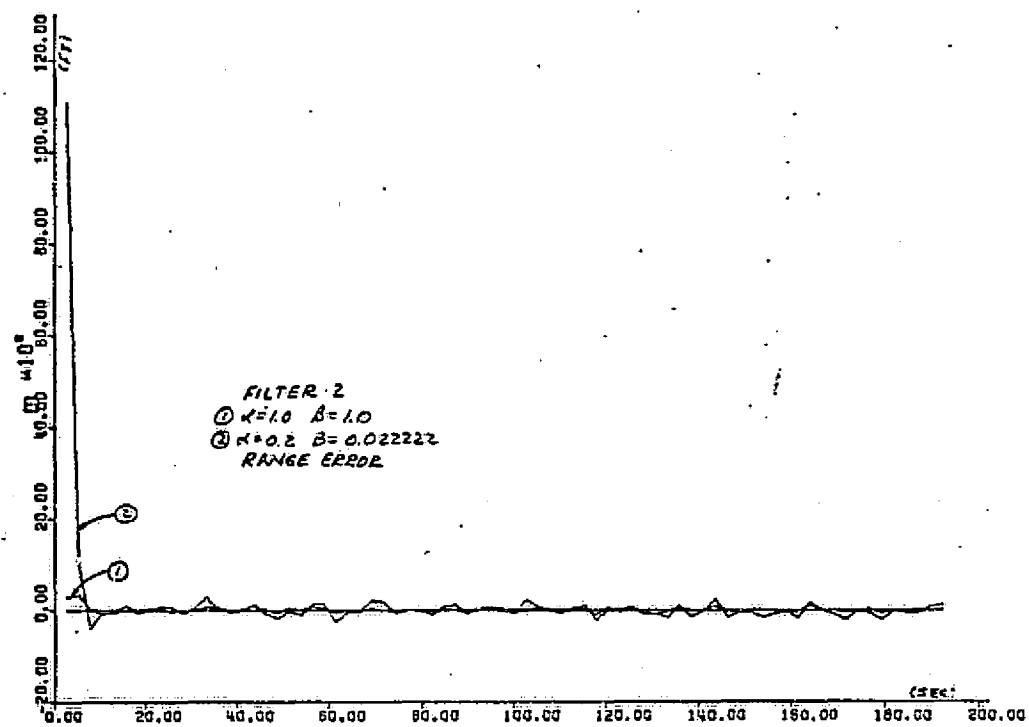


FIGURE 48 - Filter 2 at 6.25 samples/sec.

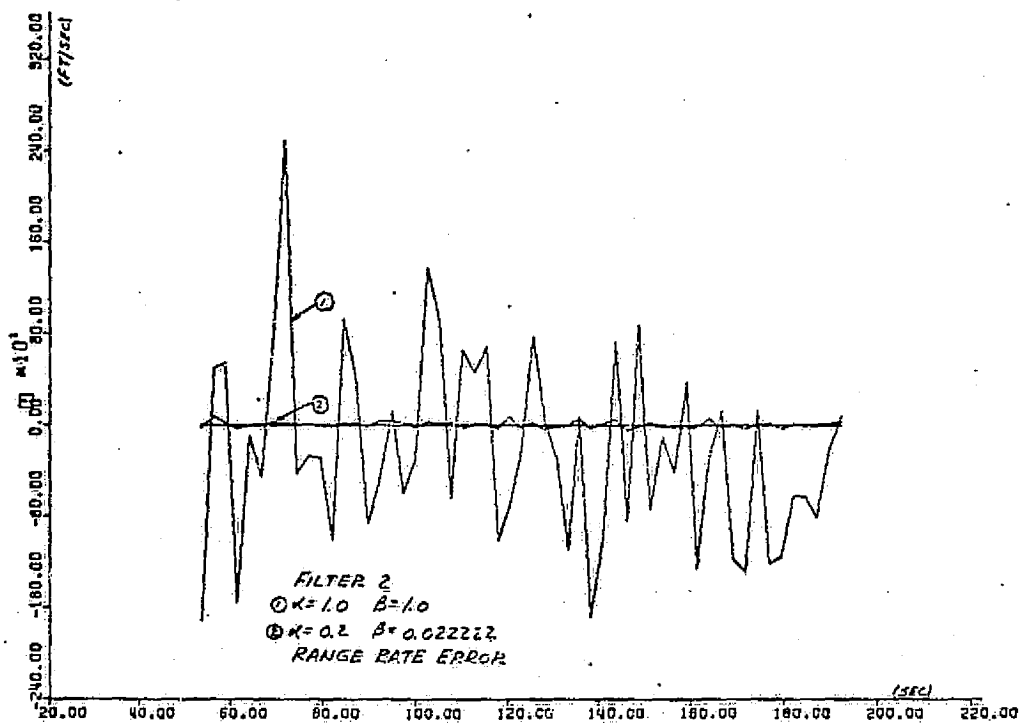
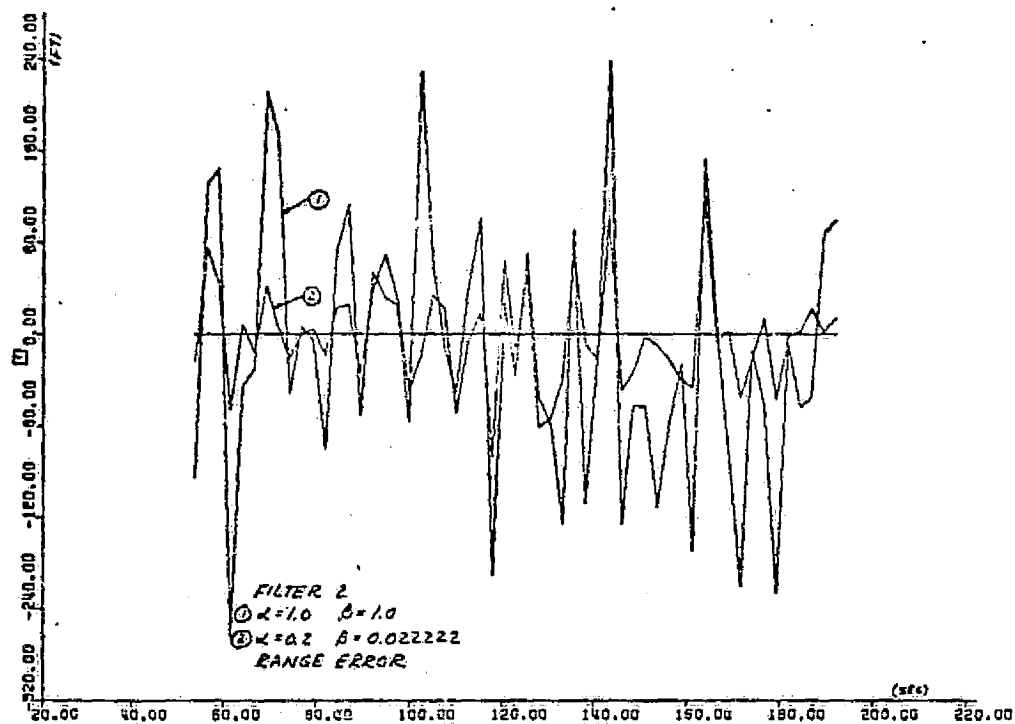


FIGURE 49 - Filter 2 at 6.25 samples/sec.

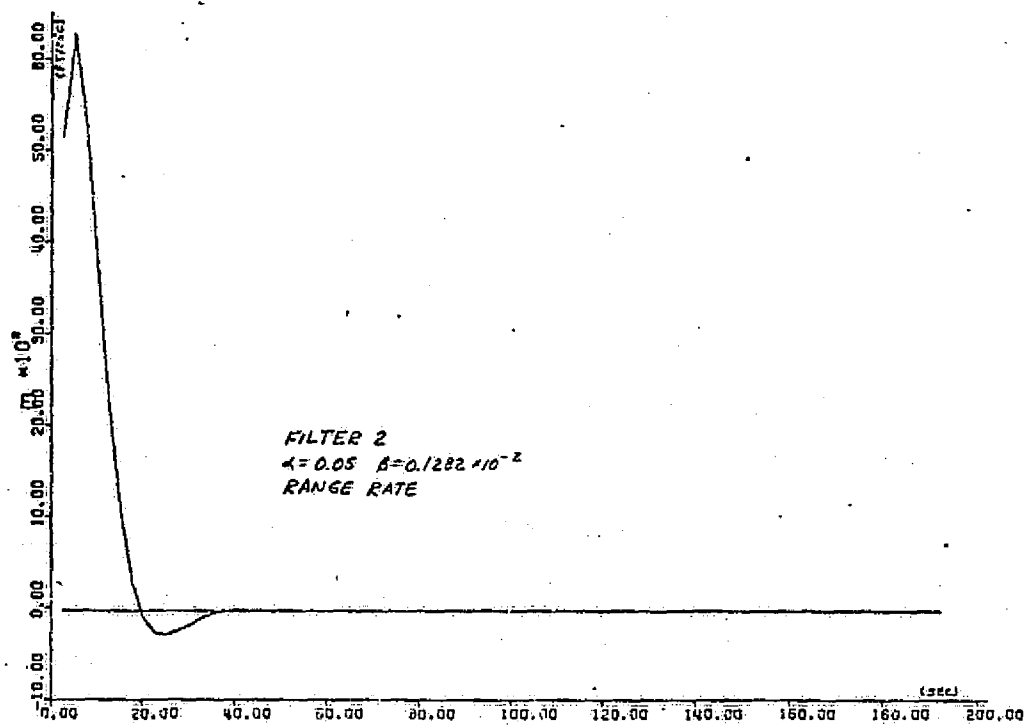
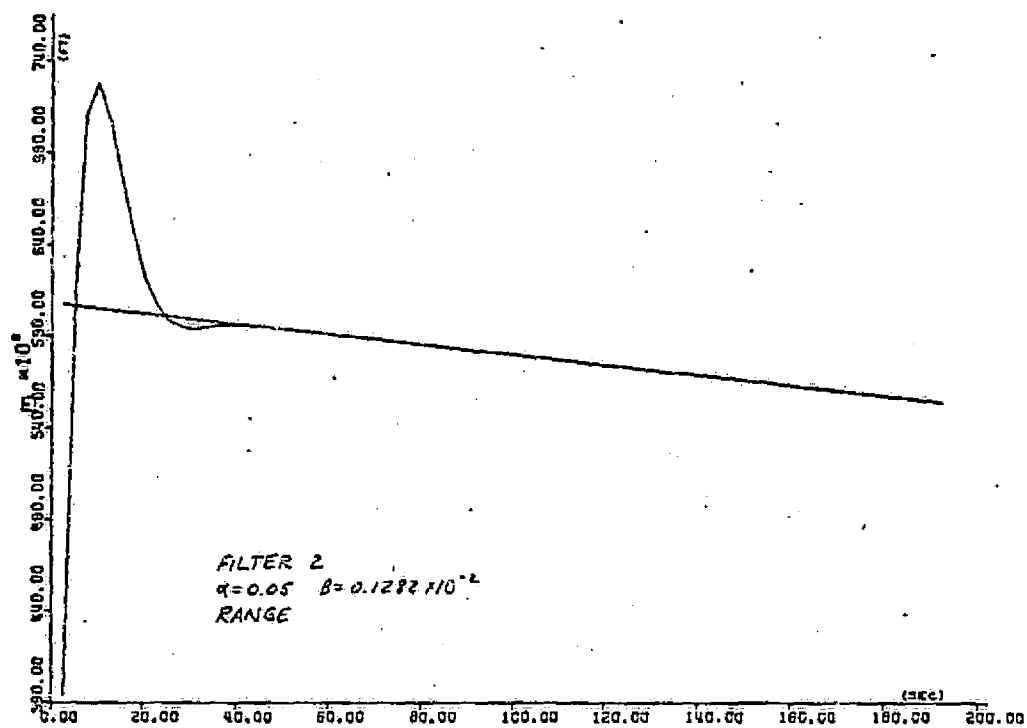


FIGURE 50 - Filter 2 at 6.25 samples/sec.

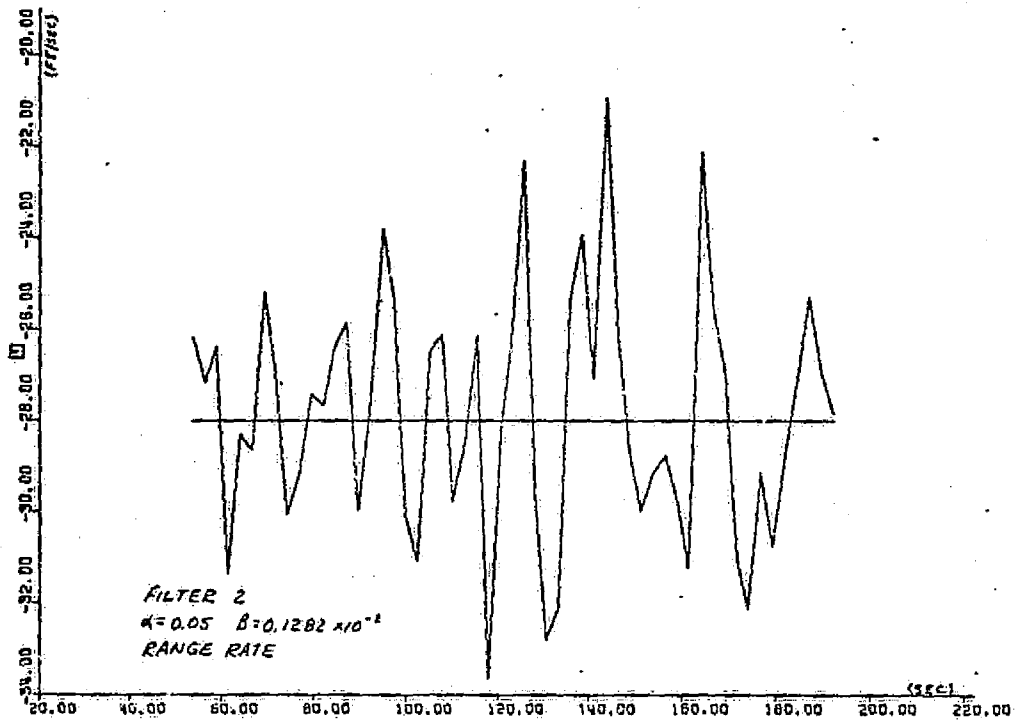
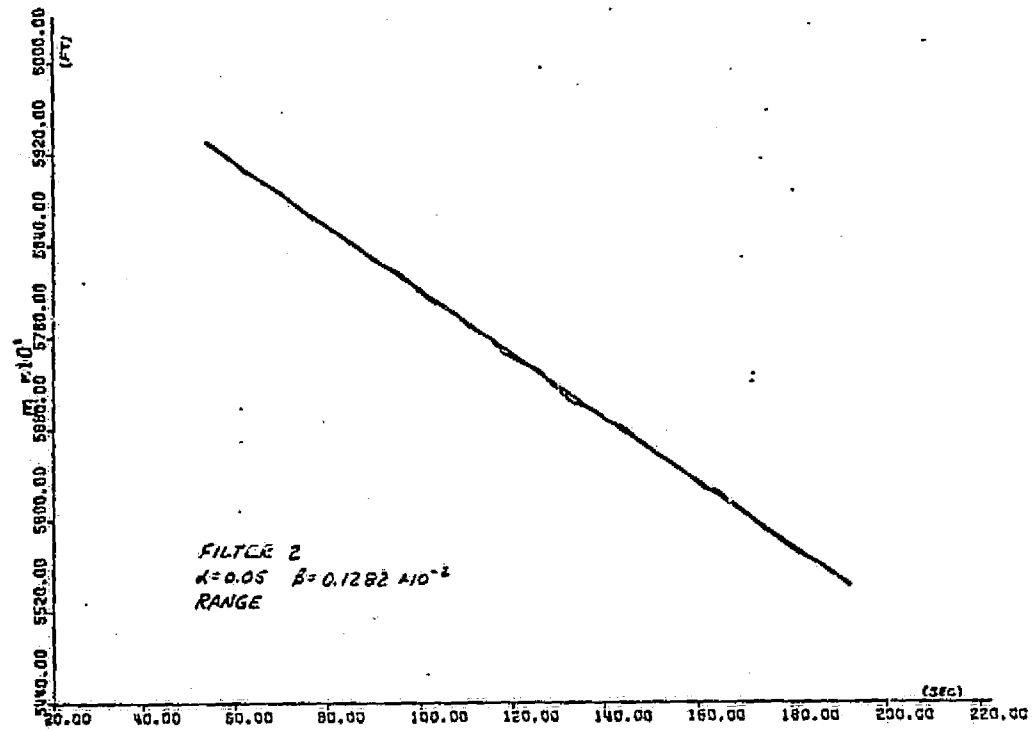


FIGURE 51 - Filter 2 at 6.25 samples/sec.

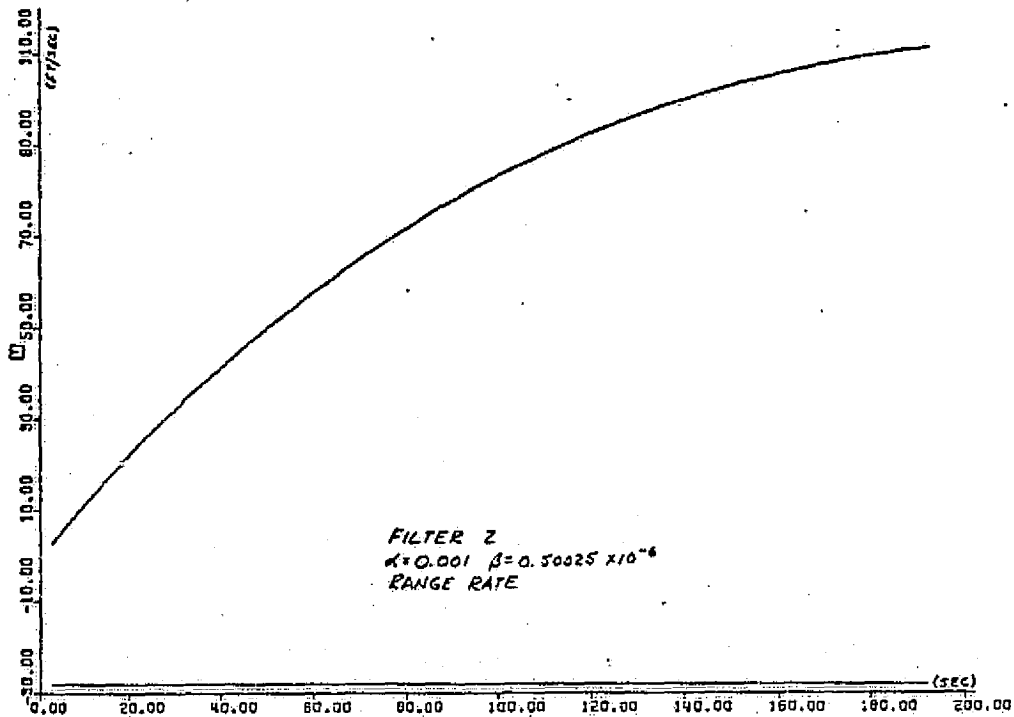
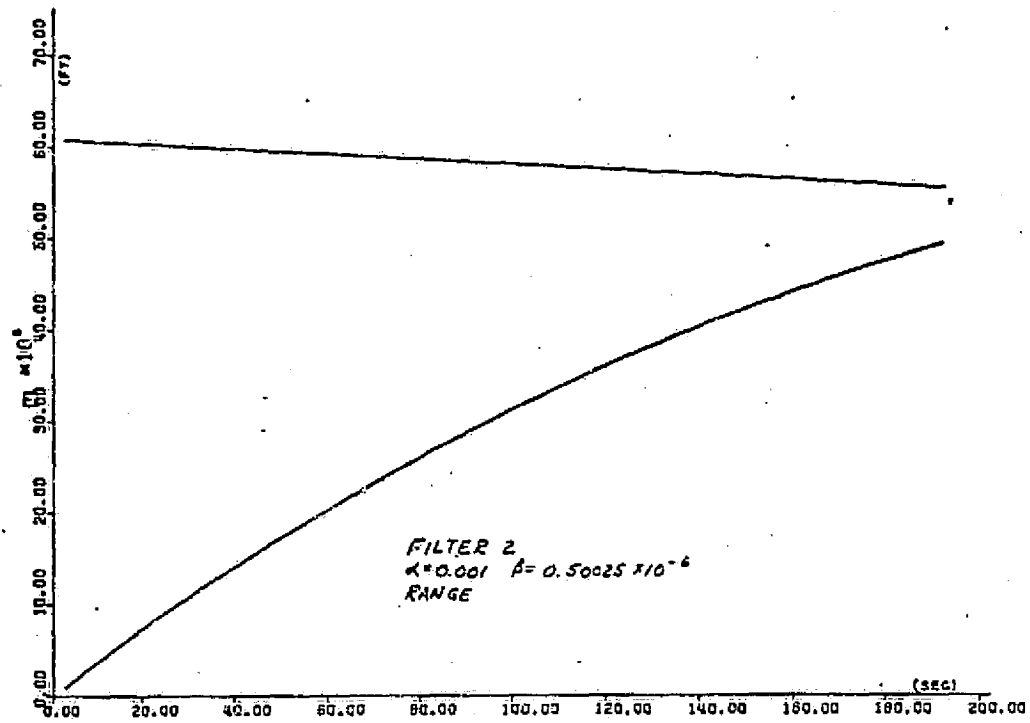


FIGURE 52 - Filter 2 at 6.25 samples/sec.

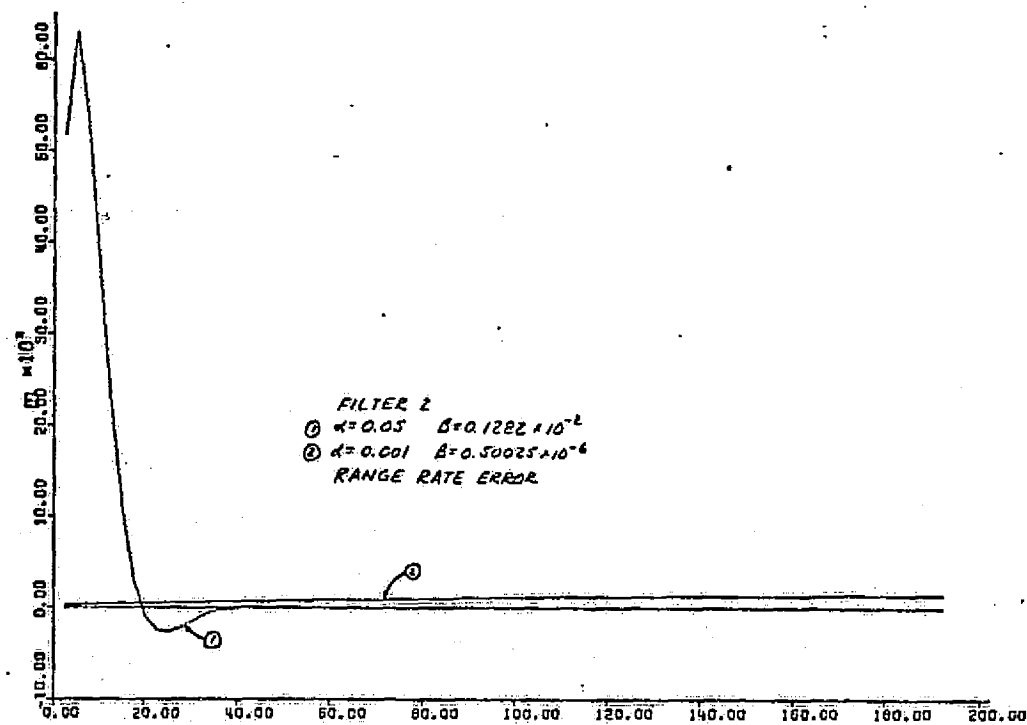
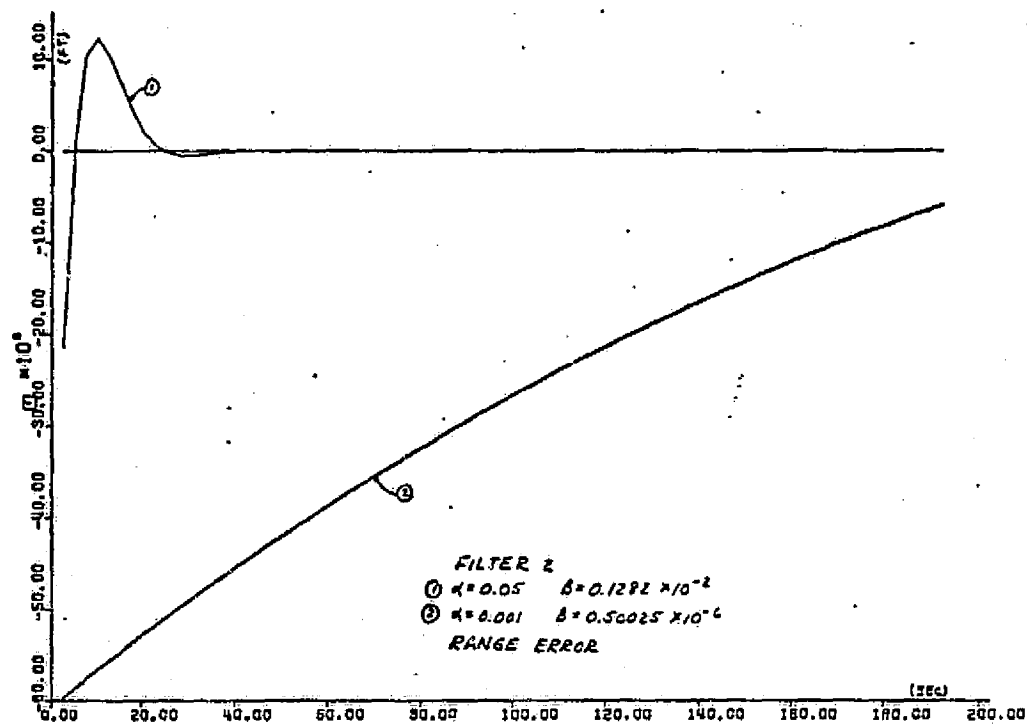


FIGURE 53 - Filter 2 at 6.25 samples/sec.

second duration of the tests. For this reason steady state values for output range and range rate standard deviations are not given.

The range variance reduction ratio is not a function of sampling rate as can be seen by comparing that value for both rates. The range rate variance reduction ratio, however, is proportional to the square of the sampling rate and demonstrates a decrease with a reduction of sampling rate.

Overall, data from the simulation runs for the alpha-beta filter agrees with the variance reduction ratios predicted by Benedict and Bordner.⁵ Deviations become apparent as alpha and especially beta decrease in magnitude.

3. Filter 3

Simulation data for Filter 3 is presented in Tables XXV - XXVIII for 2500 samples/second and Tables XXIX - XXXII for the 6.25 samples/second sampling rate. A graphical display of this data is shown in Figures 54 and 55 for 2500 samples/second and Figures 56 and 57 for 6.25 samples/second.

The N sample smoother employed to derive range in Filter 3 smooths over sixteen samples. That is a sufficient number of samples to provide good filtering. However, its performance is not readily apparent from the data. Since the smoother provides a range value that lags the present sample instant by 7.5 sampling periods, updating is necessary. To accomplish this a correction term is determined from the estimated range rate calculated for that sample instant. This updating introduces error into the estimated range.

For simulation RUN 2, since both the range and range rate segments of Filter 3 are linear systems and the input is a random signal, the estimated range before correction for the 7.5 sample period lag (RE lagging) and the estimated range rate (RDOT) are random signals. The operation of correction for lag can then be described in terms of variances as

$$\sigma_{RE \text{ lagging}}^2 + (K \sigma_{RDOT})^2 = \sigma_{RE}^2 \quad (22)$$

where K is the constant $7.5 * DT$. Using the values for estimated range standard deviation (σ_{RE}) and estimated range rate standard deviation (σ_{RDOT}) listed under RUN 2 in the tables for Filter 3, the standard deviation of the lagging estimated range ($\sigma_{RE \text{ lagging}}$) can be obtained. This value is 3.9291m.(12.8908 ft.) and 7.0905m.(23.2627 ft.) for the 2500 and 6.25 samples/second simulations, respectively. The associated range variance reduction ratios are

$$KR_{2500} = \left(\frac{12.8908}{110.7364} \right)^2 = 1.355 \times 10^{-2} \quad (23)$$

$$KR_{6.25} = \left(\frac{23.2627}{91.3215} \right)^2 = 6.489 \times 10^{-2} \quad (24)$$

This is in agreement with the $\frac{1}{N}$ variance reduction ratio of an N sample smoother. A comparison can be made of these ratios to those given in the tables to demonstrate the effect the updating procedure has on noise rejection in range determination.

TABLE XXV

FILTER 3 RANGE STANDARD DEVIATIONS(FT.) AT 2500 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 3	0.1634	0.1634	110.7364	34.7179	110.7339	34.6758

TABLE XXVI

FILTER 3 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 2500 SAMPLES/SEC.

	RUN 1 - OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 3	0.0	10745.3359	10745.3359

TABLE XXVII

FILTER 3 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 3	0.364×10^{-11}	0.207×10^{-8}	114.486	28399.3	114.4859	28399.3

TABLE XXVIII

FILTER 3 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

	RUN 2		
	KR range variance reduction ratio	KRR range rate variance reduction ratio	τ Convergence Time (Sec.)
FILTER 3	9.829×10^{-2}	9.416×10^3	<1

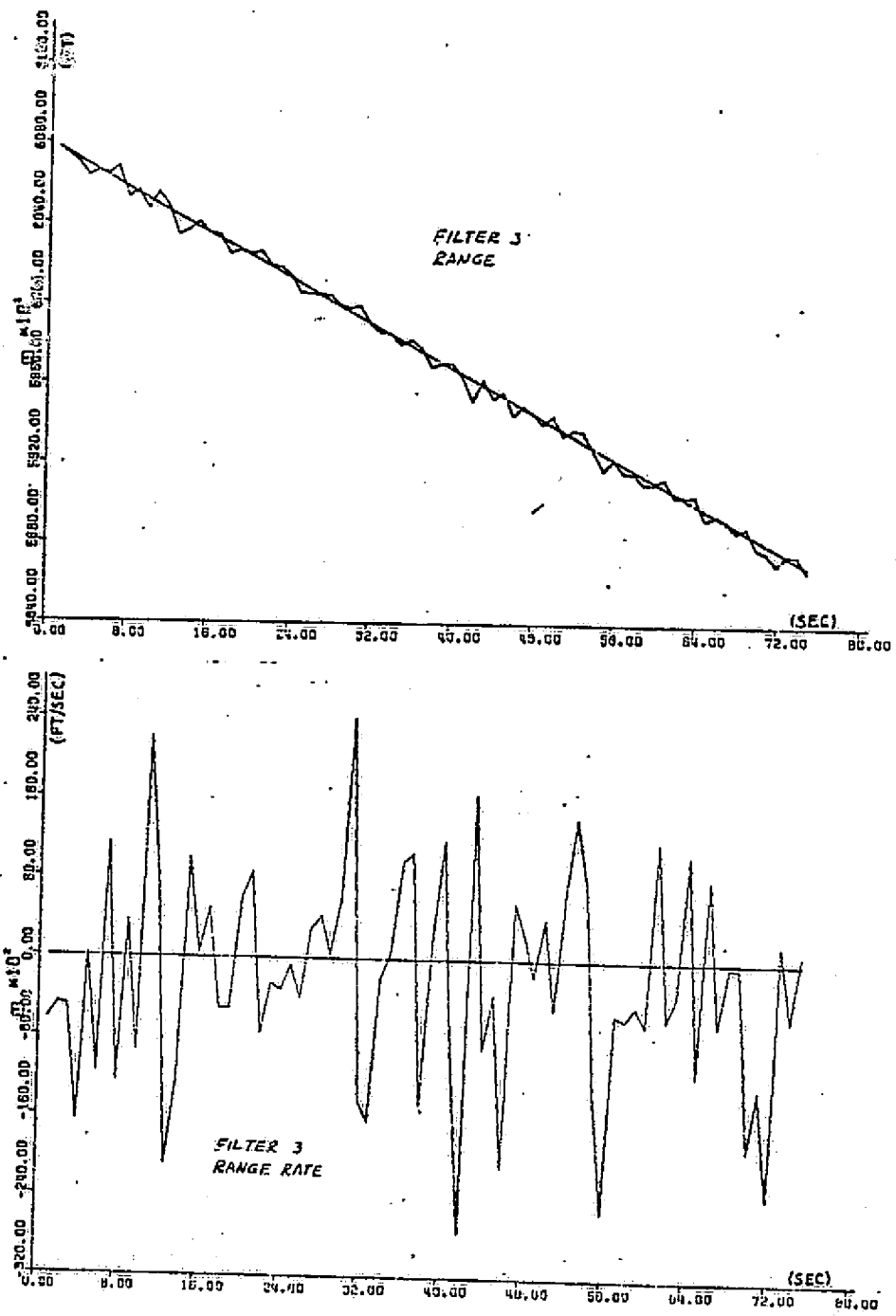


FIGURE 54 - Filter 3 at 2500 samples/sec.

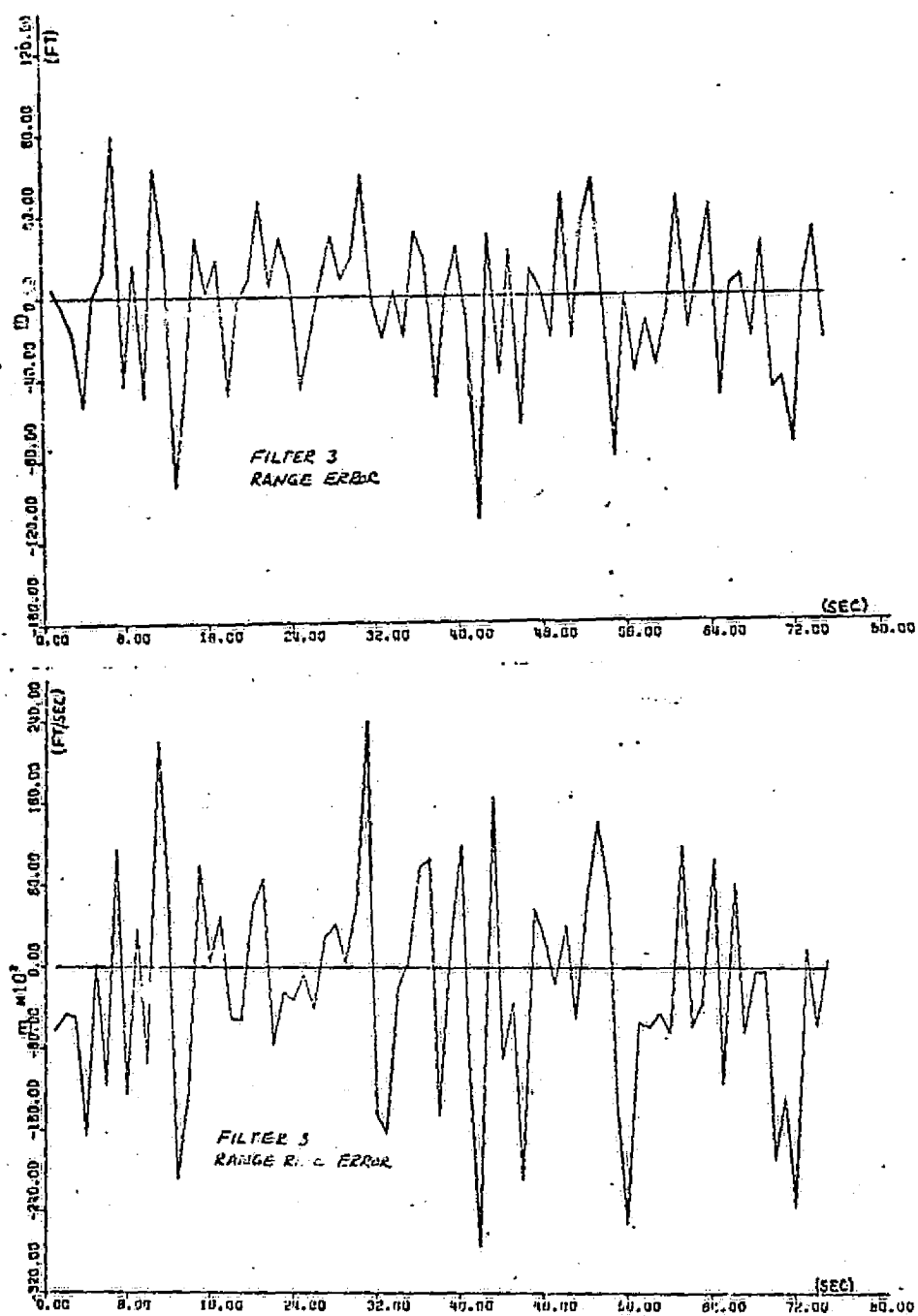


FIGURE 55 - Filter 3 at 2500 samples/sec.

TABLE XXIX

FILTER 3 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 3	65.3068	65.3068	91.3215	36.3516	115.2178	86.9980

TABLE XXX

FILTER 3 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 6.25 SAMPLES/SEC.

	RUN 1 -OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 3	0.0	23.2780	23.2780

TABLE XXXI

FILTER 3 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 6.25 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 3	0.727×10^{-11}	0.296×10^{-11}	104.0342	65.5469	104.0342	65.5469

TABLE XXXII

FILTER 3 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

	RUN 2		
	KR range variance reduction ratio	KRR range rate variance reduction ratio	τ Convergence Time (Sec.)
FILTER 3	1.584×10^{-1}	6.497×10^{-2}	.3

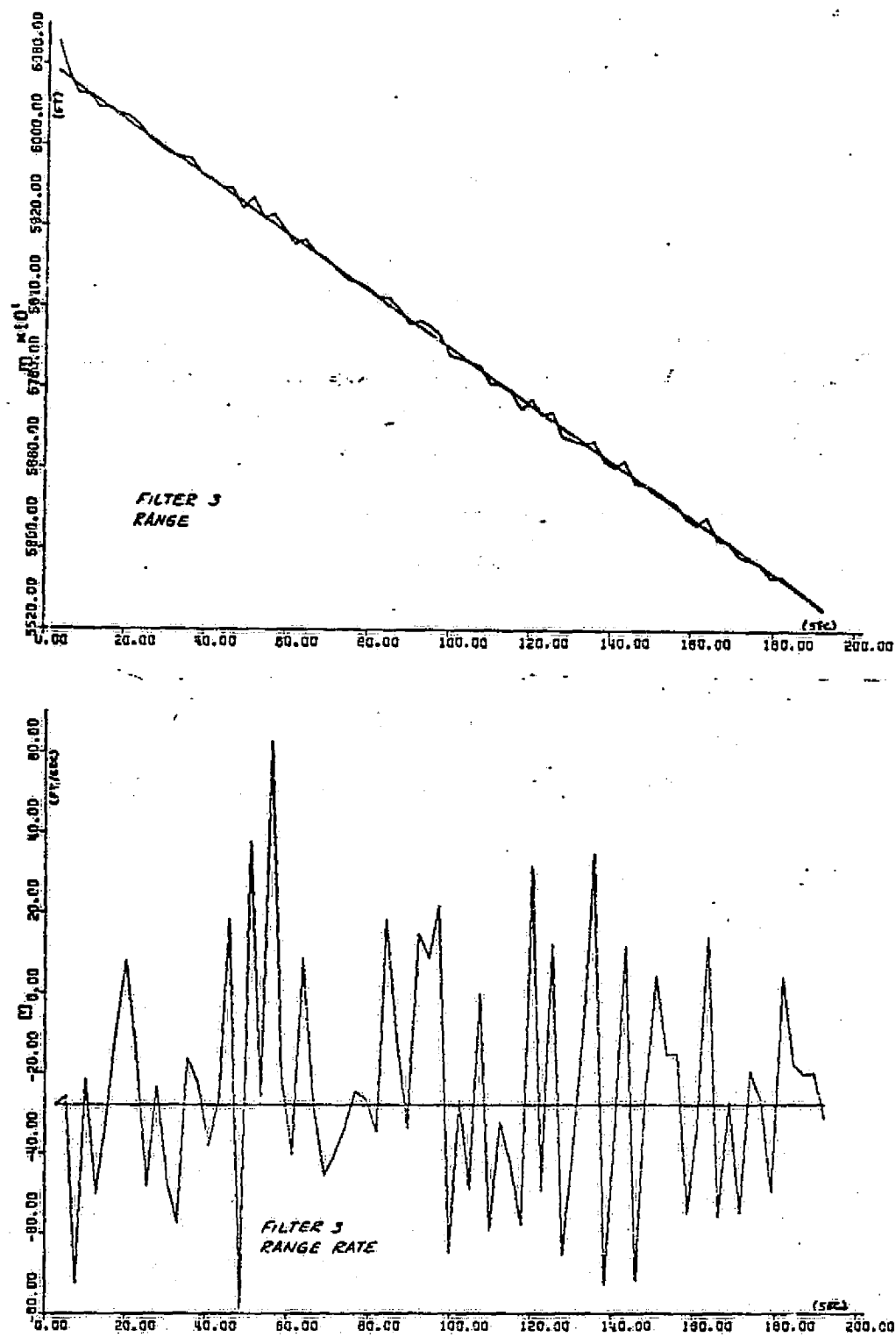


FIGURE 56 - Filter 3 at 6.25 samples/sec.

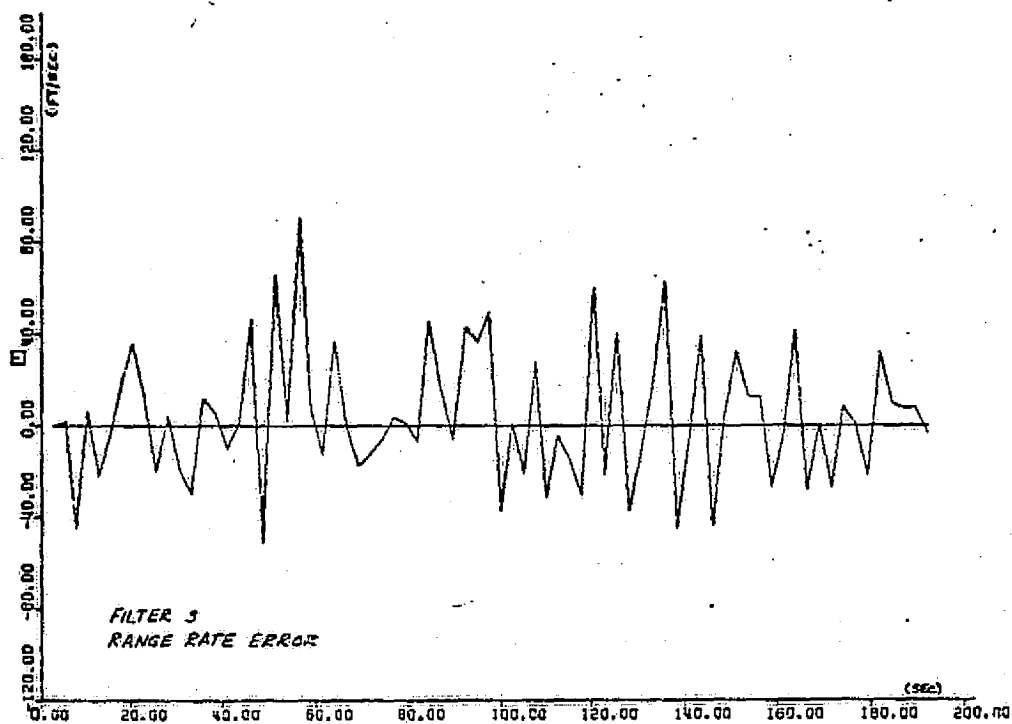
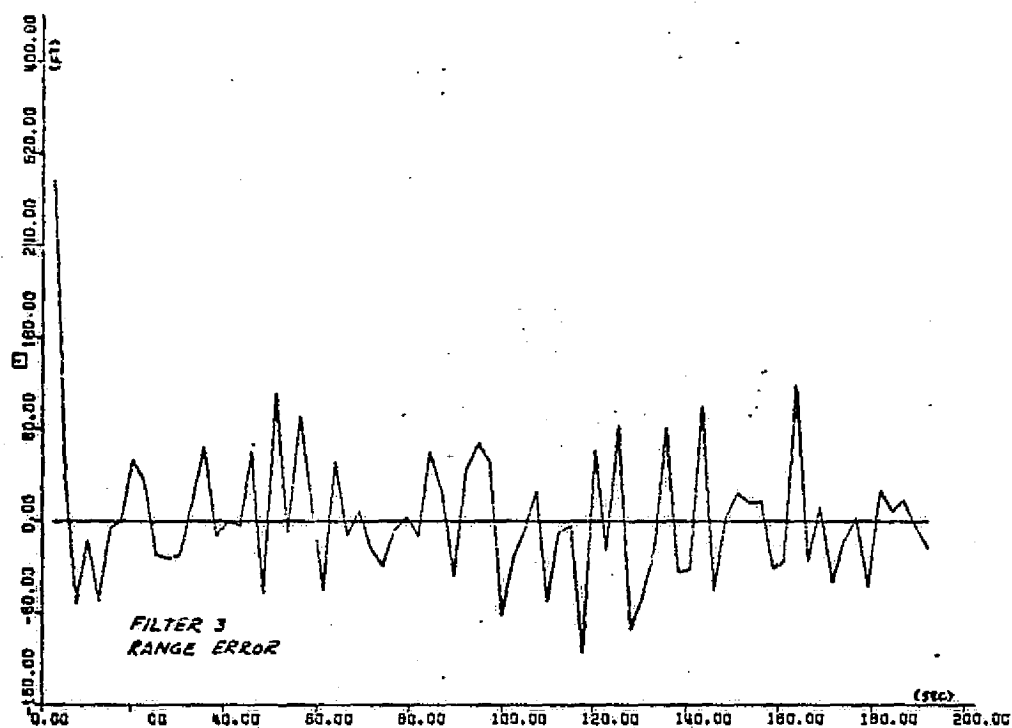


FIGURE 57 - Filter 3 at 6.25 samples/sec.

The range rate determining section of Filter 3 is a cascaded simple average smoother. The range rate variance reduction ratio is given by⁶

$$\frac{\sigma_{\text{RANGE RATE OUTPUT}}^2}{\sigma_{\text{RANGE INPUT}}^2} = \frac{2 n_2}{(n_1 n_2 DT)^2} \quad (25)$$

where n_1 is the number of samples over which the first differences are taken ($n_1 = 12$) and n_2 is the number of first differences that are smoothed ($n_2 = 6$). Simulation runs at both 2500 and 6.25 samples/second support the above relation. Operation at 6.25 samples/second gives improved results due to the dependence of the variance reduction ratio on the square of the sampling period DT.

4. Filter 4

Simulation data for Filter 4 is presented in Tables XXXIII - XXXVI for 2500 samples/second and in Tables XXXVII - XL for 6.25 samples/second. A graphical display of this data is shown in Figures 58 and 59 for 2500 samples/second and in Figure 60 and 61 for 6.25 samples/second.

The digital tracking filter smoothing over four samples has the transfer functions

$$H_R = \frac{RE}{RM} = \frac{1 + z^{-1} + z^{-2} + z^{-3}}{1 - 0.75z^{-1} + 0.25z^{-2} + 0.25z^{-3} + 0.25z^{-4}} \quad (26)$$

and

TABLE XXXIII

FILTER 4 RANGE STANDARD DEVIATIONS(FT.) AT 2500 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 4	0.1634	0.1634	110.7364	144.8444	110.7339	144.8549

TABLE XXXIV

FILTER 4 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 2500 SAMPLES/SEC.

	RUN 1 OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 4	0.0	247907.688	247907.688

TABLE XXXV

FILTER 4 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 4	0.182×10^{-11}	0.320×10^{-8}	290.7632	501279.4	290.7632	501279.375

TABLE XXXVI

FILTER 4 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

	RUN 2		τ Convergence Time (Sec.)
	KR range variance reduction ratio	KRR range rate variance reduction ratio	
FILTER 4	1.712	5.012×10^6	<1

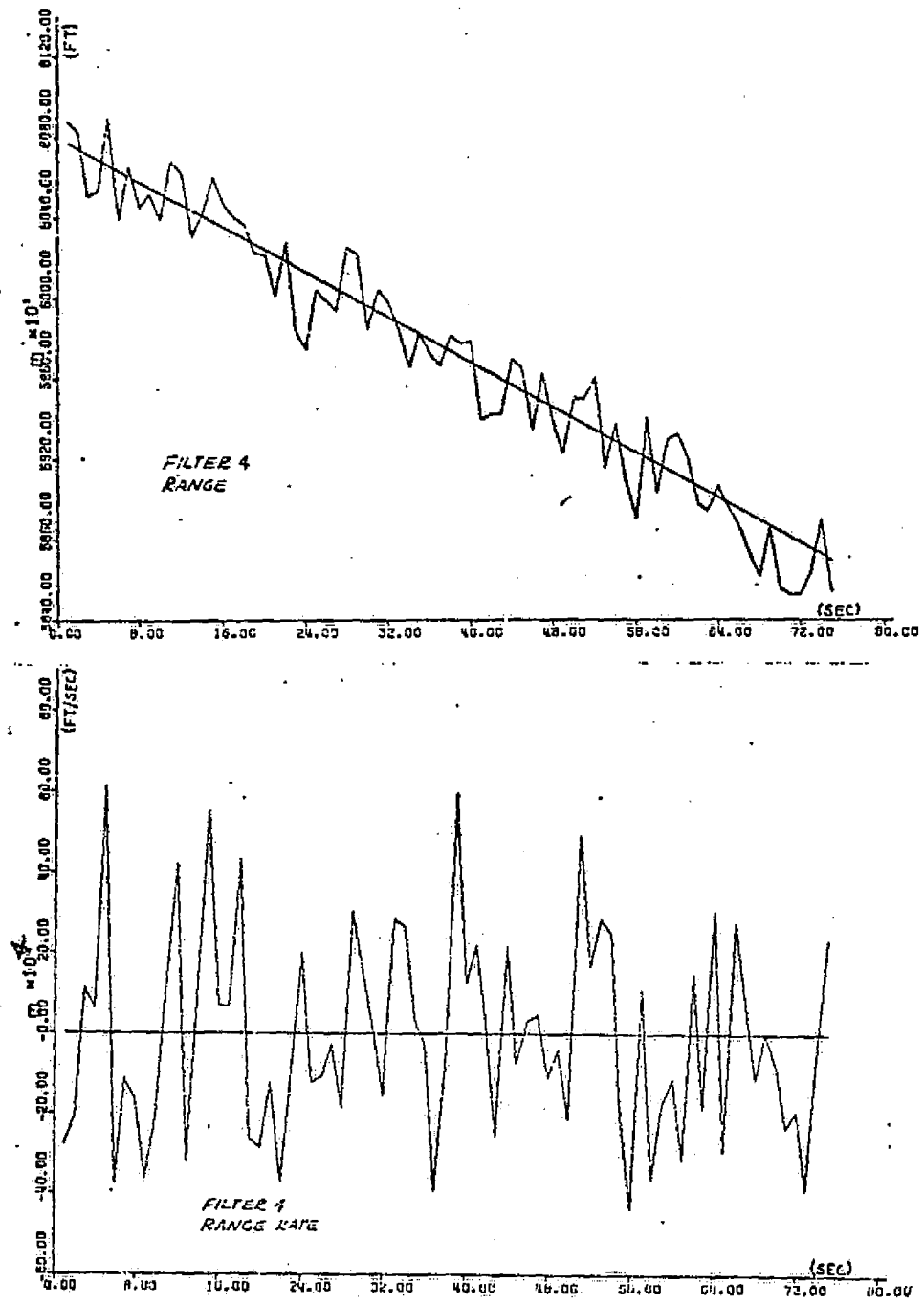


FIGURE 58 - Filter 4 at 2500 samples/sec.

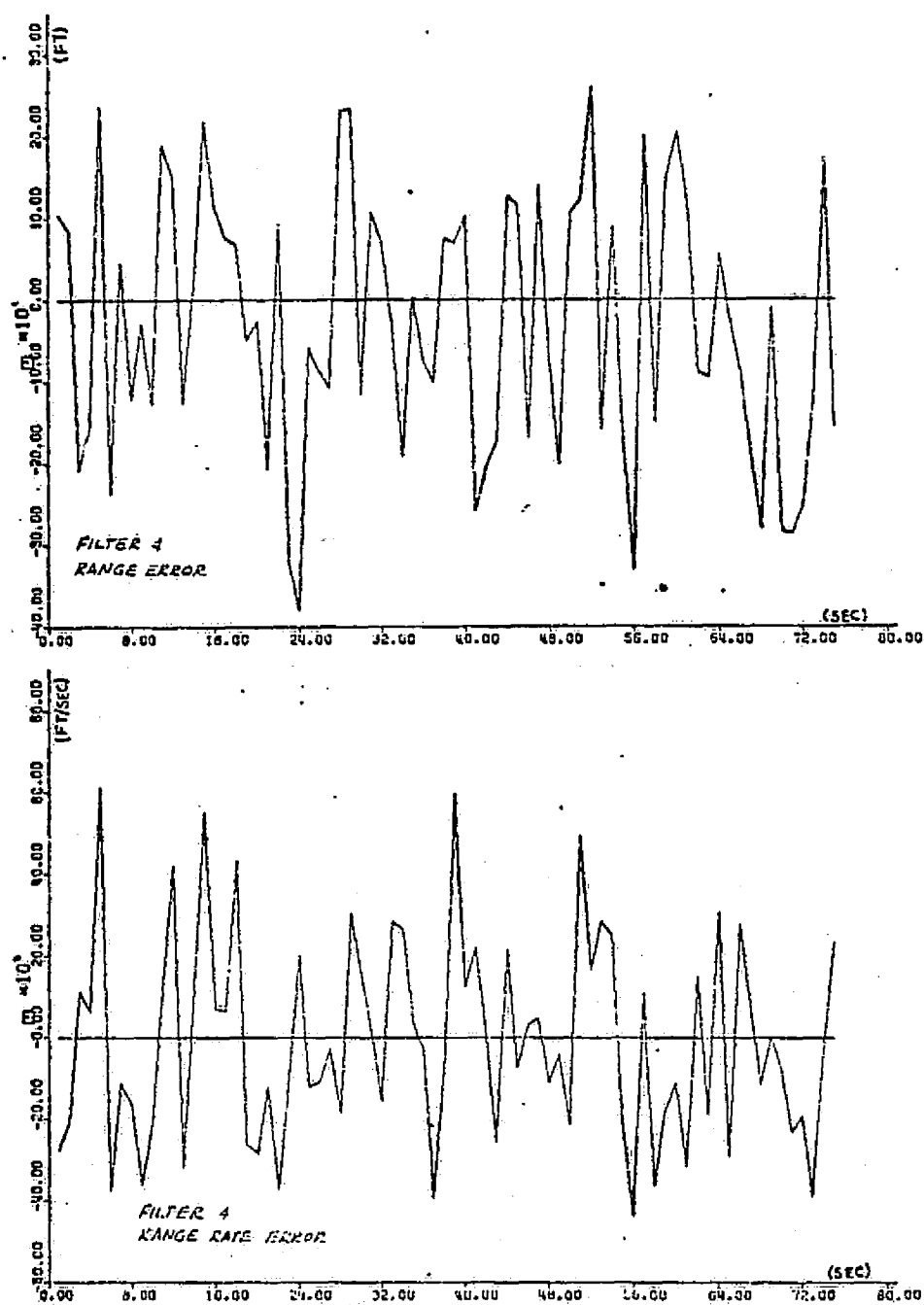


FIGURE 59 - Filter 4 at 2500 samples/sec.

TABLE XXXVII

FILTER 4 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 4	65.3068	65.3068	91.3215	157.1930	115.2178	178.7200

TABLE XXXVIII

FILTER 4 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 6.25 SAMPLES/SEC.

	RUN 1 OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 4	0.0	693.4480	693.4480

TABLE XXXIX

FILTER 4 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 6.25 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 4	0.182×10^{-11}	0.580×10^{-11}	323.3982	1553.83	323.3982	1553.82

TABLE XL

FILTER 4 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

	RUN 2		τ Convergence Time (Sec.)
	KR range variance reduction ratio	KRR range rate variance reduction ratio	
FILTER 4	2.9603	57.6609	2

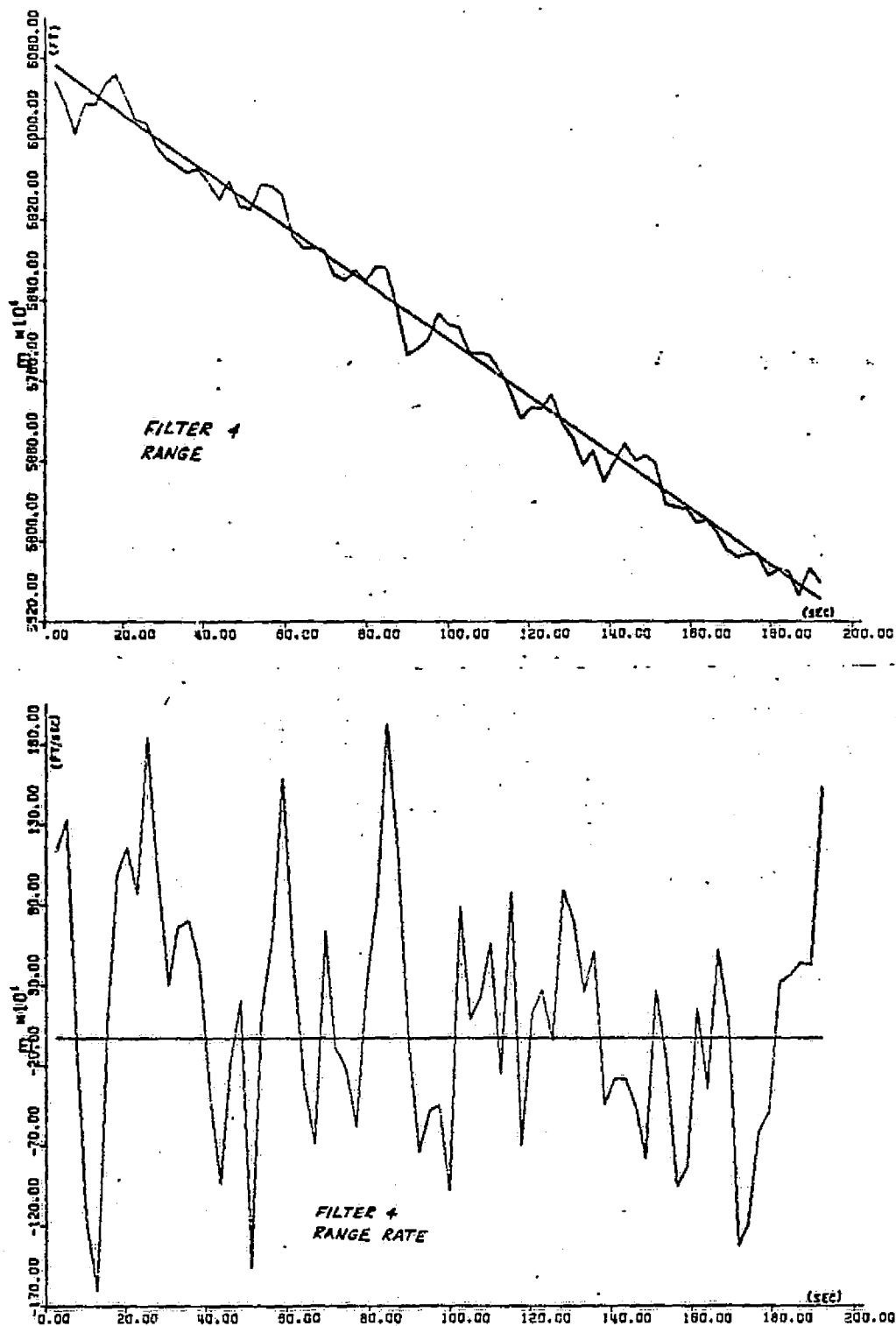


FIGURE 60 - Filter 4 at 6.25 samples/sec.

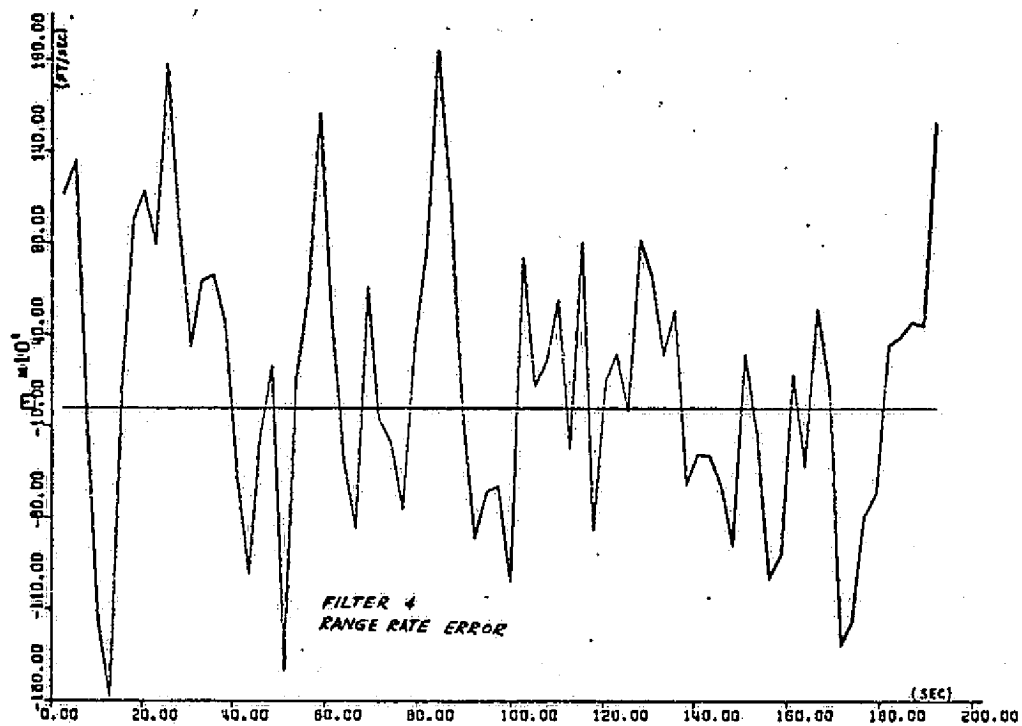
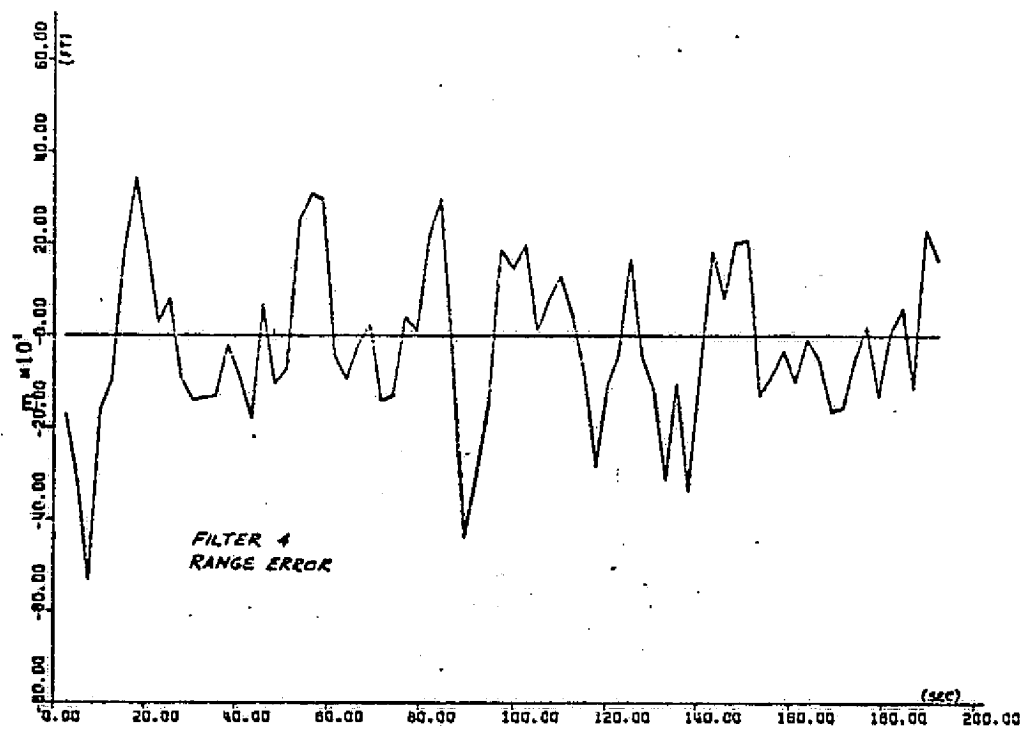


FIGURE 61 - Filter 4 at 6.25 samples/sec.

$$H_{RR}^{\text{RDOT}} = \frac{1-0.75z^{-1}-1.5z^{-2}-2.25z^{-3}-4z^{-4}-2.25z^{-5}-1.5z^{-6}-0.75z^{-7}}{4DT(1-0.75z^{-1}+0.25z^{-2}+0.25z^{-3}+0.25z^{-4})} \quad (27)$$

The variance reduction ratios, KR and KRR, can be determined from the weighting sequences $h_R(i)$ and $h_{RR}(i)$ by

$$K = \sum_{i=0}^{\infty} h^2(i). \quad (28)$$

The program presented in Appendix D is used to evaluate this series. The resulting variance reduction ratios calculated are

$$KR = 48 \quad (29)$$

$$KRR(2500 \text{ samples/sec.}) = 1.47 \times 10^8 \quad (30)$$

$$KRR(6.25 \text{ samples/sec.}) = 916. \quad (31)$$

The variance reduction ratios presented in the tables for Filter 4 are calculated from actual simulation data. Although these are slightly better results than determined above, smoothing over four samples does not offer an adequate level of noise rejection.

5. Filter 5

Simulation data for Filter 5 is presented in Tables XLI - XLIV for 2500 samples/second and in Tables XLV - XLVIII for 6.25 samples/second. A graphical display of this data is presented in Figures 62 and 63 and in Figures 64 and 65 for 6.25 samples/second.

Filter 5 is a digital tracking filter employing skip sampling over sixteen samples. It is characterized by the transfer functions

$$H_R = \frac{RE}{RM} = \frac{23.5 + 23.5z^{-1} + \dots + 23.5z^{-15} - 7.5z^{-16} - \dots - 7.5z^{-31}}{256} \quad (32)$$

and

$$H_{RR} = \frac{RDOT}{RM} = \frac{1 + z^{-1} + \dots + z^{-15} - z^{-16} - \dots - z^{-31}}{256 \text{ DT}} \quad (33)$$

The program to evaluate variance reduction ratios (listed in Appendix D) using

$$K = \sum_{i=0}^{\infty} h^2(i) \quad (28)$$

determined the ratios:

$$KR = 1.72 \times 10^{-1} \quad (34)$$

$$KRR(2500 \text{ samples/sec.}) = 5.74 \times 10^3 \quad (35)$$

$$KRR(6.25 \text{ samples/sec.}) = 3.59 \times 10^{-2} \quad (36)$$

These values are in close agreement with those derived from simulation data and presented in the tables for Filter 5. As with this type of filter the range rate variance reduction ratio

is proportional to the square of the sampling rate while the range variance reduction ratio is independent of sampling rate.

TABLE XLI

FILTER 5 RANGE STANDARD DEVIATIONS(FT.) AT 2500 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 5	2.6123	2.6123	106.1125	39.3340	106.1860	39.3009

TABLE XLII

FILTER 5 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 2500 SAMPLES/SEC.

	RUN 1 OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 5	0.0	5693.6289	5693.6289

TABLE XLIII

FILTER 5 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 2500 SAMPLES/SEC.

	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 5	0.0	0.207×10^{-8}	122.399	15702.340	122.399	15702.340

TABLE XLIV

FILTER 5 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 2500 SAMPLES/SEC.

	RUN 2		T Convergence Time (Sec.)
	KR range variance reduction ratio	KRR range rate variance reduction ratio	
FILTER 5	1.374×10^{-1}	2.879×10^3	<1

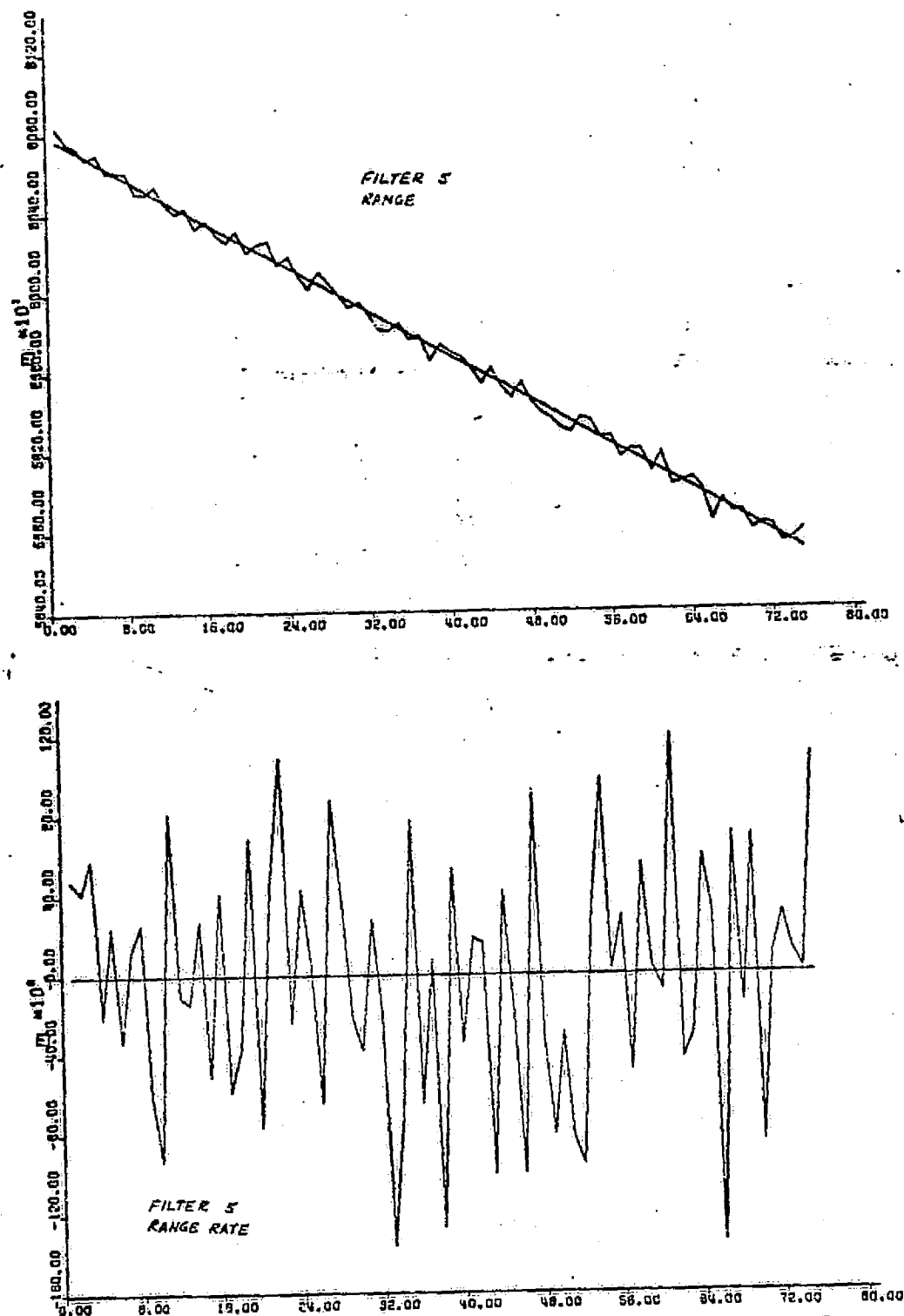


FIGURE 62 - Filter 5 at 2500 samples/sec.

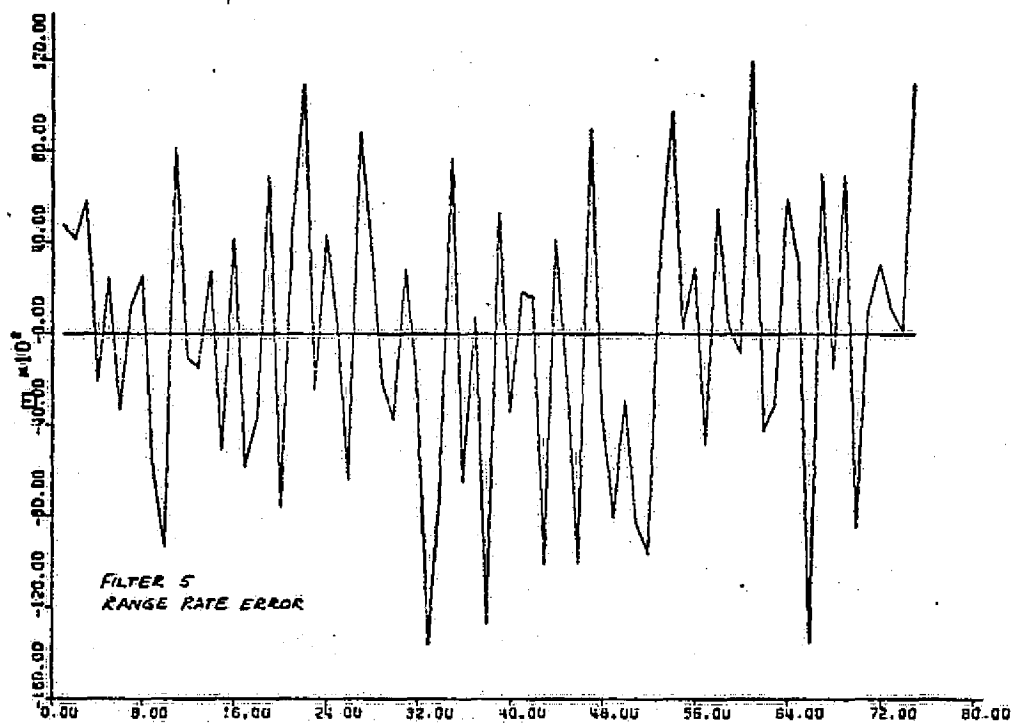
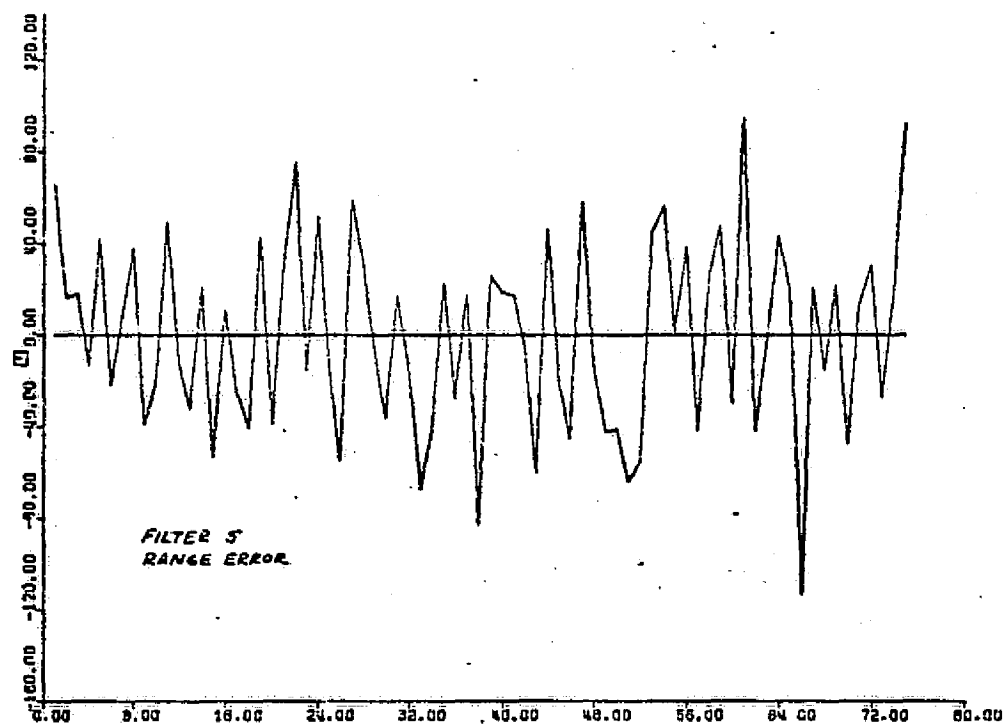


FIGURE 63 - Filter 5 at 2500 samples/sec.

TABLE XLV

FILTER 5 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
FILTER 5	1044.906	1044.96	112.3026	41.1501	1083.432	1044.917

TABLE XLVI

FILTER 5 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 6.25 SAMPLES/SEC.

	RUN 1 OUTPUT RANGE RATE STND. DEV.	RUN 2 OUTPUT RANGE RATE STND. DEV.	RUN 3 OUTPUT RANGE RATE STND. DEV.
FILTER 5	0.0	15.0085	15.0085

TABLE XLVII

FILTER 5 MAXIMUM RANGE(FT.) AND RANGE RATE(FT./SEC.) ERRORS AT 6.25 SAMPLES/SEC.

	RUN 1 input = ramp		RUN 2 input = noise		RUN 3 input = ramp + noise	
	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR	MAX RANGE ERROR	MAX RANGE RATE ERROR
FILTER 5	0.909×10^{-12}	0.278×10^{-11}	122.002	37.450	122.0018	37.450

TABLE XLVIII

FILTER 5 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

	RUN 2		τ Convergence Time (Sec.)
	KR range variance reduction ratio	KRR range rate variance reduction ratio	
FILTER 5	1.342×10^{-1}	1.786×10^{-2}	5

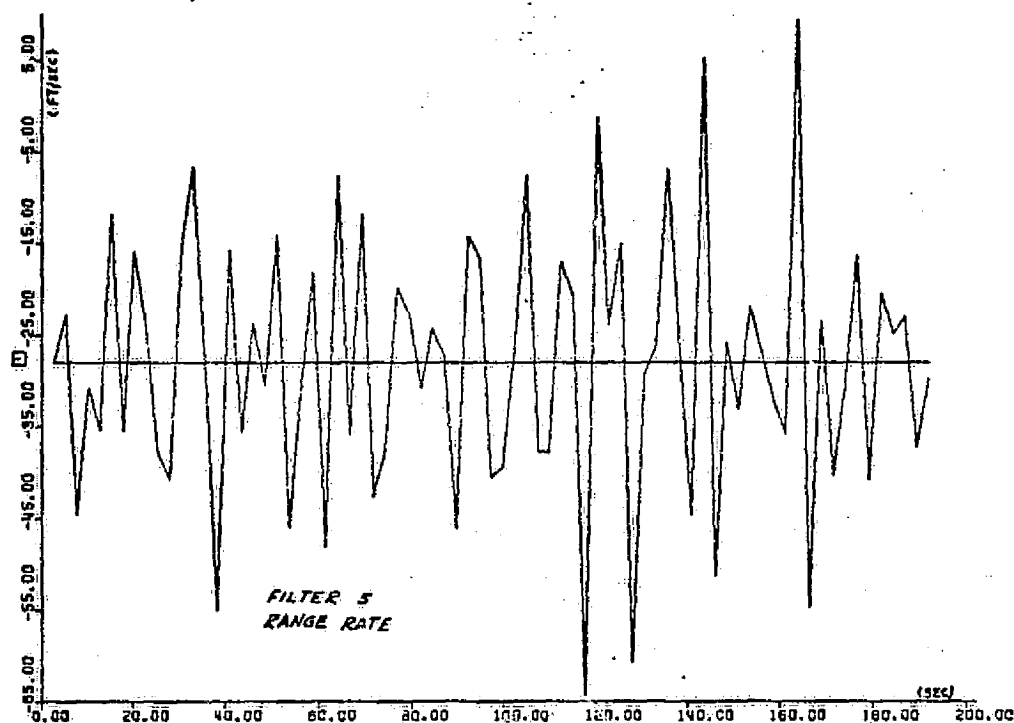
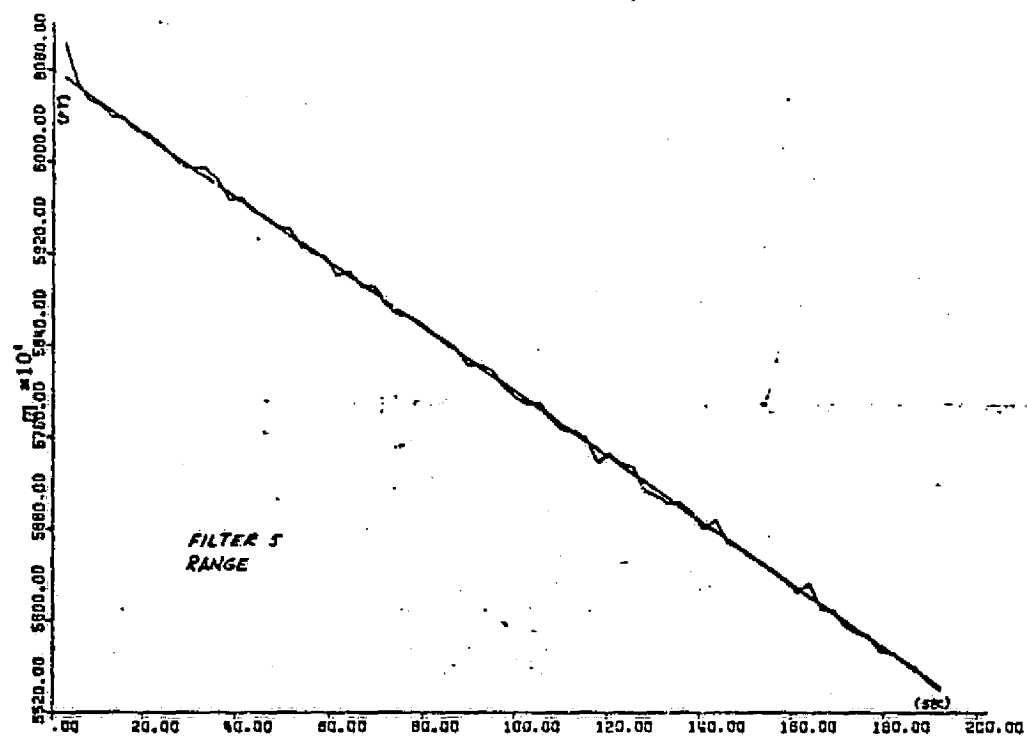


FIGURE 64 - Filter 5 at 6.25 samples/sec.

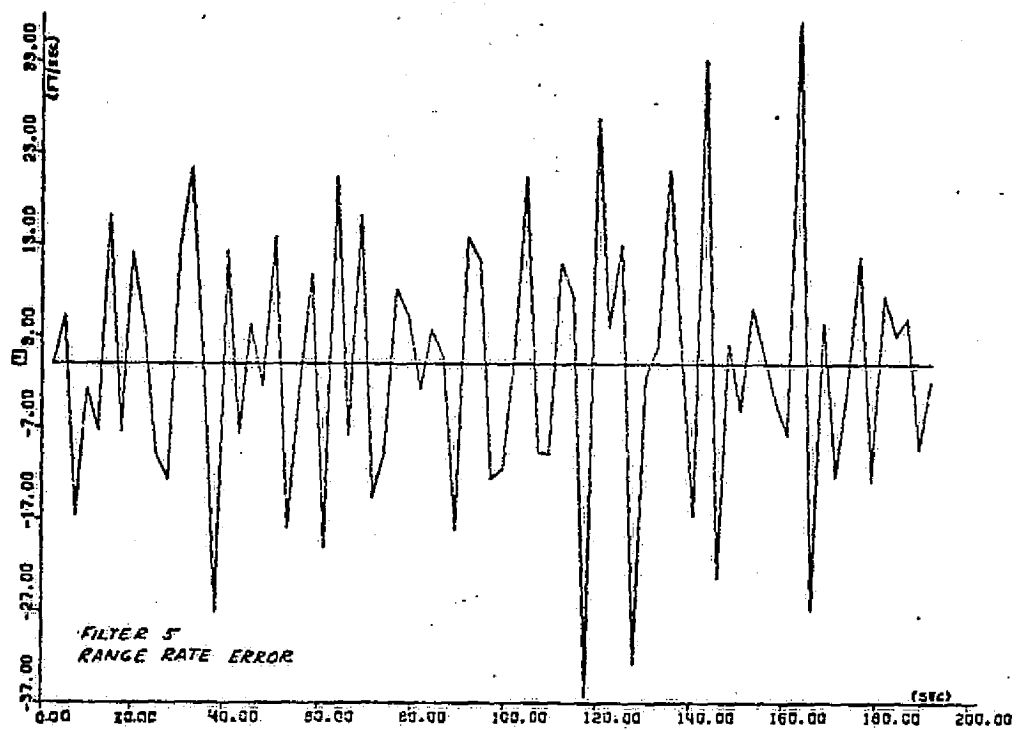
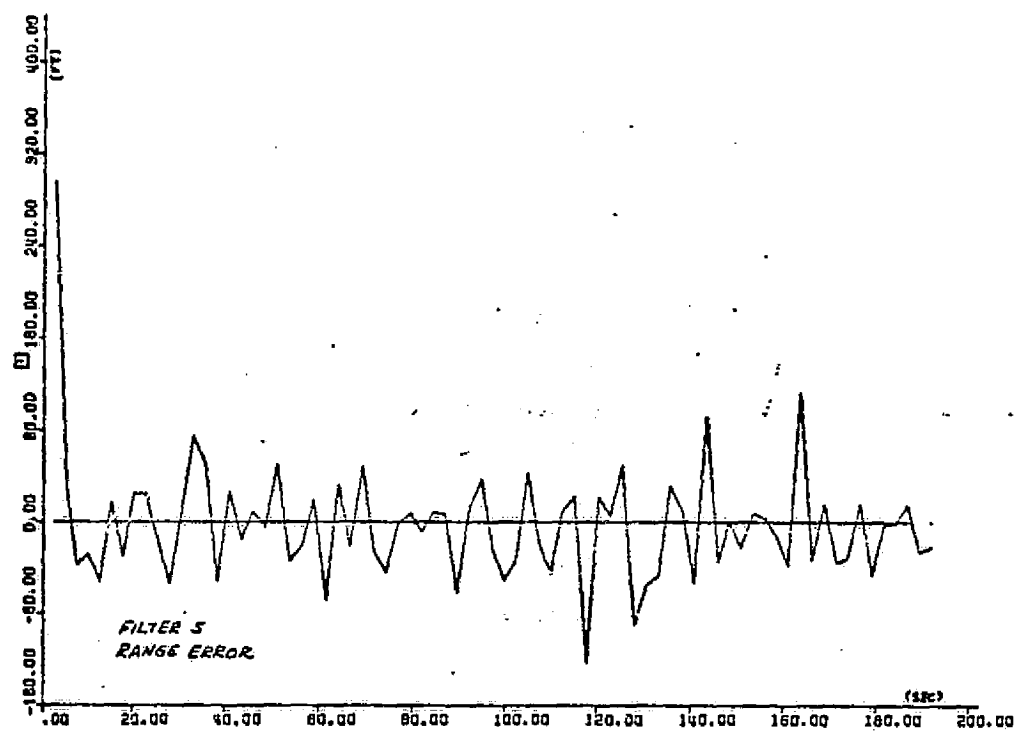


FIGURE 65 - Filter 5 at 6.25 samples/sec.

IV. MICROPROCESSOR SIMULATIONS

Two of the filter designs are selected to be implemented on a microprocessor based system. They are Filter 3 and Filter 5. Simulations are conducted in the same manner as is done on the IBM 370. The same range input signals are employed, and statistical analysis of the outputs is performed as was done in the IBM 370 simulation runs. This allows direct comparisons to be drawn.

The simulations are not conducted in real time. Considerations are taken for execution time, however, and are presented in the test results.

A. Description of Microprocessor System

The microprocessor simulations are conducted on an expanded version of the Motorola MEK 6800D2 Evaluation Kit II, a microcomputer system based on the Motorola MC 6800 microprocessing unit. The details of the revisions made to the system are outlined in the Motorola application note "MEK 6800D2 Microcomputer Kit System Expansion Techniques".⁷ The basic system memory of 512 bytes of static RAM is replaced with 32 kilobytes of dynamic RAM. The MINIBUG III monitor ROM and its associated serial interface are added co-resident to the basic system's JBUG monitor ROM and serial interface. Under MINIBUG control the system is connected via RS-232 interface to a data terminal. Under JBUG control the audio cassette interface provides an inexpensive means of permanent storage.

B. Description of Simulating Programs

The software for implementing the algorithms for Filter 3 and Filter 5 is developed at the assembly language level. The primary considerations in its development are the one byte word size and the lack of a hardware multiplier on the 6800 microprocessor. The consequences of the small word size are that internal values must be represented in multiple byte lengths. Since these values can only be written into or read from memory one byte at a time, execution time is dependent on the number of bytes used to represent a value. For this reason, it is necessary to determine the minimum number of bytes needed to preserve precision at any given point in the program. Although the range input, range output and range rate output are two byte values, internal word size varies from two to five bytes.

Multiplications are handled by a multiplication subroutine. Under worst case conditions this routine requires 2,235 periods of the system clock to return a product. This represents a significant percentage of program execution time.

1. Program FILTER 3

An assembly language listing of the microprocessor realization of Filter 3 is presented in Appendix B. This program accepts a two byte unsigned binary word representing a noisy range value and outputs a two byte unsigned binary word representing the present estimated range and a two byte, two's complement binary word representing the present estimated range rate. When a noisy range value is received it is loaded onto the bottom of a 36 byte stack, REGS, containing the previous 17 noisy range

inputs. A summation is then made of the six most recent entries onto the stack and subtracted from that is the summation of the six oldest entries on the stack. The result of these operations is a three byte, two's complement number stored in the variable TSUM. This value is multiplied by the scaled constant $(1/(72*DT)) * 2^{16}$ to determine the estimated range rate. The 2^{16} scaling factor is included in the constant because the constant is typically less than one. Since integer arithmetic is employed, trying to represent a number between zero and one by a zero or a one would cause a very significant loss of accuracy. To circumvent this situation the constant is premultiplied by a factor of 2^{16} , which is equivalent to shifting the constant two bytes to the left. After the multiplication of TSUM by the constant has been accomplished the two least significant bytes of the five byte product are truncated which is equivalent to a two byte shift back to the right. The magnitude of the estimated range rate is such that it can be represented sufficiently by a two byte value so the most significant byte of the three remaining bytes of the product can also be discarded. The end result is a two byte, two's complement number, RDOT, representing the estimated range rate.

TSUM is multiplied by the scaled constant $(7.5/72)*2^{16}$ to derive a range correction term, RCRCT, which will be used in the range calculations to give the estimated range. In this case, the two lowest significant bytes are retained until after the addition in the range calculations to preserve accuracy. However, the most significant byte can still be discarded leaving

a four byte, two's complement term RCRCT.

To derive the estimated range, RE, the summation of the 16 most recent entries onto the stack REGS is taken. The result is a three byte number located in SUM. This value is multiplied by the scaled constant $(1/16)*2^{16}$ producing a five byte result. The most significant byte is discarded and the remaining four byte value is added to RCRCT. The two most significant bytes of the sum from this addition are the unsigned binary representation of the estimated range.

At this point FILTER 3 receives the next noisy range term and pushes it onto the bottom of the stack REGS causing the oldest entry on the stack to be lost off the top. The process of obtaining the estimated range and range rate is then repeated for these 18 entries on the stack.

Program FILTER 3 utilizes four subroutines: SHFTR, ADD, SUB and MULT. Subroutine SHFTR shifts all entries in the stack REGS up one and loads the latest range error term onto the bottom of the stack. Subroutine ADD adds two three-byte numbers. The numbers are entered into the subroutine through the variables A and B. The three byte sum is returned through the variable SUM. Subroutine SUB is a three byte subtraction routine. Values are entered and returned through the same variables used in subroutine ADD.

All multiplication required by FILTER 3 is performed by subroutine MULT, a two byte by three byte, two's complement multiplication routine. The two byte multiplicand is entered through MLTCD and the three byte multiplier is entered through MLTPR. The five byte product is returned through the variable PROD.

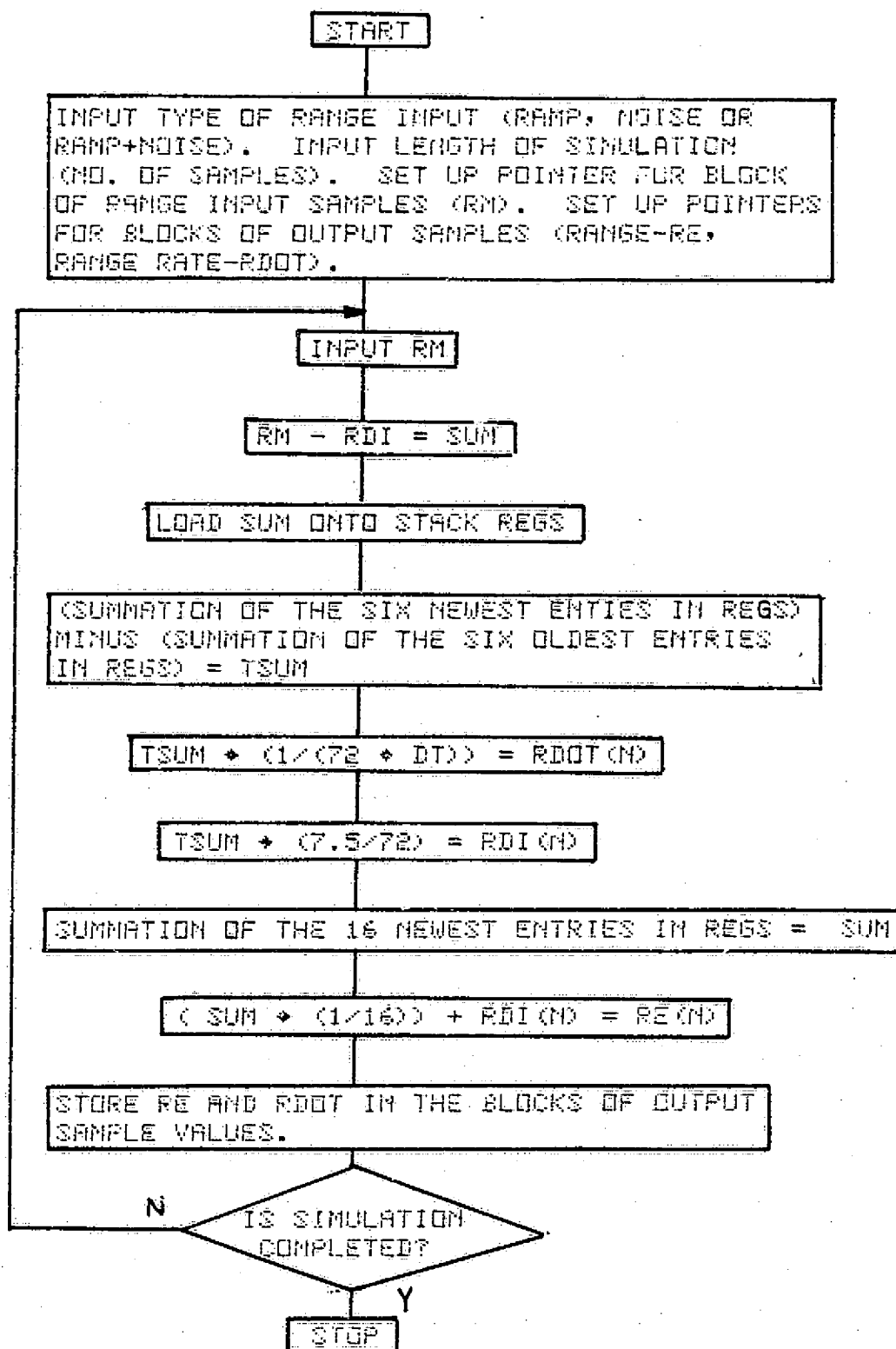


FIGURE 66 - Flowchart for program FILTER 3

Subroutine MULT also contains a rounding routine. Since one of the factors of PROD has been scaled up by 2^{16} , PROD must be scaled down by 2^{16} to obtain the correct result. Rather than truncating, PROD is rounded off to its three most significant bytes. The flag ROUND is used to turn the rounding routine on or off as needed.

To facilitate the display of data the FILTER DATA PRINTER program is appended to the FILTER 3 program. For each sample, the printer program lists the sample number, the value of the noisy range input at that sample, the value of the actual range at that sample and the estimated range and range rate calculated by FILTER 3. This program converts all hexadecimal values to decimal values.

2. Program FILTER 5

An assembly language listing of the microprocessor realization of Filter 5 is presented in Appendix C. This program accepts a two byte unsigned binary word representing a noisy range value. After receiving sixteen of these values the program outputs a two byte unsigned binary word representing the present estimated range and a two byte, two's complement binary word representing the present estimated range rate. Also calculated but not output is a two byte unsigned binary word, RDI, representing the estimated range at a previous point in time. This value is subtracted from each incoming noisy range word to produce a two byte, two's complement binary range error term. These error terms are placed on a 32 byte stack, REGS. When sixteen noisy

range samples have been received, REGS is then full and a summation of its contents is taken. The result is a 3 byte, two's complement binary value, TSUM, which is used to calculate the new estimated range rate, RDOT, and to update the two estimated range values, RE and RDI.

The estimated range rate, RDOT, is derived by multiplying TSUM by the scaled constant $(1/(256*DT))*2^{16}$. Since the constant is typically less than one and integer arithmetic is employed, the scale factor 2^{16} is included to maintain the two byte level of accuracy. The multiplication of the 2 byte constant and 3 byte TSUM results in a 5 byte product. The estimated range rate value is contained in the three most significant bytes of the product due to the scale factor included in the constant. This term can be accurately expressed in two bytes so that the most significant byte can also be discarded yielding a two byte, two's complement, estimated range rate value.

The three byte TSUM is then multiplied by the scaled constant $(1/16)*2^{16}$ and the resulting five byte product is reduced to two bytes in the same manner as described above. This two's complement number is added to the present value of RDI to update it for use in determining the range error terms for the next sixteen noisy range inputs. RDI is a two byte, unsigned binary value.

The estimated range, RE, is updated in the same manner. TSUM is multiplied by the scaled constant $(23.5/256)*2^{16}$. The resulting five byte product is reduced to a two byte, two's complement number and added to the old value of RDI. The result of this addition is a two byte, unsigned binary number representing the current estimated range.

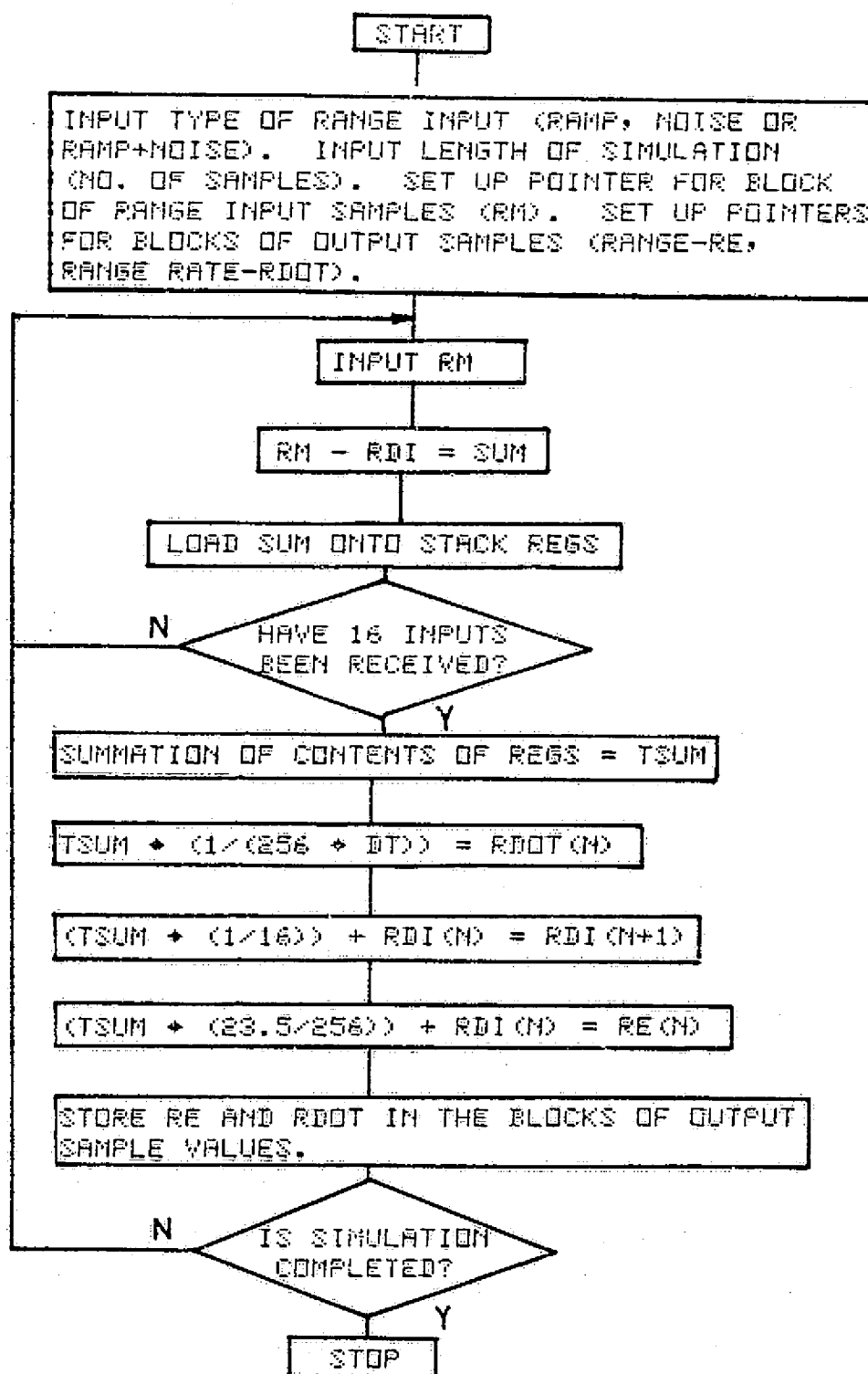


FIGURE 67 - Flowchart for program FILTER 5

Program FILTER 5 utilizes the same four subroutines used by Filter 3. The operation of these subroutines is exactly as described in the section on FILTER 3. Temporary storage is allocated for the stack REGS in such a way that subroutine SHFTR, which is responsible for pushing data onto the stack and shifting the other entries up, need not be changed to reflect the fact that REGS is a 36 byte stack in FILTER 3 and a 32 byte stack in FILTER 5. Both programs utilize premultiplication of constants by 2^{16} so that the rounding routine in subroutine MULT also need not be changed.

Since FILTER 5 outputs estimated range and range rate data only once every sixteen sample periods, the FILTER DATA PRINTER program is modified to display data for only those sample periods for which outputs are generated.

C. Test Conditions

As noted previously, test conditions for the microprocessor simulation runs are very similar to those used in the IBM 370 simulations. Tests are conducted for each of the three range input signals: the ramp, Gaussian noise and the summation of the ramp and noise. The tests are not conducted in real-time but rather a block of data containing sequential samples of the input is resident in memory. The simulating program steps through this block retrieving input values as needed. A block consists of 630 sampled range values taken at the sampling rate. Each value is a two byte unsigned binary word. These are rounded integer representations of the same sampled input values employed in the IBM 370 simulations.

The microprocessor simulations are conducted at the 6.25 samples/second sampling rate only. Although simulation at the higher sampling rate is possible, the speed limitations of the microprocessor would make a real-time system unrealizable. The two filter designs chosen both offer better noise rejection at the lower sampling rate.

D. Test Results

1. Filter 3

Simulation runs for Filter 3 are conducted for a duration of 200 sample periods. The last 50 sample values of the range input, estimated range and estimated range rate are statistically analyzed and presented in Tables XLIX - LI. Also presented is the data for an equivalent run on the IBM 370. The results indicate that the implementation of the filter on a microprocessor based system is accomplished without significant loss of accuracy.

Program FILTER 3 requires 15,777 cycles of the microprocessor system clock to compute estimated range and range rate values after being given a sampled range input. At the 614.4KHz clock rate of the MEK 6800D2 Evaluation Kit, the execution time is 25.68 milliseconds. The 160 millisecond sampling period provides more than adequate time for filter calculations, hence, this program could be employed in a real-time system.

2. Filter 5

Filter 5 simulations are conducted for a duration of 608 sample periods. Since this filter returns an output only once

TABLE XLIX

FILTER 3 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

FILTER 3	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
MICROPROCESSOR	65.380	65.375	103.531	53.105	102.647	61.596
IBM 370	65.3068	65.3068	105.9378	53.1095	108.1119	68.1752

TABLE L

FILTER 3 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 6.25 SAMPLES/SEC.

FILTER 3	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	INPUT RANGE RATE STND. DEV.	INPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.
MICROPROCESSOR	0.0	0.0	33.288	33.288	33.132	33.132
IBM 370	0.0	0.0	33.1917	33.1917	33.1917	33.1917

TABLE LI

FILTER 3 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

FILTER 3	RUN 2		
	input = noise		
	KR VARIANCE RANGE REDUCTION RATIO	KRR VARIANCE RANGE RATE REDUCTION RATIO	τ CONVERGENCE TIME (SEC.)
MICROPROCESSOR	2.632×10^{-1}	1.034×10^{-1}	3
IBM 370	2.513×10^{-1}	9.817×10^{-2}	3

for every sixteen sampled inputs, 38 output values are determined during a simulation run. The filter converges to steady state after two execution cycles (32 sample periods), therefore, the last 35 output values represent steady state and are statistically analyzed. This set of samples is of sufficient size to insure statistically accurate results. Data for the microprocessor simulations of Filter 5 and an equivalent IBM 370 run are presented in Tables LII - LIV. Here again, as in the microprocessor realization of Filter 3, results agree with those obtained on the IBM 370.

The worst case execution time for an "OFF" sample period, when no filter outputs are produced, is 907 cycles of the microprocessor system clock. The execution time for an "ON" sample period, when filter output values are determined, is 11,639 cycles of the system clock. At the 614.4KHz clock rate of the MEK 6800D2 Evaluation Kit, execution times are 1.48 and 18.94 milliseconds for "OFF" and "ON" sample periods, respectively. However, the 6800 family is capable of clock rates up to 2MHz which would yield execution times of 0.45 and 5.82 milliseconds for "OFF" and "ON" sample periods, respectively.

To offset the problem of smoothed range and range rate data being available only every sixteenth sample instant, it has been proposed that sixteen filters be operated in parallel with each filter lagging the next by one sample period.⁸ In this manner an output from one of the filters would be available at each sample instant. At the 6.25 samples/second sampling rate

TABLE LII

FILTER 5 RANGE STANDARD DEVIATIONS(FT.) AT 6.25 SAMPLES/SEC.

FILTER 5	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.	INPUT RANGE STND. DEV.	OUTPUT RANGE STND. DEV.
MICROPROCESSOR	734.514	734.516	91.516	32.489	755.846	735.662
IBM 370	734.5007	734.5007	111.0584	35.4184	739.8486	735.4358

TABLE LIII

FILTER 5 RANGE RATE STANDARD DEVIATIONS(FT./SEC.) AT 6.25 SAMPLES/SEC.

FILTER 5	RUN 1		RUN 2		RUN 3	
	input = ramp		input = noise		input = ramp + noise	
	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.	OUTPUT RANGE RATE STND. DEV.
MICROPROCESSOR	0.0		11.543		11.508	
IBM 370	0.0		12.5710		12.5710	

TABLE LIV

FILTER 5 RANGE AND RANGE RATE VARIANCE REDUCTION RATIOS AT 6.25 SAMPLES/SEC.

FILTER 5	RUN 2 input = noise		T CONVERGENCE TIME (SEC.)
	KR VARIANCE RANGE REDUCTION RATIO	KRR VARIANCE RANGE RATE REDUCTION RATIO	
MICROPROCESSOR	1.260×10^{-1}	1.591×10^{-2}	5
IBM 370	1.017×10^{-1}	1.281×10^{-2}	5

the execution time of program FILTER 5 is only a fraction of the sampling period. Hence, the same processor can be used for all sixteen filters. In a given sampling period fifteen filters would be at "OFF" sampling periods while one would be at its "ON" sampling period. Operating with the 614.4KHz system clock, total execution time would be 42.62 milliseconds, slightly over a quarter of the sampling period.

CONCLUSIONS AND RECOMMENDATIONS

The IBM 370 simulations demonstrate that Filter 1 offers improved noise reduction at higher sampling rates in both the range and range rate modes for both constant gains and bandlimited differentiator. Maneuver-following capabilities, as measured by convergence time, are independent of sampling rate. Noise reduction for Filter 2, Filter 3 and Filter 5 is independent of sampling rate in the range mode and increases with decreasing sampling rate in the range rate mode. These filters all demonstrate increased maneuver-following capability at higher sampling rates. Filter 4 does not provide sufficient noise reduction to be considered further.

At the low sampling rate (6.25 samples/second) and with a short convergence time, Filter 5 has greater noise reduction than Filter 3. If parameter pairs are chosen for Filter 1 and Filter 2 to yield an equally short convergence time, they too do not provide the level of noise reduction of Filter 5. However, if a longer convergence time is allowed, the parameter pairs of Filter 1 and Filter 2 can be chosen to provide higher levels of noise reduction than Filter 5. In this case, Filter 1 (bandlimited differentiator) provides greater noise reduction than Filter 2 for a given convergence time.

The advantage of Filter 1 becomes more apparent at the higher sampling rate. Convergence time is unaffected but noise reduction increases with sampling rate. Hence, the capabilities of this filter are determined by the limitations of the processor with which it is implemented. Future generations of processors will offer faster clock

rates which will allow a corresponding increase in sampling rates and result in better filter performance.

Filter performance can be summarized:

- Filter 1: τ - independent of sampling rate
 - dependent on K_1 , K_2
 KR - decreases with increasing sampling rate
 - dependent on K_1 , K_2
 KRR - decreases with increasing sampling rate
 - dependent on K_1 , K_2
 The tradeoff between τ and the ratios KR and KRR is controlled by K_1 , K_2 (b in the BLD).
- Filter 2: τ - decreases with increasing sampling rate
 - dependent on α , β
 KR - independent of sampling rate
 - dependent on α , β
 KRR - decreases with decreasing sampling rate
 - dependent on α , β
 The tradeoff between τ and the ratios KR and KRR is controlled by α , β .
- Filter 3: τ - decreases with increasing sampling rate
 KR - dependent on sampling rate slightly, due to updating
 KRR - decreases with decreasing sampling rate
- Filter 5: τ - decreases with increasing sampling rate
 KR - independent of sampling rate
 KRR - decreases with decreasing sampling rate.

Future study should be directed toward implementation of Filter 1 or Filter 2 at a higher sampling rate with a 16 bit microprocessor having a hardware multiplier. It would also be worthwhile to investigate different methods of approximating the continuous system of Filter 1 with discrete systems. The method used in this study is only one way of mapping from the s-plane to the z-plane. Other methods should be examined to determine if any offer mappings more suitable for this application.

REFERENCES

1. Budak, A., Passive and Active Network Analysis and Synthesis, 1st ed., Houghton Mifflin, Boston, 1974, pp. 527-528.
2. Benedict, T. R and Bordner, G. W., "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations", IRE Transactions on Automatic Control, Vol. AC-7, No. 4 (July, 1962), pp. 27-32.
3. Sklansky, J., "Optimizing the Dynamic Parameters of a Track-While-Scan System", RCA Review, June 1957, Vo. 18, 00. 167-168.
4. Final Report Radar Range Data Signal Enhancement Tracker, NAS 9-13739, April 1975, National Aeronautics and Space Administration, Johnson Space Center, Houston, Texas, p. 2 - 23.
5. Benedict, p. 29.
6. Blackman, R. B., Linear Data-Smoothing and Prediction in Theory and Practice, 1st ed., Addison-Wesley, Reading, Massachusetts, 1965, p.101.
7. Harrington, W., MEK 6800D2 Microcomputer Kit System Expansion Techniques, AN-771, 1977, Motorola Semiconductor Products, Phoenix, Arizona.
8. Final Report Radar Range Data Signal Enhancement Tracker, p. 2-25.
9. Lam, H. Y-F., Analog and Digital Filters: Design and Realization, 1st ed., Prentice Hall, Englewood Cliffs, New Jersey, 1979, p. 525.

BIBLIOGRAPHY

- Benedict, T. R. and Bordner, G. W., "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations", IRE Transactions on Automatic Control, Vol. AC-7, No. 4 (July, 1962), pp. 27-32.
- Blackman, R. B., Linear Data-Smoothing and Prediction in Theory and Practice, 1st ed., Addison-Wesley, Reading, Massachusetts, 1965.
- Budak, A., Passive and Active Network Analysis and Synthesis, 1st ed., Houghton Mifflin, Boston, 1974.
- Cadzow, J. A., Discrete Time Systems, 1st ed., Prentice-Hall, Englewood Cliffs, New Jersey, 1973.
- Final Report Radar Range Data Signal Enhancement Tracker, NAS 9-13739, April 1975, National Aeronautics and Space Administration, Johnson Space Center, Houston, Texas.
- Harrington, W., MEK 6800D2 Microcomputer Kit System Expansion Techniques, AN-771, 1977, Motorola Semiconductor Products, Phoenix, Arizona.
- Lam, H. Y-F., Analog and Digital Filters: Design and Realization, 1st ed., Prentice-Hall, Englewood Cliffs, New Jersey, 1979.
- Sklansky, J., "Optimizing the Dynamic Parameters of a Track-While-Scan System", RCA Review, June 1957, Vol. 18.

APPENDIX A

The following pages contain a FORTRAN listing of program FLT, the associated CALC subroutines for each of the five filters and the two plotting subroutines, PLOTB and MCPLOT.

```

C
C   THIS PROGRAM SIMULATES A DIGITAL FILTER.
C   THE PARAMETERS OF THE FILTER AND INITIAL CONDITIONS ARE READ IN
C   ON NAMEDLIST LIST.
C   THE PLOTTING PARAMETERS ARE READ IN ON NAMEDLIST DIM.
C   THE PARAMETERS OF THE RANDOM NUMBER GENERATOR ARE READ IN CN
C   NAMEDLIST XRAND.
C   SUBROUTINE CALC PERFORMS ALL FILTER CALCULATIONS.
C   SUBROUTINE GGNRF GENERATES THE NORMALLY DISTRIBUTED RANDOM VARIABLES
C   THAT ARE USED AS RANGE NOISE.
C   SUBROUTINE BECDRI CALCULATES THE MEAN AND STANDARD DEVIATION OF THE
C   INPUT RANGE, OUTPUT RANGE AND OUTPUT RANGE RATE.
C   SUBROUTINE PLOTB FORMATS DATA TO USE THE PRINTER FOR PLOTTING.
C   SUBROUTINE MCPLT FORMATS DATA FOR THE CALCOMP PLOTTER.
C
C   PROGRAM FLT MAKES THREE FILTER SIMULATION RUNS FOR EACH SET OF
C   FILTER PARAMETERS. A NEW SET OF FILTER PARAMETERS IS READ IN EVERY
C   TIME NAMEDLIST LIST IS READ. THE FIRST SIMULATION RUN IS MADE WITH A
C   RAMP INPUT. DATA FROM THIS RUN IS TABULATED IN A TABLE ONLY (NO
C   PLOTS). THE SECOND SIMULATION RUN IS MADE WITH AN INPUT OF GAUSSIAN
C   NOISE. DATA FROM THIS RUN IS ALSO TABULATED IN A TABLE ONLY.
C   THE THIRD SIMULATION RUN IS MADE WITH THE INPUT OF A NOISY RAMP.
C   DATA FROM THIS RUN IS TABULATED IN A TABLE AND ALSO PLOTTED.
C   THE PLOTS RETURNED ARE DETERMINED BY THE PARAMETERS IN NAMEDLIST DIM.
C   ALL PLOTS ARE OF A FUNCTION VERSUS TIME.
C
C   NAMEDLIST LIST
C       LIST IS READ AT THE BEGINNING OF EACH RUN THROUGH THE MAIN PROGRAM
C       DRE = INITIAL ESTIMATE OF RANGE
C       DRDOT = INITIAL ESTIMATE OF RANGE RATE
C       DT = TIME BETWEEN INPUT SAMPLES
C       K1 AND K2 = PARAMETERS OF FILTER
C       DR = INITIAL RANGE
C       DV = INITIAL RANGE RATE

```

```

C      NDN = NUMBER OF FILTERS TO BE SIMULATED (THE NUMBER OF TIMES
C      NAMELIST LIST IS TO BE READ.)
C      IDI = NUMBER OF SAMPLE PERIODS BETWEEN THE READING OUT OF
C      RESULTS
C      IDRI = NUMBER OF RESULTS TO BE READ OUT
C
C      NAMELIST DIM
C      DIM IS READ ONLY ONCE.
C      PLR = 1 SPECIFIES THE OUTPUT OF TWO RANGE PLOTS. BOTH
C      PLOTS CONTAIN THREE FUNCTIONS-R,RM AND RE. THE FIRST
C      PLOT USES A SCALE CHOSEN FROM THE MAX AND MIN FUNCTION
C      VALUES. THE SECOND USES BU1 AND BL1 AS THE PLOT'S MAX
C      AND MIN VALUES. IF PLR IS NOT SPECIFIED NO RANGE PLOTS
C      ARE GIVEN. THE PLOTS ARE GENERATED AFTER THE THIRD
C      FILTER SIMULATION RUN FOR EACH FILTER.
C      PLRR = 1 SPECIFIES THE OUTPUT OF TWO RANGE RATE PLOTS. BOTH
C      PLOTS CONTAIN TWO FUNCTIONS-V AND RDOT. THE FIRST
C      PLOT USES A SCALE CHOSEN FROM THE MAX AND MIN FUNCTION
C      VALUES. THE SECOND USES BU2 AND BL2 AS THE PLOT'S MAX
C      AND MIN VALUES. IF PLR IS NOT SPECIFIED NO RANGE RATE
C      PLOTS ARE GIVEN. THE PLOTS ARE GENERATED AFTER THE
C      THIRD FILTER SIMULATION RUN FOR EACH FILTER.
C      PLRER = 1 SPECIFIES THE OUTPUT OF ONE OR TWO RANGE ERROR
C      PLOTS. UNLIKE THE FIRST TWO SETS OF PLOTS, THESE ARE
C      NOT GENERATED UNTILL ALL FILTERS HAVE BEEN RUN.
C      HENCE, THE RANGE ERROR FOR THE THIRD SIMULATION RUN
C      FOR EACH FILTER APPEARS ON THE SAME PLOT. THIS IS
C      FOR COMPARISON PURPOSES. THE FIRST
C      PLOT USES A SCALE CHOSEN FROM THE FUNCTION'S MAX AND
C      MIN VALUES. THE SECOND USES BU3 AND BL3 AS THE
C      PLOT'S MAX AND MIN VALUES. IF PLRER IS NOT SPECIFIED
C      NO RANGE ERROR PLOTS ARE GENERATED.
C      PLRRER = 1 SPECIFIES THE OUTPUT OF ONE OR TWO RANGE RATE
C      ERROR PLOTS. THESE ARE ALSO NOT GENERATED UNTIL ALL

```

C FILTERS HAVE BEEN RUN. HENCE, THE RANGE RATE ERROR
 C FOR THE THIRD SIMULATION RUN FOR EACH FILTER APPEARS
 C ON THE SAME PLOT. THE FIRST
 C PLOT USES A SCALE CHOSEN FROM THE FUNCTION'S MAX AND
 C MIN VALUES. THE SECOND USES BU4 AND BL4 AS THE
 C PLOT'S MAX AND MIN VALUES. IF PLRRR IS NOT
 C SPECIFIED NO RANGE RATE ERROR PLOTS ARE GENERATED.
 C BU1 AND BL1 = THE UPPER AND LOWER LIMITS ON THE SECOND GRAPH
 C OF THE RANGE DATA. IF NOT SPECIFIED THEY ARE
 C SET
 C BU1 = (MEDIUM ACTUAL RANGE) + SPAN1
 C BL1 = (MEDIUM ACTUAL RANGE) - SPAN1
 C WHERE SPAN1 IS A MAIN PROGRAM CONSTANT.
 C BU2 AND BL2 = THE UPPER AND LOWER LIMITS ON THE SECOND GRAPH
 C OF THE RANGE RATE DATA. IF NOT SPECIFIED THEY
 C ARE INTERNALLY SET
 C BU2 = (MEDIUM ACTUAL RANGE) + SPAN2
 C BL2 = (MEDIUM ACTUAL RANGE) - SPAN2
 C WHERE SPAN2 IS A MAIN PROGRAM CONSTANT.
 C BU3 AND BL3 = THE UPPER AND LOWER LIMITS ON THE SECOND PLOT
 C OF THE RANGE ERROR FUNCTION. IF NOT SPECIFIED,
 C A SECOND PLOT IS NOT GENERATED.
 C BU4 AND BL4 = THE UPPER AND LOWER LIMITS ON THE SECOND PLOT
 C OF THE RANGE RATE ERROR FUNCTION. IF NOT
 C SPECIFIED, A SECOND PLOT IS NOT GENERATED.
 C CALCOM = 1 CREATES CALCOMP PLOTS. A RANGE PLOT AND RANGE
 C RATE PLOT AFTER THE THIRD SIMULATION RUN FOR EACH
 C FILTER. THE RANGE ERROR AND RANGE RATE ERROR
 C PLOTS ARE GENERATED AFTER ALL FILTERS HAVE BEEN
 C RUN. HENCE, THESE PLOTS CONTAIN THAT DATA FROM
 C THE THIRD SIMULATION RUN FOR EACH FILTER. THIS
 C IS FOR COMPARISON PURPOSES.

```

C
C      *NOTE*  INTERNAL SPECIFICATION OF BU1,BL1,BU2 AND BL2 CAN
C              ONLY BE USED WHEN NO ACCELERATION IS ADDED. IF THE
C              VALUES OF SPAN1 AND SPAN2 ARE NOT ADEQUATE, THEY MAY
C              BE CHANGED BY ALTERING THEIR VALUE IN THE MAIN
C              PROGRAM.
C
C  NAMELIST XRAND
C      XRAND IS READ ONLY ONCE .
C      XM = THE MEAN OF THE NORMALLY DISTRIBUTED RANDOM VARIABLES
C      XDEV = THE STANDARD DEVIATION OF THE NORMALLY DISTRIBUTED
C             RANDOM VARIABLES
C
C  INTERNAL PARAMETERS
C      NN = 0 TO NDN, NUMBER OF FILTERS THAT HAVE BEEN SIMULATED.
C      L = 0 TO IDRI, NUMBER OF TIMES SUBROUTINE CALC IS CALLED
C           (ONE DATA POINT IS GENERATED FOR EACH L)
C
C  PROGRAM FLT
C  DOUBLE PRECISION
C      IMPLICIT REAL*8(D,K,R,V)
C      INTEGER PLR,PLRR,PLRRR,PLRRRR,RUN
C      DIMENSION TIMA(500)
C      DIMENSION X(50,5),DIFF(5,500),RERR(5,500),Y(2500)
C      DIMENSION RNG(5,500),RNCR(5,500),TIME(500)
C      DIMENSION SNG(5,500),SNCR(5,500),SIFF(5,500),SERR(5,500)
C      DIMENSION Z(6,6),XMI(6),SDI(6)
C  MATIX Z IS A DUMMY VARIABLE USED IN BECORI.
C      NAMELIST/LIST/DRE,DRDOT,DT,K1,K2,DR,DV,NDN,IDI,IDRI
C      NAMELIST/DIM/PLR,PLRR,PLRRR,PLRRRR,BU1,BL1,BU2,BL2,BU3,BL3,BU4,BL4
C      2,CALCOM
C      NAMELIST/XRAND/XM,XDEV
C      CALL PLOTS (1024,4096)

```

```

CALL PLTMSG ('*** DON FLEISCHER PLOTS READY ***',33)
CALL PLOT (0.0,-11.0,-3)
CALL PLOT (0.0,1.0,-3)
NN = 0
BU1 = 0.0
BL1 = 0.0
BU2 = 0.0
BL2 = 0.0
BU3 = 0.0
BL3 = 0.0
BU4 = 0.0
BL4 = 0.0
PLR = 0
PLRR = 0
PLRER = 0
PLRRER = 0
CALCOM = 0.0
CALL GGNRF1(DSEED)
READ(S,DIM)
READ(S,XRAND)
100 READ(S,LIST)
NN = NN+1
BETA = IDI*DT
110 RUN = 1
RE = DRE
ROOT = DRDOT
R = DR
V = DV
XXM = XM
XXDEV = XDEV
XM = 0.0
XDEV = 0.0
GOTO 140
120 RUN = 2

```

```

      CALL GGNRF2(DSEED)
      RE = 0.0
      RDOT = 0.0
      R = 0.0
      V = 0.0
      XM = XXM
      XDEV = XXDEV
      GOTO 140
130  RUN = 3
      CALL GGNRF2(DSEED)
      RE = DRE
      RDOT = DRDOT
      R = DR
      V = DV
140  CONTINUE
      WRITE(6,150)
150  FORMAT(1H0,2X,'LIST')
      WRITE(6,160) RE,RDOT,DT,K1,K2,R,V,NDN,IDI,IDRI
160  FORMAT(1H0,4X,'RE = ',G20.9/5X,'RDOT = ',G20.9/5X,'DT = ',G20.9/5X
2,'K1 = ',G20.9/5X,'K2 = ',G20.9/5X,'R = ',G20.9/5X,'V = ',G20.9/5X
3,'NDN = ',I11/5X,'IDI = ',I11/5X,'IDRI = ',I11///)
      WRITE(6,170)
170  FORMAT(1H0,2X,'XRAND')
      WRITE(6,180) XM,XDEV
180  FORMAT(1H0,4X,'XM = ',G20.9/5X,'XDEV = ',G20.9)
      IF(RUN .NE. 3) GOTO 300
C
C  CALCULATION OF BU1,BL1,BU2 AND BL2 IF NOT GIVEN AS A INPUT
C
      IF (BU1 .NE. 0.0 .OR. BL1 .NE. 0.0) GOTO 190
      SPAN1 = 1000.0
      BU1 = R + V*IDI*DT*IDRI/2 + SPAN1
      BL1 = BU1 - 2*SPAN1
190  IF (BU2 .NE. 0.0 .OR. BL2 .NE. 0.0) GOTO 200
      SPAN2 = 2.0

```

```

      BU2 = V + SPAN2
      BL2 = V - SPAN2
200  WRITE(6,210)
210  FORMAT(1H0,2X,'DIM')
      WRITE(6,220) PLR,PLRR,PLRER,PLRRER,BU1,BL1,BU2,BL2
220  FORMAT(1H0,4X,'PLR = ',I6/5X,'PLRR = ',I6/5X,'PLRER = ',I6/5X,'PLR
2RER = ',I6/5X,'BU1 = ',G20.9/5X,'BL1 = ',G20.9/5X,'BU2 = ',G20.9/5
3X,'BL2 = ',G20.9)
      IF (BU3 .EQ. 0.0 .AND. BL3 .EQ. 0.0) GO TO 240
      WRITE(6,230) BU3,BL3
230  FORMAT(1H0,4X,'BU3 = ',G20.9/5X,'BL3 = ',G20.9)
      GOTO 260
240  WRITE(6,250)
250  FORMAT(1H0,4X,'BU3 AND BL3 WERE NOT SPECIFIED IN DIM')
260  IF (BU4 .EQ. 0.0 .AND. BL4 .EQ. 0.0) GO TO 280
      WRITE(6,270) BU4,BL4
270  FORMAT(1H0,4X,'BU4 = ',G20.9/5X,'BL4 = ',G20.9///)
      GOTO 300
280  WRITE(6,290)
290  FORMAT(1H0,4X,'BU4 AND BL4 WERE NOT SPECIFIED IN DIM'///)
300  CONTINUE
      L = 0
      WRITE(6,310)
310  FORMAT(1H0,2X,'TIME',17X,'R',17X,'RE',15X,'RERR',15X,'V',15X,'RDOT
2',15X,'DIFF')
      RDI = 0.0
      RVI = 0.0
C
C  AN IF STATEMENT WILL BE INSERTED HERE TO CHECK TO SEE IF THE RANDOM
C  NUMBER GENERATOR IS TO BE USED ON THIS RUN.  IF IT IS, THE FOLLOWING
C  LOOP WILL BE SKIPPED.
C
320  IF (XDEV .NE. 0.0) GO TO 340
      DO 330 LAP = 1,IDI
          Y(LAP) = 0.0
330  CONTINUE

```

```

      GOTO 350
C
C THE CALL STATEMENT FOR THE RANDOM NUMBER GENERATOR WILL BE
C HERE. THIS SUBROUTINE WILL BE CALLED EACH TIME CALC IS CALLED.
C
340 CALL GGNRF (IDI,Y)
      DO 350 LAP = 1,IDI
        Y(LAP) = XM + XDEV*Y(LAP)
350 CONTINUE
      L = L+1
      TIME(L) = IDI*DT*L
      CALL CALC(R,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RH,Y,IDI)
C
C VARIABLES BEGINNING WITH R OR D ARE DOUBLE PRECISION. THEY ARE USED
C IN DATA TABLE OUTPUT. VARIABLES BEGINNING WITH S ARE SINGLE
C PRECISION. THEY ARE USED IN THE PLOT OUTPUT.
C
C RNG CONTAINS 3 FUNCTIONS TO BE PLOTTED ON THE RANGE GRAPH.
      RNG(1,L) = RE
      RNG(2,L) = R
      RNG(3,L) = RM
      SNG(1,L) = RE
      SNG(2,L) = R
      SNG(3,L) = RM
C
C RNGR CONTAINS 2 FUNCTIONS TO BE PLOTTED ON THE RANGE RATE GRAPH.
      RNGR(1,L) = RDOT
      RNGR(2,L) = V
      SNGR(1,L) = RDOT
      SNGR(2,L) = V
C
C RERR CONTAINS 1 FUNCTION TO BE PLOTTED ON THE RANGE ERROR GRAPH.
      RERR(NN,L) = RE - R
      SERR(NN,L) = RE - R
C
C DIFF CONTAINS 1 FUNCTION TO BE PLOTTED ON THE RANGE
C RATE ERROR GRAPH.
      DIFF(NN,L) = RDOT - V

```

```

      SIFF(NN,L) = RDOT - V
C
      WRITE(6,360) TIME(L),R,RE,RERR(NN,L),V,RDOT,DIFF(NN,L)
360  FORMAT(1H0,3X,G7.1,6(1X,G18.11))
      IF(L.LT.IDRI) GO TO 320
      RERRMX = 0.0
      DIFFMX = 0.0
      DO 370 LAP=1,50
      IF(ABS(X(LAP,4)).GT.RERRMX) RERRMX = ABS(X(LAP,4))
      IF(ABS(X(LAP,5)).GT.DIFFMX) DIFFMX = ABS(X(LAP,5))
370  CONTINUE
      WRITE(6,380) RERRMX,DIFFMX
380  FORMAT(1H0,4X,*MAXIMUM STEADY STATE RANGE ERROR = *,G20.9/5X,*MAXI
1MUM STEADY STATE RANGE RATE ERROR = *,G20.9)
      CALL BECORI(X,50,3,50,XMI,SDI,Z,IER)
      WRITE(6,390) XMI(1),SDI(1)
390  FORMAT(1H0,4X,*INPUT RANGE MEAN = *,G20.9/5X,*INPUT RANGE STND DEV
1 = *,G20.9)
      WRITE(6,400) XMI(2),SDI(2)
400  FORMAT(1H0,4X,*RANGE MEAN = *,G20.9/5X,*RANGE STND DEV = *,G20.9)
      WRITE(6,410) XMI(3),SDI(3)
410  FORMAT(1H0,4X,*RANGE RATE MEAN = *,G20.9/5X,*RANGE RATE STND DEV =
1 *,G20.9)
      IF(RUN .EQ. 1) GOTO 120
      IF(RUN .EQ. 2) GOTO 130
      IF(PLR .EQ. 0) GOTO 460
      CALL PLOTB (3,IDRI,BETA,SNQ,TIME,0.0,0.0,0)
      WRITE(6,420)
      WRITE(6,430)
      WRITE(6,440)
      WRITE(6,450)
420  FORMAT(1H0,26X,*****,*FOR RANGE DATA*,24X,*****)
430  FORMAT(1H0,26X,*****,*1 IS THE CALCULATED RANGE (RE)*,8X,*****)
440  FORMAT(1H0,26X,*****,*2 IS THE ACTUAL RANGE (R)*,13X,*****)
450  FORMAT(1H0,26X,*****,*3 IS THE NOISY RANGE (RM)*,14X,*****)

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

CALL PLOTB (3,IDRI,BETA,SNG,TIME,RU1,BL1,0)
WRITE(6,420)
WRITE(6,430)
WRITE(6,440)
WRITE(6,450)
460 IF(PLRR .EQ. 0) GOTO 500
CALL PLOTB (2,IDRI,BETA,SNGR,TIME,0.0,0.0,0)
WRITE(6,470)
WRITE(6,480)
WRITE(6,490)
470 FORMAT(1H0,26X,'*****','FOR RANGE RATE DATA',19X,'*****')
480 FORMAT(1H0,26X,'*****','1 IS THE CALCULATED RANGE RATE (RDOT)',1X,'*
2**')
490 FORMAT(1H0,26X,'*****','2 IS THE ACTUAL RANGE RATE (V)',5X,'*****')
CALL PLOTB (2,IDRI,BETA,SNGR,TIME,BU2,BL2,0)
WRITE(6,470)
WRITE(6,480)
WRITE(6,490)
500 CONTINUE
IF(CALCOM .EQ. 0.0) GOTO 510
CALL MCPILOT(SNG,TIME,IDRI,2,0,SW,DATA)
CALL MCPILOT(SNGR,TIME,IDRI,2,0,SW,DATA)
IDRII = IDRI - 15
DO 505 I=1,IDRII
IA = I + 15
TIMA(I) = TIME(IA)
SNG(1,I) = SNG(1,IA)
SNG(2,I) = SNG(2,IA)
SNGR(1,I) = SNGR(1,IA)
SNGR(2,I) = SNGR(2,IA)
505 CONTINUE
CALL MCPILOT(SNG,TIMA,IDRII,2,0,SW,DATA)
CALL MCPILOT(SNGR,TIMA,IDRII,2,0,SW,DATA)
510 IF(NN .LT. NDN) GOTO 100
IF (PLRR .EQ. 0) GO TO 530

```

```

      CALL PLOTB (NDN,IDRI,BETA,SERR,TIME,0.0,0.0,1)
      WRITE(6,520)
520  FORMAT(1H0,26X,'*****',*FOR RANGE ERROR (RERR)',16X,'*****')
      IF (BU3 .EQ. 0.0 .AND. BL3 .EQ. 0.0) GO TO 530
      CALL PLOTB (NDN,IDRI,BETA,SERR,TIME,BU3,BL3,1)
      WRITE(6,520)
530  IF (PLRRER .EQ. 0) GO TO 550
      CALL PLOTB (NDN,IDRI,BETA,SIFF,TIME,0.0,0.0,1)
      WRITE(6,540)
540  FORMAT(1H0,26X,'*****',*FOR RANGE RATE ERROR (DIFF)',11X,'*****')
      IF (BU4 .EQ. 0.0 .AND. BL4 .EQ. 0.0) GO TO 550
      CALL PLOTB (NDN,IDRI,BETA,SIFF,TIME,BU4,BL4,1)
      WRITE(6,540)
550  CONTINUE
      IF (CALCOM .EQ. 0.0) GOTO 570
C   THIS LOOP GENERATES A ZERO REFERENCE LINE FOR THE CALCOMP PLOT.
      NDN1 = NDN + 1
      DO 560 L=1,IDRI
      SERR(NDN1,L) = 0.0
560  SIFF(NDN1,L) = 0.0
      CALL MCPLLOT(SERR,TIME,IDRI,NDN1,0,SW,DATA)
      CALL MCPLLOT(SIFF,TIME,IDRI,NDN1,0,SW,DATA)
      DO 567 J=1,NDN1
      DO 565 I=1,IDRII
      IA = I + 15
      SIFF(J,I) = SIFF(J,IA)
      SERR(J,I) = SERR(J,IA)
565  CONTINUE
567  CONTINUE
      CALL MCPLLOT(SERR,TIMA,IDRII,NDN1,0,SW,DATA)
      CALL MCPLLOT(SIFF,TIMA,IDRII,NDN1,0,SW,DATA)
570  CONTINUE
      CALL PLOT (0.0,0.0,999)
      STOP
      END

```

```

      SUBROUTINE PLOTB (NP,MAX,BETA,FNCT,TIME,BU,BL,NZERO)
C
C   PLOTB USES THE PRINTER AS A PLOTTER.
C       NP = NUMBER OF FUNCTIONS PLOTTED ON THE GRAPH, MAXIMUM OF 10
C       MAX = NUMBER OF DATA POINTS IN EACH FUNCTION
C       BETA = SCALE OF THE Y AXIS (ABSCISSA) IN UNITS OF TIME
C       BU AND BL = THE UPPER AND LOWER LIMITS OF THE X AXIS (ORDINATE).
C               IF BU=BL=0.0 THE THE MAXIMUM AND MINIMUM VALUES OF
C               THE FUNCTION ARE USED AS THE LIMITS.
C       NZERO = 1 WILL PUT A REFERENCE LINE AT X=0. IF A REFERENCE
C               LINE IS NOT DESIRED SET NZERO = 0.
C
      INTEGER BLANK, DASH, DOT, DOLAR, AM(10)
      DIMENSION KK(10), LINE(101)
      DIMENSION FNCT(5,500), TIME(500)
      DATA NPR / 6 /
      DATA BLANK,DASH,DOT,AM(1),AM(2),AM(3),AM(4),AM(5),AM(6),AM(7),
1 AM(8),AM(9),AM(10) / ' ', '- ', '.', '1', '2', '3', '4', '5',
2 '6', '7', '8', '9', '0' /
      DATA DOLAR / '$' /
      IF (BU .EQ. 0.0 .AND. BL .EQ. 0.0) GO TO 210
      BIG = BU
      SMALL = BL
      GO TO 507
210  BIG=FNCT(1,1)
      SMALL=BIG
      DO 212 M = 1, NP
      DO 212 L = 2, MAX
      IF (FNCT(M,L)-BIG) 260,260,213
213  BIG=FNCT(M,L)
260  IF (FNCT(M,L)-SMALL) 214,212,212
214  SMALL=FNCT(M,L)
212  CONTINUE
507  PEAK=BIG-SMALL
      APEAK = ABS(PEAK)
      AMID=(BIG+SMALL)/2.0

```

```

        SCALE=APEAK/80.0
        WRITE (NPR,112) AMID, BIG, SMALL, SCALE
112  FORMAT (1H1,4X,'1-CURVE ORDINATE',5X,'MID-VALUE=',
        2E10.3,21X,'MAX-VALUE=',E10.3,8X,'MIN-VALUE=',
        3E10.3,/,27X,'SCALE=',E10.3,2X,'PER LINE',/)
        WRITE (NPR,113) BETA
113  FORMAT (1H ,4X,'ABSCISSA',13X,'SCALE=',
        2E10.3,2X,'PER LINE',62X,'W-RPS',/)
        IF ( APEAK .NE. 0.0 ) GOTO 35
        WRITE(NPR,30)
30   FORMAT(1H0,4X,'*****
1*****'/15X,'NO PLOT WILL BE PRODUCED HERE BECAUSE'/15X,
2'FUNCTION MAX VALUE = FUNCTION MIN VALUE.'/15X,'THE FUNCTION IS CO
3NSTANT. TO RECEIVE A '/15X,'PLOT OF THIS FUNCTION, PREDETERMINED'
4'/15X,'UPPER AND LOWER LIMITS BU AND BL MUST'/15X,'BE SPECIFIED.'/5
5X,'*****
6**')
        GOTO 41
35   CONTINUE
        DO 38 I = 1,101
38   LINE(I)=DASH
        LINE(11) = DOT
        LINE(51) = DOT
        LINE(91) = DOT
        DO 41 J = 1,MAX
        DO 40 K=1,NP
        KK(K)=80.0*(FNCT(K,J)-AMID)/APEAK + 51.0
        KKK = KK(K)
        IF(KKK) 51,51,52
51   LINE(1) = DOLAR
        GO TO 54
52   IF(KKK-101) 50,50,53
53   LINE(101) = DOLAR
        GO TO 54
50   LINE(KKK) = AM(K)

```

```

      GOTO 12
11  CENTER=0.0
      LETTER=-1
12  CONTINUE
      DO 4 KK=1,J
          KKK=KK+J
          YY(KK)=FUNCT(KK,1)
          YY(KKK)=FUNCT(KK,1)
      DO 5 L=1,JJ
          Y(L)=FUNCT(KK,L)
          IF (YY(KK).GT.Y(L)) YY(KK)=Y(L)
          IF (YY(KKK).LT.Y(L)) YY(KKK)=Y(L)
5  CONTINUE
4  CONTINUE
      JJJ=2*J
      CALL SCALE (YY,SH,JJJ,K)
      CALL SCALE (TIME, SW, JJ, K)

```

C
C
C
C
C

X-SCALE OVERRIDE CAN BE DONE HERE BY SPECIFYING TIME(JJ+1) AS
THE VALUE OF THE FIRST TIC MARK ON THE X-AXIS AND TIME(JJ+2) AS
THE CHANGE IN X VALUE BETWEEN TWO TIC MARKS.

```

      CALL AXIS (0.0,CENTER,' ',LETTER,SW,0.0,TIME(JJ+1),TIME(JJ+2))
      CALL AXIS (0.0,0.0,' ',0,SH,90.0,YY(JJJ+1),YY(JJJ+2))
      DO 8 NP=1,J
          DO 9 LL=1,JPTS
              Y(LL)=FUNCT(NP, LL)
9  CONTINUE
          Y(JJ+1)=YY(JJJ+1)
          Y(JJ+2)=YY(JJJ+2)
2  CALL LINE (TIME, Y, JPTS, 1, 0, 0)
8  CONTINUE
      CALL PLOT (SW+2.0, 0.0, -3)
      RETURN
      END

```

OF POOR QUALITY

```

54  CONTINUE
40  CONTINUE
    IF ( NZERO .NE. 1 ) GOTO 44
    KKK = 80.0 * (-AMID)/APEAK + 51.0
    LINE(KKK) = AM(10)
44  CONTINUE
    WRITE (NPR, 39) J, LINE, TIME(J)
39  FORMAT (1H,4X,I3,1X,101A1,1X,E10.3)
    DO 43 L = 1,101
43  LINE(L) = BLANK
    LINE(1)=DOT
41  CONTINUE
    WRITE (NPR, 442 )
442 FORMAT ( 1H , 26X,'***' , ' PLOTTING ROUTINE COMPLETED', 12X, '***PLOT*
1*   )
    RETURN
    END

```

```

      SUBROUTINE MCPLLOT (FUNCT,TIME,JJ,J,NOMAG1,SW,DATA)
C      THIS SUBROUTINE PLOTS GRAPHS ON CALCOMP PLOTTER
C      VARIABLES ARE DEFINED AS FOLLOWS:
C          FUNCT= ARRAY OF Y VALUES
C          TIME= ARRAY OF X VALUES
C          J= NUMBER OF PLOTS PER GRAPH
C          JJ= NUMBER OF POINTS PER PLOT
C          SW= X AXIS LENGTH IN INCHES
C          SH= Y AXIS LENGTH IN INCHES
      DIMENSION FUNCT (5,500), TIME (500), Y(500), DATA (1024), YY(500)
      SW=10.1
      JPTS=JJ
      SH=7.5
      K=1
      IF (NOMAG1) 11, 11, 10
10  CENTER=SH
      LETTER=0

```

```

      SUBROUTINE CALC(R,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RM,Y,IDRI)
C   FILTER 1   BLOCK DIAGRAM 1
      IMPLICIT REAL*8(D,K,R,V)
      DIMENSION X(50,5),Y(2500)
      IDRIST = IDRI - 50/IDI
      IDIST = IDI-50+(50/IDI)*IDI
      A=0.0
      DO 50 LAP=1,IDI
C   ACCELERATION CAN BE ADDED FOR A PERIOD OF 10*IDI*DT HERE, L=41 TO 50
      IF((L.GT.40).AND.(L.LE.50)) A=0.0
      V=V+A*DT
C   NEW RANGE, R (TRUE RANGE)
      R=R+V*DT
C   ADD RANDOM NO., RM (NOISY RANGE)
      RM = R + Y(LAP)
C   NOISY RANGE MINUS PREVIOUS ESTIMATED RANGE
      RV=RM-RE
C   ESTIMATED RANGE RATE, RDOT
      RDOT=RDOT+K2*RV*DT
C   ESTIMATED RANGE, RE
      RDI=RDI+RDOT*DT
      RVI=RVI+K1*RV*DT
      RE=RDI+RVI
      IF(L .LT. IDRIST) GOTO 50
      IF(L .EQ. IDRIST .AND. LAP .LE. IDIST) GOTO 50
      LAPX = (L-1)*IDI + LAP - ((IDRIST-1)*IDI + IDIST)
      X(LAPX,1) = RM
      X(LAPX,2) = RE
      X(LAPX,3) = RDOT
      X(LAPX,4) = RE-R
      X(LAPX,5) = RDOT-V
50  CONTINUE
      RETURN
      END

```

```

      SUBROUTINE CALC(R,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RM,Y,IDRI)
C  ALPHA-BETA FILTER
C  BETA = ALPHA**2/(2-ALPHA)
      IMPLICIT REAL*8(D,K,R,V)
      DIMENSION X(50,5),Y(2500)
      IDRIST = IDRI - 50/IDI
      IDIST = IDI-50+(50/IDI)*IDI
      A=0.0
      DO 50 LAP=1,IDI
C  ACCELERATION CAN BE ADDED FOR A PERIOD OF 10*IDI*DT HERE, L=41 TO 50
      IF((L.GT.40).AND.(L.LE.50)) A=0.0
      V=V+A*DT
C  NEW RANGE, R (TRUE RANGE)
      R=R+V*DT
C  ADD RANDOM NO., RM (NOISY RANGE)
      RM = R + Y(LAP)
      RE = RDI+K1*(RM-RDI)
      RDOT = RDOT+K2/DT*(RM-RDI)
      RDI = RE+DT*RDOT
      IF(L.LT.IDRIST) GOTO 50
      IF(L.EQ.IDRIST .AND. LAP.LE.IDIST) GOTO 50
      LAPX = (L-1)*IDI + LAP - ((IDRIST-1)*IDI + IDIST)
      X(LAPX,1) = RM
      X(LAPX,2) = RE
      X(LAPX,3) = RDOT
      X(LAPX,4) = RE-R
      X(LAPX,5) = RDOT-V
50  CONTINUE
      RETURN
      END

```

```

      SUBROUTINE CALC(P,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RN,Y,IDI)
C   FILTER 3          SMOOTHER
      IMPLICIT REAL*8(D,K,R,V)
      DIMENSION X(50,5),REG(18),Y(2500)
      IF (L .GT. 1) GOTO 10
      DATA REG/18*0.000/
10   CONTINUE
      IDIST = IDI - 50/IDI
      IDIST = IDI-50+(50/IDI)*IDI
      A=0.0
      DO 50 LAP=1,IDI
C   ACCELERATION CAN BE ADDED FOR A PERIOD OF 10*IDI*DT HERE, L=41 TO 50
      IF((L.GT.40).AND.(L.LE.50))A=0.0
      V=V+A*DT
C   NEW RANGE, R (TRUE RANGE)
      R=R+V*DT
C   ADD RANDOM NO., RM (NOISY RANGE)
      RN = R + Y(LAP)
      REG(18) = REG(17)
      REG(17) = REG(16)
      REG(16) = REG(15)
      REG(15) = REG(14)
      REG(14) = REG(13)
      REG(13) = REG(12)
      REG(12) = REG(11)
      REG(11) = REG(10)
      REG(10) = REG( 9)
      REG( 9) = REG( 8)
      REG( 8) = REG( 7)
      REG( 7) = REG( 6)
      REG( 6) = REG( 5)
      REG( 5) = REG( 4)
      REG( 4) = REG( 3)
      REG( 3) = REG( 2)
      REG( 2) = REG( 1)

```

```

      REG( 1) = RM
      DSUM = 0.0
      DO 30 I=1,6
      DSUM = DSUM + REG(I) - REG(I+12)
30    CONTINUE
C    ESTIMATED RANGE RATE, RDOT
      RDOTT = DSUM * 4.5/(18.0 * DT)
      RDOT = RDOTT/18.0
      DSUM = 0.0
      DO 40 I=1,16
      DSUM = DSUM + REG(I)
40    CONTINUE
C    ESTIMATED RANGE, RE
      RE = DSUM/16.0 + RDOT *DT *7.5
      IF(L .EQ. 1) RE = RM
      IF(L .EQ. 1) RDOT = V
      IF(L .LT. IDRIST) GOTO 50
      IF((L .EQ. IDRIST) .AND. (LAP .LE. IDIST)) GOTO 50
      LAPX = (L-1)*IDI + LAP - ((IDRIST-1)*IDI + IDIST)
      X(LAPX,1) = RM
      X(LAPX,2) = RE
      X(LAPX,3) = RDOT
      X(LAPX,4) = RE-R
      X(LAPX,5) = RDOT-V
50    CONTINUE
      RETURN
      END

```

```

      SUBROUTINE CALC(R,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RM,Y,IDRI)
C   FILTER 4   DESIGN 1
      IMPLICIT REAL*8(D,K,R,V)
      DIMENSION X(50,5),REG(4),Y(2500)
      IF (L .GT. 1) GOTO 10
      DATA REG/4*0.0D0/
10   CONTINUE
      IDRIST = IDRI - 50/IDI
      IDIST = IDI-50+(50/IDI)*IDI
      A=0.0
      DO 50 LAP=1,IDI
C   ACCELERATION CAN BE ADDED FOR A PERIOD OF 10*IDI*DT HERE, L=41 TO 50
      IF((L.GT.40).AND.(L.LE.50)) A=0.0
      V=V+A*DT
C   NEW RANGE, R (TRUE RANGE)
      R=R+V*DT
      RM = R + Y(LAP)
      REG( 4) = REG( 3)
      REG( 3) = REG( 2)
      REG( 2) = REG( 1)
      REG( 1) = RM-RE
      DELTA = (REG(1)+REG(2)+REG(3)+REG(4))/4.0
      RE = DELTA + RE
      RDOT = DELTA/DT
      IF(L .LT. IDRIST) GOTO 50
      IF(L .EQ. IDRIST .AND. LAP .LE. IDIST) GOTO 50
      LAPX = (L-1)*IDI + LAP - ((IDRIST-1)*IDI + IDIST)
      X(LAPX,1) = RM
      X(LAPX,2) = RE
      X(LAPX,3) = RDOT
      X(LAPX,4) = RE-R
      X(LAPX,5) = RDOT-V
50   CONTINUE
      RETURN
      END

```

```

      SUBROUTINE CALC(R,V,DT,X,IDI,RE,RDOT,K1,K2,RDI,RVI,L,RM,Y,IDRI)
C   FILTER 5
C   FOR THIS FILTER, CALCULATIONS ARE MADE EVERY 16 SAMPLES.
C   THIS MEANS THAT FOR EVERY RUN THROUGH CALC IDI/16 VALUES ARE
C   DETERMINED RATHER THAN IDI VALUES. TO ALLOW FOR THIS, IDI IS CHANGED
C   TO IDI/16. THIS IS NECESSARY BECAUSE ALL THE VALUES CALCULATED
C   IN THE LAST RUN THROUGH CALC ARE RETURNED TO BE USED IN VARIANCE
C   CALCULATIONS.
C
C   *NOTE* - THE IMPLICATION OF THIS CHANGE IS THAT IDI MUST BE AN
C   INTEGRAL MULTIPLE OF 16. ALSO, IDRI*IDI/16 MUST BE GREATER THAN 50.
C   THESE LIMITATIONS APPLY FOR FILTER 5 ONLY.
C
      IMPLICIT REAL*8(D,K,R,V)
      DIMENSION X(50,5),REG(16),Y(2500)
      IF (L .GT. 1) GOTO 10
      DATA REG/16*0.0D0/
10   CONTINUE
      IDRI = IDRI - (50/(IDI/16))
      IDIST = IDI - ((50 - (50/(IDI/16)))*(IDI/16))*16)
      A=0.0
      DO 50 LAP=1,IDI
C   ACCELERATION CAN BE ADDED FOR A PERIOD OF 10*IDI*DT HERE, L=41 TO 50
      IF((L.GT.40).AND.(L.LE.50))A=0.0
      V=V+A*DT
C   NEW RANGE, R (TRUE RANGE)
      R=R+V*DT
C   ADD RANDOM NO., RM (NOISY RANGE)
      RM = R + Y(LAP)
      REG(16) = REG(15)
      REG(15) = REG(14)
      REG(14) = REG(13)
      REG(13) = REG(12)
      REG(12) = REG(11)
      REG(11) = REG(10)

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

REG(10) = REG( 9)
REG( 9) = REG( 8)
REG( 8) = REG( 7)
REG( 7) = REG( 6)
REG( 6) = REG( 5)
REG( 5) = REG( 4)
REG( 4) = REG( 3)
REG( 3) = REG( 2)
REG( 2) = REG( 1)
REG( 1) = RM-RDI
IF((LAP-(LAP/16)*16) .NE. 0) GOTO 50
DSUM = 0.0
DO 20 I=1,16
DSUM = DSUM + REG(I)
20 CONTINUE
RDOT = DSUM/16.0/DT/16.0
RDI = RDI + DSUM/16.0
RE = RDI + DSUM/16.0*7.5/16.0
IF(L .EQ. 1) RE = RM
IF(L .EQ. 1) RDOT = V
IF(L .LT. IDRIST) GOTO 50
IF(L .EQ. IDRIST .AND. LAP .LE. IDIST) GOTO 50
LAPX = ((L-1)*IDI + LAP - ((IDRIST-1)*IDI + IDIST))/16
X(LAPX,1) = RM
X(LAPX,2) = RE
X(LAPX,3) = RDOT
X(LAPX,4) = RE-R
X(LAPX,5) = RDOT-V
50 CONTINUE
RETURN
END

```

APPENDIX B

The following pages contain an assembly language listing of program FILTER 3.

FILTER 3

DECEMBER 19, 1978

♦
♦
♦
♦
♦ FILTER 3
♦

0000	ORG	\$0000
0000	XTMP	RMB 2
0002	SGN	RMB 1
0003	CNT	RMB 1
0004	MLTCD1	RMB 1
0005	MLTCD2	RMB 1
0006	MLTPR1	RMB 1
0007	MLTPR2	RMB 1
0008	MLTPR3	RMB 1
0009	PRDD1	RMB 1
000A	PRDD2	RMB 1
000B	PRDD3	RMB 1
000C	PRDD4	RMB 1
000D	PRDD5	RMB 1
000E	A1	RMB 1
000F	A2	RMB 1
0010	A3	RMB 1
0011	B1	RMB 1
0012	B2	RMB 1
0013	B3	RMB 1
0014	SUM1	RMB 1
0015	SUM2	RMB 1
0016	SUM3	RMB 1
001E	ORG	\$001E
001E	RTMP	RMB 2
0020	REGS	RMB 36
0044	ORG	\$0044
0044	RCNT	RMB 1
0045	SKPFLG	RMB 1
0046	RORCT1	RMB 1
0047	RORCT2	RMB 1
0048	RORCT3	RMB 1
0049	RORCT4	RMB 1
004A	RDOT1	RMB 1
004B	RDOT2	RMB 1
004C	RE1	RMB 1
004D	RE2	RMB 1
004E	TSUM1	RMB 1
004F	TSUM2	RMB 1
0050	TSUM3	RMB 1
0051	SAMP	RMB 2
0053	RAMPTR	RMB 2
0055	DATPTR	RMB 2

FILTER 3

DECEMBER 19, 1978

0057	REPTR	RMB	2
0059	RDOTPT	RMB	2
005B	SAMPND	RMB	2
005D	DIGIT	RMB	1
005E	TYPE	RMB	1
005F	ROUND	RMB	1
0060	NFLAG	RMB	1
0061	C1	RMB	1
0062	C2	RMB	1
0063	D1	RMB	1
0064	D2	RMB	1
0065	DIFF1	RMB	1
0066	DIFF2	RMB	1

♦
♦

0200			ORG	\$0200		
0200	C6	01	FLT	LDA B	#01	
0202	D7	5F		STA B	ROUND	
0204	CE	00 01		LDX	#0001	
0207	DF	51		STX	SAMP	
0209	CE	30 00		LDX	#03000	
020C	DF	57		STX	REPTR	
020E	CE	38 00		LDX	#03800	
0211	DF	59		STX	RDOTPT	
0213	CE	04 3C		LDX	#INIT1	
0216	BD	E1 4B		JSR	\$E14B	MINIBUG PDATA1
0219	BD	E1 33		JSR	\$E133	MINIBUG INCH
021C	97	5E		STA A	TYPE	
021E	44			LSR A		
021F	24	05		BCC	INPUT1	
0221	CE	20 00		LDX	#02000	
0224	20	0B		BRA	INPUT3	
0226	44		INPUT1	LSR A		
0227	24	05		BCC	INPUT2	
0229	CE	28 00		LDX	#02800	
022C	20	03		BRA	INPUT3	
022E	CE	18 00	INPUT2	LDX	#01800	
0231	DF	55	INPUT3	STX	DATPTR	
0233	CE	04 9E		LDX	#INIT2	
0236	BD	E1 4B		JSR	\$E14B	MINIBUG PDATA1
0239	BD	E0 F8		JSR	\$E0F8	MINIBUG BADDR
023C	DF	5B		STX	SAMPND	
023E	CE	00 20		LDX	#REGS	
0241	6F	00	FLTA	CLR	0,X	
0243	08			INX		
0244	8C	00 44		CPX	#RCNT	
0247	26	F8		BNE	FLTA	
0249	DE	55	FLT1	LDX	DATPTR	
024B	EE	00		LDX	0,X	

FILTER 3

DECEMBER 19, 1978

0240	BD	04	20		JSR	SHFTR
0250	7F	00	14		CLR	SUM1
0253	7F	00	15		CLR	SUM2
0256	7F	00	16		CLR	SUM3
0259	CE	00	20		LDX	#REG3
025C	C6	06			LDA B	#B06
025E	D7	44			STA B	RCNT
0260	7F	00	45		CLR	SKPFLG
0263	DF	00		FLT2	STX	XTEMP
0265	CE	00	0E		LDX	#A1
0268	A6	06		FLT3	LDA A	6,X
026A	A7	00			STA A	0,X
026C	08				INX	
026D	8C	00	11		CPX	#B1
0270	26	F6			BNE	FLT3
0272	DE	00			LDX	XTEMP
0274	8C	00	2C		CPX	#B002C
0277	26	03			BNE	FLT4
0279	7A	00	45		DEC	SKPFLG
027C	8C	00	38	FLT4	CPX	#B0038
027F	26	03			BNE	FLT5
0281	7F	00	45		CLR	SKPFLG
0284	D6	45		FLT5	LDA B	SKPFLG
0286	26	18			BNE	FLT7
0288	7F	00	11		CLR	B1
028B	A6	00			LDA A	0,X
028D	E6	01			LDA B	1,X
028F	97	12			STA A	B2
0291	D7	13			STA B	B3
0293	7A	00	44		DEC	RCNT
0296	2A	05			BPL	FLT6
0298	BD	04	19		JSR	SUB
029B	20	03			BRA	FLT7
029D	BD	04	06	FLT6	JSR	ADD
02A0	08			FLT7	INX	
02A1	08				INX	
02A2	8C	00	44		CPX	#RCNT
02A5	26	8C			BNE	FLT2
02A7	96	14			LDA A	SUM1
02A9	97	4E			STA A	TSUM1
02AB	97	06			STA A	MLTPR1
02AD	DE	15			LDX	SUM2
02AF	DF	4F			STX	TSUM2
02B1	DF	07			STX	MLTPR2
02B3	CE	1A	AB		LDX	#B1AAB
02B6	DF	04			STX	MLTCD1
02B8	7F	00	5F		CLR	ROUND
02BB	BD	03	64		JSR	MULT
02BE	DE	0A			LDX	PROD2

FILTER 3

DECEMBER 19, 1978

02C0 DF 46		STX	RORCT1
02C2 DE 0C		LDX	PRDD4
02C4 DF 48		STX	RORCT3
02C6 D6 4E		LDA B	TSUM1
02C8 D7 06		STA B	MLTPR1
02CA DE 4F		LDX	TSUM2
02CC DF 07		STX	MLTPR2
02CE CE 16 39		LDX	#81639
02D1 DF 04		STX	MLTCD1
02D3 7C 00 5F		INC	ROUND
02D5 BD 03 64		JSR	MULT
02D9 DE 0A		LDX	PRDD2
02DB DF 4A		STX	RDOT1
02DD 7F 00 14		CLR	SUM1
02E0 7F 00 15		CLR	SUM2
02E3 7F 00 16		CLR	SUM3
02E6 CE 00 20		LDX	#REGS
02E9 DF 00	FLT10	STX	XTEMP
02EB CE 00 0E		LDX	#A1
02EE A6 06	FLT12	LDA A	6.X
02F0 A7 00		STA A	0.X
02F2 08		INX	
02F3 8C 00 11		CPX	#B1
02F6 26 F6		BNE	FLT12
02F8 DE 00		LDX	XTEMP
02FA 7F 00 11		CLR	B1
02FD EE 00		LDX	0.X
02FF DF 12		STX	B2
0301 BD 04 06		JSR	ADD
0304 DE 00		LDX	XTEMP
0306 08		INX	
0307 08		INX	
0308 8C 00 40		CPX	#80040
030B 26 DC		BNE	FLT10
030D D6 14		LDA B	SUM1
030F D7 06		STA B	MLTPR1
0311 DE 15		LDX	SUM2
0313 DF 07		STX	MLTPR2
0315 CE 1C 00		LDX	#81000
0318 DF 04		STX	MLTCD1
031A 7F 00 5F		CLR	ROUND
031D BD 03 64		JSR	MULT
0320 DE 46		LDX	RORCT1
0322 08		INX	
0323 D6 49		LDA B	RORCT4
0325 96 48		LDA A	RORCT3
0327 DB 0D		ADD B	PRDD5
0329 99 0C		ADC A	PRDD4
032B 2A 02		BPL	FLT14

FILTER 3

DECEMBER 19, 1978

```

032D DF 46          STX      RCRCT1
032F D6 47          LDA B    RCRCT2
0331 96 46          LDA A    RCRCT1
0333 D9 0B          ADC B    PROD3
0335 99 0A          ADC A    PROD2
0337 DE 57          LDX      REPTR
0339 A7 00          STA A     0,X
033B E7 01          STA B     1,X
033D 08             INX
033E 08             INX
033F DF 57          STX      REPTR
0341 DE 59          LDX      RDOTPT
0343 96 4A          LDA A     RDOT1
0345 D6 4B          LDA B     RDOT2
0347 A7 00          STA A     0,X
0349 E7 01          STA B     1,X
034B 08             INX
034C 08             INX
034D DF 59          STX      RDOTPT
034F DE 55          LDX      DATPTR
0351 08             INX
0352 08             INX
0353 DF 55          STX      DATPTR
0355 DE 51          LDX      SAMP
0357 9C 5B          CPX      SAMPND
0359 27 06          BEQ      END
035B 08             INX
035C DF 51          STX      SAMP
035E 7E 02 49       JMP      FLT1
0361 7E 04 E4       END      JMP      PRINTR
♦
♦ 3 BYTE X 2 BYTE 2'S COMPLEMENT MULTIPLY
♦
0364 DF 00          MULT    STX      XTEMP
0366 CE 00 02       LDX      #35H
0369 6F 00          CLR      0,X
036B 6C 00          INC      0,X
036D A6 02          LDA A     2,X
036F 2A 0B          BPL      MULT1
0371 E6 03          LDA B     3,X
0373 BD 03 FE       JSR      NEGATE
0376 A7 02          STA A     2,X
0378 E7 03          STA B     3,X
037A 6A 00          DEC      0,X
037C A6 04          MULT1   LDA A     4,X
037E 2A 14          BPL      MULT2
0380 A6 05          LDA A     5,X
0382 E6 06          LDA B     6,X
0384 BD 03 FE       JSR      NEGATE

```

FILTER 3

DECEMBER 19, 1978

0387	A7	05		STA A	5,X
0389	E7	06		STA B	6,X
038B	A6	04		LDA A	4,X
038D	BD	03	FF	JSR	NEG4
0390	A7	04		STA A	4,X
0392	6A	00		DEC	0,X
0394	86	E7	MULT2	LDA A	88E7
0396	A7	01		STA A	1,X
0398	4F			CLR A	
0399	5F			CLR B	
039A	6C	01	MULT3	INC	1,X
039C	27	16		BEQ	MULT4
039E	64	04		LSR	4,X
03A0	66	05		ROR	5,X
03A2	66	06		ROR	6,X
03A4	84	04		BCC	MULT31
03A6	EB	03		ADD B	3,X
03A8	A9	02		ADC A	2,X
03AA	44		MULT31	LSR A	
03AB	56			ROR B	
03AC	66	09		ROR	9,X
03AE	66	0A		ROR	10,X
03B0	66	0B		ROR	11,X
03B2	20	E6		BRA	MULT3
03B4	A7	07	MULT4	STA A	7,X
03B6	E7	08		STA B	8,X
03B8	7D	00	SF ROUND1	TST	ROUND
03BB	27	18		BEQ	MULT5
03BD	7D	00	0C	TST	PROD4
03C0	2A	0D		BFL	ROUND2
03C2	7C	00	0B	INC	PROD3
03C5	26	08		BNE	ROUND2
03C7	7C	00	0A	INC	PROD2
03CA	26	03		BNE	ROUND2
03CC	7C	00	09	INC	PROD1
03CF	7F	00	0C ROUND2	CLR	PROD4
03D2	7F	00	0D	CLR	PROD5
03D5	66	00	MULT5	ROR	0,X
03D7	25	1D		BCC	MULTND
03D9	A6	0A		LDA A	10,X
03DB	E6	0B		LDA B	11,X
03DD	BD	03	FE	JSR	NEGATE
03E0	A7	0A		STA A	10,X
03E2	E7	0B		STA B	11,X
03E4	A6	08		LDA A	8,X
03E6	E6	09		LDA B	9,X
03E8	BD	03	F9	JSR	NEG3
03EA	A7	08		STA A	8,X
03ED	E7	09		STA B	9,X

FILTER 3

DECEMBER 19, 1978

03EF	A6	07		LDA	A	7,X
03F1	BD	03	FF	JSR		NEG4
03F4	A7	07		STA	A	7,X
03F6	DE	00		MULTHD	LDX	XTEMP
03F8	39			RTS		

♦
♦ 2'S COMPLEMENT ROUTINE
♦

03F9	24	03		NEG3	BCC	NEGATE
03FB	53			COM	B	
03FC	20	06		BRN		NEG1
03FE	50			NEGATE	NEG	B
03FF	25	03		NEG4	BCS	NEG1
0401	40				NEG	A
0402	20	01			BRN	NEG2
0404	43			NEG1	COM	A
0405	39			NEG2	RTS	

♦
♦ 3 BYTE ADDITION ROUTINE
♦

0406	D6	10		ADD	LDA	B	A3
0408	96	0F			LDA	A	A2
040A	DB	13			ADD	B	B3
040C	99	12			ADC	A	B2
040E	D7	16			STA	B	SUM3
0410	97	15			STA	A	SUM2
0412	96	0E			LDA	A	A1
0414	99	11			ADC	A	B1
0416	97	14			STA	A	SUM1
0418	39				RTS		

♦
♦ 3 BYTE SUBTRACTION ROUTINE
♦

0419	D6	10		SUB	LDA	B	A3
041B	96	0F			LDA	A	A2
041D	D0	13			SUB	B	B3
041F	92	12			SBC	A	B2
0421	D7	16			STA	B	SUM3
0423	97	15			STA	A	SUM2
0425	96	0E			LDA	A	A1
0427	92	11			SBC	A	B1
0429	97	14			STA	A	SUM1
042B	39				RTS		

♦
♦ ROUTINE TO SHIFT DATA IN REGS
♦

042C	DF	1E		SHFTR	STX	RTEMP	
042E	CE	00	41		LDM	00041	
0431	E6	00		SHFTRL	LDA	B	0,X

FILTER 3

DECEMBER 19, 1978

0433 E7 02	STA B	2,X
0435 09	DEX	
0436 8C 00 1D	CPX	*\$001D
0439 26 F6	BNE	\$HFTR1
043B 39	RTS	
♦		
043C 0D	INIT1	FCB \$D,\$A,\$A,\$A,\$A,\$A,\$A
043D 0A 0A		
043F 0A 0A		
0441 0A 0A		
0443 46	FCC	/FILTER 3 MICROPROCESSOR/
0444 49 4C		
0446 54 45		
0448 52 20		
044A 33 20		
044C 4D 49		
044E 43 52		
0450 4F 50		
0452 52 4F		
0454 43 45		
0456 53 53		
0458 4F 52		
045A 20	FCC	/ SIMULATION RUN/
045B 53 49		
045D 4D 55		
045F 4C 41		
0461 54 49		
0463 4F 4E		
0465 20 52		
0467 55 4E		
0469 0D	FCB	\$D,\$A,\$A
046A 0A 0A		
046C 49	FCC	/INPUT CHOICE?/
046D 4E 50		
046F 55 54		
0471 20 43		
0473 48 4F		
0475 49 43		
0477 45 3F		
0479 0D	FCB	\$D,\$A
047A 0A		
047B 28	FCC	/(SIGNAL=0,NOISE=1,/
047C 53 49		
047E 47 4E		
0480 41 4C		
0482 3D 30		
0484 2C 4E		
0486 4F 49		
0488 53 45		

FILTER 3

DECEMBER 19, 1978

048A 3D 31
 048C 2C
 048D 53
 048E 49 47
 0490 4E 41
 0492 4C 2B
 0494 4E 4F
 0496 49 53
 0498 45 3D
 049A 32 29
 049C 20
 049D 04
 049E 0D
 049F 0A 0A
 04A1 4E
 04A2 55 4D
 04A4 42 45
 04A6 52 20
 04A8 4F 46
 04AA 20
 04AB 53
 04AC 41 4D
 04AE 50 4C
 04B0 45 53
 04B2 20 54
 04B4 4F 20
 04B6 42 45
 04B8 20 52
 04BA 55 4E
 04BC 3F
 04BD 0D
 04BE 0A
 04BF 28
 04C0 45 4E
 04C2 54 45
 04C4 52 20
 04C6 41 20
 04C8 34 20
 04CA 44 49
 04CC 47 49
 04CE 54 20
 04D0 48
 04D1 45 53
 04D3 49 44
 04D5 45 43
 04D7 49 40
 04D9 41 4C
 04DB 20 56
 04DD 41 4C

FCC /SIGNAL+NOISE=2) /

INIT2 FCB 04
 FCB 3D, 3A, 3A

FCC /NUMBER OF /

FCC /SAMPLES TO BE RUN? /

FCB 3D, 3A

FCC /CENTER A 4 DIGIT /

FCC /HEXIDECIMAL VALUE) /

FILTER 3

DECEMBER 19, 1978

04DF 55 45
 04E1 29 20
 04E3 04

FCB 04

♦
 ♦
 ♦
 ♦ FILTER DATA PRINTER PROGRAM
 ♦

04E4 CE 00 01	PRINTR	LDX	#30001	
04E7 DF 51		STX	SAMP	
04E9 CE 18 00		LDX	#31800	
04EC DF 53		STX	RAMPTR	
04EE CE 30 00		LDX	#33000	
04F1 DF 57		STX	REPTR	
04F3 CE 38 00		LDX	#33800	
04F6 DF 59		STX	RDOTPT	
04F8 CE 06 18		LDX	#HEAD1	
04FB BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
04FE 96 5E		LDA A	TYPE	
0500 44		LSR A		
0501 24 0B		BCC	PRINT1	
0503 CE 06 43		LDX	#NHEAD	
0506 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0509 CE 20 00		LDX	#32000	
050C 20 17		BRA	PRINT3	
050E 44	PRINT1	LSR A		
050F 24 0B		BCC	PRINT2	
0511 CE 06 4E		LDX	#RNHEAD	
0514 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0517 CE 28 00		LDX	#32800	
051A 20 09		BRA	PRINT3	
051C CE 06 38	PRINT2	LDX	#AHEAD	
051F BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0522 CE 18 00		LDX	#31800	
0525 DF 55	PRINT3	STX	DATPTR	
0527 CE 06 59		LDX	#HEAD2	
052A BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
052D C6 05		LDA B	#305	
052F CE 06 74		LDX	#HEAD4	
0532 20 09		BRA	PRINT5	
0534 CE 06 80	PRINT4	LDX	#HEAD5	
0537 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
053A CE 06 73		LDX	#HEAD3	
053D BD E1 4B	PRINT5	JSR	\$E14B	MINIBUG PDATA1
0540 5A		DEC B		
0541 26 F1		BNE	PRINT4	
0543 CE 06 82	LOOP	LDX	#HEAD6	
0546 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0549 CE 00 51		LDX	#SAMP	

FILTER 3

DECEMBER 19, 1978

054C	DF	00		STX	XTEMP	
054E	20	2D		BRA	LOOPS	
0550	CE	06	8C	LDX	*HEAD7	
0553	BD	E1	4B	JSR	\$E14B	MINIBUG PDATA1
0556	DE	00		LDX	XTEMP	
0558	8C	00	59	CPX	*RDOTPT	
055B	26	1E		BNE	LOOPS	
055D	86	20		LDA A	*\$20	
055F	BD	E1	26	JSR	\$E126	OUTCH
0562	EE	00		LDX	0,X	
0564	E6	01		LDA B	1,X	
0566	A6	00		LDA A	0,X	
0568	2A	0B		BPL	LOOP2	
056A	36			PSH A		
056B	CE	06	95	LDX	*MINUS	
056E	BD	E1	4B	JSR	\$E14B	MINIBUG PDATA1
0571	32			PUL A		
0572	BD	03	FE	JSR	NEGATE	
0575	97	61		STA A	C1	
0577	D7	62		STA B	C2	
0579	20	06		BRA	LOOP4	
057B	EE	00		LDX	0,X	
057D	EE	00		LDX	0,X	
057F	DF	61		STX	C1	
0581	BD	05	BD	JSR	CONV	
0584	DE	00		LDX	XTEMP	
0586	08			INX		
0587	08			INX		
0588	DF	00		STX	XTEMP	
058A	8C	00	5B	CPX	*SAMPND	
058D	26	C1		BNE	LOOP1	
058F	DE	53		LDX	RAMPTR	
0591	08			INX		
0592	08			INX		
0593	DF	53		STX	RAMPTR	
0595	DE	55		LDX	DATPTR	
0597	08			INX		
0598	08			INX		
0599	DF	55		STX	DATPTR	
059B	DE	57		LDX	REPTR	
059D	08			INX		
059E	08			INX		
059F	DF	57		STX	REPTR	
05A1	DE	59		LDX	RDOTPT	
05A3	08			INX		
05A4	08			INX		
05A5	DF	59		STX	RDOTPT	
05A7	DE	51		LDX	SAMP	
05A9	9C	5B		CPX	SAMPND	

FILTER 3

DECEMBER 19, 1978

05A8 27 05		BEG	PEND	
05AD 08		INX		
05AE DF 51		STX	SAMP	
05B0 20 91		BRA	LOOP	
05B2 86 0A	PEND	LDA A	#30A	
05B4 06 05		LDA B	#305	
05B6 BD E1 26	PEND1	JSR	\$E126	MINIBUS DUTCH
05B9 5A		DEC B		
05BA 26 FA		BNE	PEND1	
05BC 3F		SWI		
♦				
♦				
05BD 7F 00 60	CONV	CLR	NFLAG	
05C0 7D 00 61		TST	C1	
05C3 2A 03		BPL	CONV1	
05C5 7A 00 60		DEC	NFLAG	
05C8 CE 27 10	CONV1	LDX	#32710	10,000
05CB BD 05 E8		JSR	FACTOR	
05CE CE 03 E8		LDX	#303E8	1,000
05D1 BD 05 E8		JSR	FACTOR	
05D4 CE 00 64		LDX	#30064	100
05D7 BD 05 E8		JSR	FACTOR	
05DA CE 00 0A		LDX	#3000A	10
05DD BD 05 E8		JSR	FACTOR	
05E0 96 62		LDA A	C2	
05E2 8A 30		ORA A	#30	
05E4 BD E1 26		JSR	\$E126	MINIBUS DUTCH
05E7 39		RTS		
♦				
♦				
05E8 DF 63	FACTOR	STX	D1	
05EA 7F 00 5D		CLR	DIGIT	
05ED BD 06 0B	FACT	JSR	SUB2BY	
05F0 2B 0C		BMI	FACTND	
05F2 7F 00 60		CLR	NFLAG	
05F5 7C 00 5D	FACT1	INC	DIGIT	
05F8 DE 65		LDX	DIFF1	
05FA DF 61		STX	C1	
05FC 20 EF		BRA	FACT	
05FE 7D 00 60	FACTND	TST	NFLAG	
0601 2B F2		BMI	FACT1	
0603 96 5D		LDA A	DIGIT	
0605 8A 30		ORA A	#30	
0607 BD E1 26		JSR	\$E126	
060A 39		RTS		
♦				
♦ 2 BYTE SYBTRACTION ROUTINE				
♦				
060B D6 62	SUB2BY	LDA B	C2	

FILTER 3

DECEMBER 19, 1978

060D 96 61
 060F D0 64
 0611 92 63
 0613 D7 66
 0615 97 65
 0617 39

LDA A C1
 SUB B D2
 SBC A D1
 STA B DIFF2
 STA A DIFF1
 RTS

0618 00
 0619 0A 0A
 061B 0A 0A
 061D 20
 061E 53 41
 0620 4D 50
 0622 4C 45
 0624 20 20
 0626 20 20
 0628 21 20
 062A 20 20
 062C 20 52
 062E 41 4D
 0630 50 20
 0632 20 20
 0634 20 21

HEAD1 FCB \$0D,\$0A,\$0A,\$0A,\$0A

FCC / SAMPLE ! RAMP ! /

0636 20
 0637 04
 0638 20
 0639 20 20
 063B 52 41
 063D 4D 50
 063F 20 20
 0641 20
 0642 04
 0643 20
 0644 20 20
 0646 4E 4F
 0648 49 53
 064A 45 20
 064C 20
 064D 04
 064E 52
 064F 41 4D
 0651 50 23
 0653 4E 4F
 0655 49 53
 0657 45
 0658 04
 0659 20
 065A 21 20

RHEAD FCB \$04
 FCC / RAMP /

NHEAD FCB \$04
 FCC / NOISE /

RNHEAD FCB \$04
 FCC /RAMP+NOISE/

HEAD2 FCB \$04
 FCC / ! RE ! RDOT/

FILTER 3

DECEMBER 19, 1978

```

0650 20 20
065E 20 20
0660 52 45
0662 20 20
0664 20 20
0666 20 21
0668 20 20
066A 20 20
066C 52 44
066E 4F 54
0670 0D      HEAD8   FCB   $0D,$0A,$04
0671 0A 04
0673 2D      HEAD3   FCC   /-/
0674 2D      HEAD4   FCC   /-----/
0675 2D 2D
0677 2D 2D
0679 2D 2D
067B 2D 2D
067D 2D 2D
067F 04      FCB     $04
0680 21      HEAD5   FCB   $21,$04
0681 04
0682 00 0D   HEAD6   FCB   $0D,$0A,$20,$20,$04
0684 00 0A
0686 00 20
0688 00 20
068A 00 04
068C 2D      HEAD7   FCC   /   !   /
068D 20 20
068F 20 21
0691 20 20
0693 20
0694 04      FCB     $04
0695 08      MINUS   FCB   $08,$2D,$04
0696 2D 04

```

END

NO ERROR(S) DETECTED

SYMBOL TABLE:

A1	000E	A2	000F	A3	0010
ADD	0406	B1	0011	B2	0012
B3	0013	C1	0061	C2	0062
CNT	0003	CONV	05ED	CONV1	0508
D1	0063	D2	0064	DATPTR	0055
DIFF1	0065	DIFF2	0066	DIGIT	005D
END	0361	FACT	05ED	FACT1	05F5
FACTND	05FE	FACTOR	05E8	FLT	0200

FILTER 3

DECEMBER 19, 1978

FLT1	0249	FLT10	02E9	FLT12	02EE
FLT14	032F	FLT2	0263	FLT3	0268
FLT4	027C	FLT5	0284	FLT6	029D
FLT7	02A0	FLTA	0241	HEAD1	0618
HEAD2	0659	HEAD3	0673	HEAD4	0674
HEAD5	0680	HEAD6	0682	HEAD7	068C
HEAD8	0670	INIT1	043C	INIT2	049E
INPUT1	0226	INPUT2	022E	INPUT3	0231
LOOP	0543	LOOP1	0550	LOOP2	0575
LOOP3	0578	LOOP4	0581	LOOP5	057D
MINUS	0695	MLTCD1	0004	MLTCD2	0005
MLTPR1	0006	MLTPR2	0007	MLTPR3	0008
MULT	0384	MULT1	037C	MULT2	0394
MULT3	039A	MULT31	03AA	MULT4	03B4
MULT5	03D5	MULTND	03F6	NEG1	0404
NEG2	0405	NEG3	03F9	NEG4	03FF
NEGATE	03FE	NFLAG	0060	NHEAD	0643
OUT	0337	OUT1	0341	PEND	05B2
PEND1	05B6	PRINT1	050E	PRINT2	051C
PRINT3	0525	PRINT4	0534	PRINT5	053D
PRINTR	04E4	PRDD1	0009	PRDD2	000A
PRDD3	000B	PRDD4	000C	PRDD5	000D
RAMPTR	0053	RCNT	0044	RCRCT1	0046
RCRCT2	0047	RCRCT3	0048	RCRCT4	0049
RDOT1	004A	RDOT2	004B	RDOTPT	0059
RE1	004C	RE2	004D	REGS	0020
REPTR	0057	RHEAD	0638	RNHEAD	064E
ROUND	005F	ROUND1	03B8	ROUND2	03CF
RTEMP	001E	SAMP	0051	SAMPND	005B
SGN	0002	SHFTR	042C	SHFTR1	0431
SKPFLG	0045	SUB	0419	SUB2BY	060B
SUM1	0014	SUM2	0015	SUM3	0016
TSUM1	004E	TSUM2	004F	TSUM3	0050
TYPE	003E	XTEMP	0000		

C-3

APPENDIX C

The following pages contain an assembly language listing of program FILTER 5.

FILTER 5

DECEMBER 19, 1978

♦
♦
♦
♦ FILTER 5
♦

0000	ORG	\$0000
0000	XTEMP RMB	2
0002	SGN RMB	1
0003	CNT RMB	1
0004	MLTCD1 RMB	1
0005	MLTCD2 RMB	1
0006	MLTPR1 RMB	1
0007	MLTPR2 RMB	1
0008	MLTPR3 RMB	1
0009	PRDD1 RMB	1
000A	PRDD2 RMB	1
000B	PRDD3 RMB	1
000C	PRDD4 RMB	1
000D	PRDD5 RMB	1
000E	A1 RMB	1
000F	A2 RMB	1
0010	A3 RMB	1
0011	B1 RMB	1
0012	B2 RMB	1
0013	B3 RMB	1
0014	SUM1 RMB	1
0015	SUM2 RMB	1
0016	SUM3 RMB	1
001E	ORG	\$001E
001E	RTEMP RMB	2
0020	REG3 RMB	32
0040	REGSND RMB	4
0044	ORG	\$0044
0044	RCNT RMB	1
0045	SKPFL5 RMB	1
0046	RDRCT1 RMB	1
0047	RDRCT2 RMB	1
0048	RDOT1 RMB	1
0049	RDOT2 RMB	1
004A	RE1 RMB	1
004B	RE2 RMB	1
004C	TSUM1 RMB	1
004D	TSUM2 RMB	1
004E	TSUM3 RMB	1
004F	SAMP RMB	2
0051	RAMPTR RMB	2
0053	DATPTR RMB	2
0055	REPTR RMB	2
0057	RDOTPT RMB	2

FILTER 5

DECEMBER 19, 1978

0059	SAMPND	RMB	2
005B	DIGIT	RMB	1
005C	TYPE	RMB	1
005D	ROUND	RMB	1
005E	NFLAG	RMB	1
005F	C1	RMB	1
0060	C2	RMB	1
0061	D1	RMB	1
0062	D2	RMB	1
0063	DIFF1	RMB	1
0064	DIFF2	RMB	1
0065	RDI	RMB	2
0067	ENDFLG	RMB	1

♦
♦

0200		ORG	\$0200	
0200 C6 01	FLT	LDA B	#\$01	
0202 D7 5D		STA B	ROUND	
0204 CE 00 01		LDX	#\$0001	
0207 DF 4F		STX	SAMP	
0209 CE 30 00		LDX	#\$3000	
020C DF 55		STX	REPTR	
020E CE 38 00		LDX	#\$3800	
0211 DF 57		STX	RDOTPT	
0213 CE 04 1F		LDX	#INIT1	
0216 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0219 BD E1 33		JSR	\$E133	MINIBUG INCH
021C 97 5C		STA A	TYPE	
021E 44		LSR A		
021F 24 05		BCC	INPUT1	
0221 CE 20 00		LDX	#\$2000	
0224 20 0B		BRA	INPUT3	
0226 44	INPUT1	LSR A		
0227 24 05		BCC	INPUT2	
0229 CE 28 00		LDX	#\$2800	
022C 20 03		BRA	INPUT3	
022E CE 18 00	INPUT2	LDX	#\$1800	
0231 DF 53	INPUT3	STX	DATPTR	
0233 CE 04 81		LDX	#INIT2	
0236 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0239 BD E0 F8		JSR	\$E0F8	MINIBUG BADDR
023C DF 59		STX	SAMPND	
023E CE 00 20		LDX	#REGS	
0241 6F 00	FLTA	CLR	0:X	
0243 03		INX		
0244 8C 00 44		CPX	#RCNT	
0247 26 F8		BNE	FLTA	
0249 7F 00 67		CLR	ENDFLG	
024C C6 10		LDA B	#\$10	

FILTER 5

DECEMBER 19, 1978

024E D7 44		STA B	RCNT
0250 DE 53		LDX	DATPTR
0252 EE 00		LDX	0,X
0254 DF 65		STX	RDI
0256 DE 53	FLT1	LDX	DATPTR
0258 EE 00		LDX	0,X
025A DF 0F		STX	A2
025C DE 65		LDX	RDI
025E DF 12		STX	B2
0260 7F 00 0E		CLR	A1
0263 7F 00 11		CLR	B1
0266 ED 03 FC		JSR	SUB
0269 DE 15		LDX	SUM2
026B ED 04 0F		JSR	SHFTR
026E DE 53		LDX	DATPTR
0270 08		INX	
0271 08		INX	
0272 DF 53		STX	DATPTR
0274 DE 4F		LDX	SAMP
0276 08		INX	
0277 DF 4F		STX	SAMP
0279 9C 59		CPX	SAMPND
027B 26 03		BNE	FLT11
027D 7C 00 67		INC	ENDFLG
0280 7A 00 44	FLT11	DEC	RCNT
0283 26 D1		BNE	FLT1
0285 C6 10		LDA B	#\$10
0287 D7 44		STA B	RCNT
0289 7F 00 14		CLR	SUM1
028C 7F 00 15		CLR	SUM2
028F 7F 00 16		CLR	SUM3
0292 CE 00 20		LDX	#REGS
0295 DF 00	FLT2	STX	XTEMP
0297 CE 00 0E		LDX	#A1
029A A6 06	FLT3	LDA A	6,X
029C A7 00		STA A	0,X
029E 08		INX	
029F 8C 00 11		CPX	#B1
02A2 26 F6		BNE	FLT3
02A4 7F 00 11		CLR	B1
02A7 DE 00		LDX	XTEMP
02A9 EE 00		LDX	0,X
02AB 2A 03		BPL	FLT4
02AD 7A 00 11		DEC	B1
02B0 DF 12	FLT4	STX	B2
02B2 BD 03 E9		JSR	ADD
02B5 DE 00		LDX	XTEMP
02B7 08		INX	
02B8 08		INX	

FILTER 5

DECEMBER 19, 1978

```

02B9 8C 00 40
02BC 26 D7
02BE 96 14
02C0 97 4C
02C2 97 06
02C4 DE 15
02C6 DF 4D
02C8 DF 07
02CA CE 06 40
02CD DF 04
02CF BD 03 47
02D2 DE 0A
02D4 DF 48
02D6 D6 4C
02D8 D7 06
02DA DE 4D
02DC DF 07
02DE CE 10 00
02E1 DF 04
02E3 BD 03 47
02E6 96 09
02E8 97 11
02EA DE 0A
02EC DF 12
02EE DE 65
02F0 DF 0F
02F2 7F 00 0E
02F5 BD 03 E9
02F8 DE 15
02FA DF 65
02FC D6 4C
02FE D7 06
0300 DE 4D
0302 DF 07
0304 CE 17 00
0307 DF 04
0309 BD 03 47
030C 96 09
030E 97 11
0310 DE 0A
0312 DF 12
0314 BD 03 E9
0317 DE 15
0319 DF 4A
031B DE 55
031D 96 4A
031F D6 4B
0321 A7 00
0323 E7 01

```

OUT

```

CPX    #REGSND
BNE    FLT2
LDA A   SUM1
STA A   TSUM1
STA A   MLTPR1
LDX     SUM2
STX     TSUM2
STX     MLTPR2
LDX     #80640
STX     MLTCD1
JSR     MULT
LDX     PRDD2
STX     RDOT1
LDA B   TSUM1
STA B   MLTPR1
LDX     TSUM2
STX     MLTPR2
LDX     #81000
STX     MLTCD1
JSR     MULT
LDA A   PRDD1
STA A   B1
LDX     PRDD2
STX     B2
LDX     RDI
STX     A2
CLR     A1
JSR     ADD
LDX     SUM2
STX     RDI
LDA B   TSUM1
STA B   MLTPR1
LDX     TSUM2
STX     MLTPR2
LDX     #81780
STX     MLTCD1
JSR     MULT
LDA A   PRDD1
STA A   B1
LDX     PRDD2
STX     B2
JSR     ADD
LDX     SUM2
STX     RE1
LDX     REPTR
LDA A   RE1
LDA B   RE2
STA A   0,X
STA B   1,X

```

FILTER 5

DECEMBER 19, 1978

0325 08		INX	
0326 08		INX	
0327 DF 55		STX	REPTR
0329 DE 57	OUT1	LDX	RDOTPT
032B 96 48		LDA A	RDOT1
032D D6 49		LDA B	RDOT2
032F A7 00		STA A	0,X
0331 E7 01		STA B	1,X
0333 08		INX	
0334 08		INX	
0335 DF 57		STX	RDOTPT
0337 7D 00 67		TST	ENDFLG
033A 26 03		BNE	END
033C 7E 02 56		JMP	FLT1
033F DE 4F	END	LDX	SAMP
0341 09		DEX	
0342 DF 59		STX	SAMPND
0344 7E 04 C7		JMP	PRINTR

♦
♦ 3 BYTE X 2 BYTE 2'S COMPLEMENT MULTIPLY
♦

0347 DF 00	MULT	STX	XTEMP
0349 CE 00 02		LDX	#SGN
034C 6F 00		CLR	0,X
034E 6C 00		INC	0,X
0350 A6 02		LDA A	2,X
0352 2A 0B		BPL	MULT1
0354 E6 03		LDA B	3,X
0356 BD 03 E1		JSR	NEGATE
0359 A7 02		STA A	2,X
035B E7 03		STA B	3,X
035D 6A 00		DEC	0,X
035F A6 04	MULT1	LDA A	4,X
0361 2A 14		BPL	MULT2
0363 A6 05		LDA A	5,X
0365 E6 06		LDA B	6,X
0367 BD 03 E1		JSR	NEGATE
036A A7 05		STA A	5,X
036C E7 06		STA B	6,X
036E A6 04		LDA A	4,X
0370 BD 03 E2		JSR	NEG4
0373 A7 04		STA A	4,X
0375 6A 00		DEC	0,X
0377 86 E7	MULT2	LDA A	#E7
0379 A7 01		STA A	1,X
037B 4F		CLR A	
037C 5F		CLR B	
037D 6C 01	MULT3	INC	1,X
037F 27 16		BEQ	MULT4

FILTER 5

DECEMBER 19, 1978

0381	64	04		LSR	4,X
0383	66	05		ROR	5,X
0385	66	06		ROR	6,X
0387	24	04		BCC	MULT31
0389	E8	03		ADD B	3,X
038B	A9	02		ADC A	2,X
038D	44		MULT31	LSR A	
038E	56			ROR B	
038F	66	09		ROR	9,X
0391	66	0A		ROR	10,X
0393	66	0B		ROR	11,X
0395	20	E6		BRA	MULT3
0397	A7	07	MULT4	STA A	7,X
0399	E7	08		STA B	8,X
039B	7D	00	ROUND1	TST	ROUND
039E	27	18		BEQ	MULT5
03A0	7D	00	0C	TST	PROD4
03A3	2A	0D		BPL	ROUND2
03A5	7C	00	0B	INC	PROD3
03A8	26	08		BNE	ROUND2
03AA	7C	00	0A	INC	PROD2
03AD	26	03		BNE	ROUND2
03AF	7C	00	09	INC	PROD1
03B2	7F	00	0C	CLR	PROD4
03B5	7F	00	0D	CLR	PROD5
03B8	66	00	MULT5	ROR	0,X
03BA	25	1D		BCS	MULTND
03BC	A6	0A		LDA A	10,X
03BE	E6	0B		LDA B	11,X
03C0	BD	03	E1	JSR	NEGATE
03C3	A7	0A		STA A	10,X
03C5	E7	0B		STA B	11,X
03C7	A6	08		LDA A	8,X
03C9	E6	09		LDA B	9,X
03CB	BD	03	DC	JSR	NEG3
03CE	A7	08		STA A	8,X
03D0	E7	09		STA B	9,X
03D2	A6	07		LDA A	7,X
03D4	BD	03	E2	JSR	NEG4
03D7	A7	07		STA A	7,X
03D9	DE	00	MULTND	LDX	XTEMP
03DB	39			RTS	

♦
♦ 2'S COMPLEMENT ROUTINE
♦

03DC	24	03	NEG3	BCC	NEGATE
03DE	53			COM B	
03DF	20	06		BRA	NEG1
03E1	50		NEGATE	NEG B	

FILTER 5

DECEMBER 19, 1978

03E2 25 03
 03E4 40
 03E5 20 01
 03E7 43
 03E8 39

NEG4 BCS NEG1
 NEG A
 BRR NEG2
 COM A
 RTS

♦

♦ 3 BYTE ADDITION ROUTINE

♦

03E9 D6 10
 03EB 96 0F
 03ED DB 13
 03EF 99 12
 03F1 D7 16
 03F3 97 15
 03F5 96 0E
 03F7 99 11
 03F9 97 14
 03FB 39

ADD LDA B A3
 LDA A A2
 ADD B B3
 ADC A B2
 STA B SUM3
 STA A SUM2
 LDA A A1
 ADC A B1
 STA A SUM1
 RTS

♦

♦ 3 BYTE SUBTRACTION ROUTINE

♦

03FC D6 10
 03FE 96 0F
 0400 D0 13
 0402 92 12
 0404 D7 16
 0406 97 15
 0408 96 0E
 040A 92 11
 040C 97 14
 040E 39

SUB LDA B A3
 LDA A A2
 SUB B B3
 SBC A B2
 STA B SUM3
 STA A SUM2
 LDA A A1
 SBC A B1
 STA A SUM1
 RTS

♦

♦ ROUTINE TO SHIFT DATA IN REGS

♦

040F DF 1E
 0411 CE 00 41
 0414 E6 00
 0416 E7 02
 0418 09
 0419 8C 00 1D
 041C 26 F6
 041E 39

SHFTR STX RTEMP
 LDX #0041
 SHFTR1 LDA B 0,X
 STA B 2,X
 DEX
 CPX #001D
 BNE SHFTR1
 RTS

♦

041F 0D
 0420 0A 0A
 0422 0A 0A
 0424 0A 0A
 0426 46
 0427 49 4C

INIT1 FCB \$D,\$A,\$A,\$A,\$A,\$A,\$A

FCC /FILTER 5 MICROPROCESSOR/

FILTER 5

DECEMBER 19, 1978

0429 54 45
 042B 52 20
 042D 35 20
 042F 4D 49
 0431 43 52
 0433 4F 50
 0435 52 4F
 0437 43 45
 0439 53 53
 043B 4F 52
 043D 20
 043E 53 49
 0440 4D 55
 0442 4C 41
 0444 54 49
 0446 4F 4E
 0448 20 52
 044A 55 4E
 044C 00
 044D 0A 0A
 044F 49
 0450 4E 50
 0452 55 54
 0454 20 43
 0456 48 4F
 0458 49 43
 045A 45 3F
 045C 00
 045D 0A
 045E 28
 045F 53 49
 0461 47 4E
 0463 41 4C
 0465 3D 30
 0467 2C 4E
 0469 4F 49
 046B 53 45
 046D 3D 31
 046F 2C
 0470 53
 0471 49 47
 0473 4E 41
 0475 4C 28
 0477 4E 4F
 0479 49 53
 047B 45 3D
 047D 32 29
 047F 20
 0480 04

FCC / SIMULATION RUN/

FCB \$D,\$A,\$A

FCC /INPUT CHOICE?/

FCB \$D,\$A

FCC / (SIGNAL=0, NOISE=1, /

FCC /SIGNAL+NOISE=2) /

FCB 04

FILTER 5

DECEMBER 19, 1978

0481 0D	INIT2	FCB	\$D,\$A,\$A
0482 0A 0A			
0484 4E		FCC	/NUMBER OF /
0485 55 4D			
0487 42 45			
0489 52 20			
048B 4F 46			
048D 20			
048E 53		FCC	/SAMPLES TO BE RUN?/
048F 41 4D			
0491 50 4C			
0493 45 53			
0495 20 54			
0497 4F 20			
0499 42 45			
049B 20 52			
049D 55 4E			
049F 3F			
04A0 0D		FCB	\$D,\$A
04A1 0A			
04A2 28		FCC	/CENTER A 4 DIGIT /
04A3 45 4E			
04A5 54 45			
04A7 52 20			
04A9 41 20			
04AB 34 20			
04AD 44 49			
04AF 47 49			
04B1 54 20			
04B3 43		FCC	/HEXIDECIMAL VALUE) /
04B4 45 58			
04B6 49 44			
04B8 45 43			
04BA 49 4D			
04BC 41 4C			
04BE 20 56			
04C0 41 4C			
04C2 55 45			
04C4 29 20			
04C6 04		FCB	04

◆
◆
◆
◆ FILTER DATA PRINTER PROGRAM

04C7 CE 00 10	PRINTR	LDX	#\$0010
04C8 DF 4F		STX	SAMP
04CC CE 18 1E		LDX	#\$181E
04CF DF 51		STX	RAMPTR

FILTER 5

DECEMBER 19, 1978

04D1 CE 30 00		LDX	#3000	
04D4 DF 55		STX	REPTR	
04D6 CE 38 00		LDX	#3800	
04D9 DF 57		STX	RDOTPT	
04DB CE 06 11		LDX	#HEAD1	
04DE BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
04E1 96 5C		LDA A	TYPE	
04E3 44		LSR A		
04E4 24 0B		BCC	PRINT1	
04E6 CE 06 3C		LDX	#NHEAD	
04E9 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
04EC CE 20 1E		LDX	#201E	
04EF 20 17		BRA	PRINT3	
04F1 44	PRINT1	LSR A		
04F2 24 0B		BCC	PRINT2	
04F4 CE 06 47		LDX	#RNHEAD	
04F7 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
04FA CE 28 1E		LDX	#281E	
04FD 20 09		BRA	PRINT3	
04FF CE 06 31	PRINT2	LDX	#RHEAD	
0502 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0505 CE 18 1E		LDX	#181E	
0508 DF 53	PRINT3	STX	DATPTR	
050A CE 06 52		LDX	#HEAD2	
050D BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0510 C6 05		LDA B	#305	
0512 CE 06 6D		LDX	#HEAD4	
0515 20 09		BRA	PRINT5	
0517 CE 06 79	PRINT4	LDX	#HEAD5	
051A BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
051D CE 06 6C		LDX	#HEAD3	
0520 BD E1 4B	PRINT5	JSR	\$E14B	MINIBUG PDATA1
0523 5A		DEC B		
0524 26 F1		BNE	PRINT4	
0526 CE 06 7B	LOOP	LDX	#HEAD6	
0529 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
052C CE 00 4F		LDX	#SAMP	
052F DF 00		STX	XTEMP	
0531 20 2D		BRA	LOOP5	
0533 CE 06 85	LOOP1	LDX	#HEAD7	
0536 BD E1 4B		JSR	\$E14B	MINIBUG PDATA1
0539 DE 00		LDX	XTEMP	
053B 8C 00 57		CPX	#RDOTPT	
053E 26 1E		BNE	LOOP3	
0540 86 20		LDA A	#320	
0542 BD E1 26		JSR	\$E126	OUTCH
0545 EE 00		LDX	0,X	
0547 E6 01		LDA B	1,X	
0549 A6 00		LDA A	0,X	

FILTER 5

DECEMBER 19, 1978

054B 2A 0B		BPL	LOOP2
054D 36		PSH A	
054E CE 06 8E		LDX	#MINUS
0551 BD E1 4B		JSR	\$E14B MINIBUG PDATA1
0554 32		PUL A	
0555 BD 03 E1		JSR	NEGATE
0558 97 5F	LOOP2	STA A	C1
055A D7 60		STA B	C2
055C 20 06		BRA	LOOP4
055E EE 00	LOOP3	LDX	0,X
0560 EE 00	LOOP5	LDX	0,X
0562 DF 5F		STX	C1
0564 BD 05 B6	LOOP4	JSR	CONV
0567 DE 00		LDX	XTEMP
0569 08		INX	
056A 08		INX	
056B DF 00		STX	XTEMP
056D 8C 00 59		CPX	#SAMPND
0570 26 C1		BNE	LOOP1
0572 96 51		LDA A	RAMPTR
0574 D6 52		LDA B	RAMPTR+1
0576 CB 20		ADD B	#\$20
0578 89 00		ADC A	#\$00
057A 97 51		STA A	RAMPTR
057C D7 52		STA B	RAMPTR+1
057E 96 53		LDA A	DATPTR
0580 D6 54		LDA B	DATPTR+1
0582 CB 20		ADD B	#\$20
0584 89 00		ADC A	#\$00
0586 97 53		STA A	DATPTR
0588 D7 54		STA B	DATPTR+1
058A DE 55		LDX	REPTR
058C 08		INX	
058D 08		INX	
058E DF 55		STX	REPTR
0590 DE 57		LDX	RDOTPT
0592 08		INX	
0593 08		INX	
0594 DF 57		STX	RDOTPT
0596 DE 4F		LDX	SAMP
0598 9C 59		CPX	SAMPND
059A 27 0F		BEQ	PEND
059C 96 4F		LDA A	SAMP
059E D6 50		LDA B	SAMP+1
05A0 CB 10		ADD B	#\$10
05A2 89 00		ADC A	#\$00
05A4 97 4F		STA A	SAMP
05A6 D7 50		STA B	SAMP+1
05A8 7E 05 26		JMP	LOOP

FILTER 5

DECEMBER 19, 1973

```

05AB 86 0A      PEND    LDA A  #30A
05AD C6 05      LDA B  #305
05AF BD E1 26   PEND1   JSR    $E126      MINIBUS OUTCH
05B2 5A                DEC B
05B3 26 FA      BNE     PEND1
05B5 3F                SWI

```

```

♦
♦
05B6 7F 00 5E   CONV    CLR     NFLAG
05B9 7D 00 5F      TST     C1
05BC 2A 03      BPL     CONV1
05BE 7A 00 5E      DEC     NFLAG
05C1 CE 27 10   CONV1   LDX     #32710      10,000
05C4 BD 05 E1      JSR     FACTOR
05C7 CE 03 E8      LDX     #303E8      1,000
05CA BD 05 E1      JSR     FACTOR
05CD CE 00 64      LDX     #30064      100
05D0 BD 05 E1      JSR     FACTOR
05D3 CE 00 0A      LDX     #3000A      10
05D6 BD 05 E1      JSR     FACTOR
05D9 96 60      LDA A  C2
05DB 8A 30      ORA A  #330
05DD BD E1 26   JSR     $E126      MINIBUS OUTCH
05E0 39      RTS

```

```

♦
♦
05E1 DF 61      FACTOR   STX     D1
05E3 7F 00 5B      CLR     DIGIT
05E6 BD 06 04   FACT     JSR     SUB2BY
05E9 2B 0C      BMI     FACTND
05EB 7F 00 5E      CLR     NFLAG
05EE 7C 00 5B   FACT1    INC     DIGIT
05F1 DE 63      LDX     DIFF1
05F3 DF 5F      STX     C1
05F5 20 EF      BRA     FACT
05F7 7D 00 5E   FACTND   TST     NFLAG
05FA 2B F2      BMI     FACT1
05FC 96 5B      LDA A  DIGIT
05FE 8A 30      ORA A  #330
0600 BD E1 26   JSR     $E126
0603 39      RTS

```

♦ 2 BYTE SUBTRACTION ROUTINE ♦

```

0604 D6 60   SUB2BY    LDA B  C2
0606 96 5F      LDA A  C1
0608 D0 62      SUB B  D2
060A 92 61      SBC A  D1
060C D7 64      STA B  DIFF2

```

FILTER 5

DECEMBER 19, 1978

060E 97 63		STA A	DIFF1		
0610 39		RTS			
	♦				
	♦				
0611 0D	HEAD1	FCB	\$0D,\$0A,\$0A,\$0A,\$0A		
0612 0A 0A					
0614 0A 0A					
0616 20		FCC	/ SAMPLE	:	RAMP
0617 53 41					
0619 4D 50					
061B 4C 45					
061D 20 20					
061F 20 20					
0621 21 20					
0623 20 20					
0625 20 52					
0627 41 4D					
0629 50 20					
062B 20 20					
062D 20 21					
062F 20					
0630 04		FCB	\$04		
0631 20	RHEAD	FCC	/ RAMP	/	
0632 20 20					
0634 52 41					
0636 4D 50					
0638 20 20					
063A 20					
063B 04		FCB	\$04		
063C 20	NHEAD	FCC	/ NOISE	/	
063D 20 20					
063F 4E 4F					
0641 49 53					
0643 45 20					
0645 20					
0646 04		FCB	\$04		
0647 52	RNHEAD	FCC	/RAMP+NOISE/		
0648 41 4D					
064A 50 2B					
064C 4E 4F					
064E 49 53					
0650 45					
0651 04		FCB	\$04		
0652 20	HEAD2	FCC	/ :	RE	:
0653 21 20					RDOT/
0655 20 20					
0657 20 20					
0659 52 45					
065B 20 20					

FILTER 5

DECEMBER 19, 1978

```

065D 20 20
065F 20 21
0661 20 20
0663 20 20
0665 32 44
0667 4F 54
0669 0D      HEAD3   FCB      $0D,$0A,$04
066A 0A 04
066C 2D      HEAD3   FCC      /~/
066D 2D      HEAD4   FCC      /-----/
066E 2D 2D
0670 2D 2D
0672 2D 2D
0674 2D 2D
0676 2D 2D
0678 04
0679 21      HEAD5   FCB      $21,$04
067A 04
067B 00 0D   HEAD6   FDB      $0D,$0A,$20,$20,$04
067D 00 0A
067F 00 20
0681 00 20
0683 00 04
0685 2D      HEAD7   FCC      /      !      /
0686 20 20
0688 20 21
068A 20 20
068C 20
068D 04
068E 08      MINUS   FCB      $08,$2D,$04
068F 2D 04

```

END

NO ERROR(S) DETECTED

SYMBOL TABLE:

A1	000E	A2	000F	A3	0010
ADD	03E9	B1	0011	B2	0012
B3	0013	C1	005F	C2	0060
CNT	0003	CONV	05B6	CONV1	05C1
D1	0061	D2	0062	DATPTR	0053
DIFF1	0063	DIFF2	0064	DIGIT	005B
END	033F	ENDFLG	0067	FACT	05E6
FACT1	05EE	FACTND	05F7	FACTOR	05E1
FLT	0200	FLT1	0256	FLT11	0280
FLT2	0295	FLT3	029A	FLT4	02B0
FLTA	0241	HEAD1	0611	HEAD2	0652
HEAD3	066C	HEAD4	066D	HEAD5	0679

FILTER 5

DECEMBER 19, 1978

HEAD6	067B	HEAD7	0685	HEAD8	0669
INIT1	041F	INIT2	0481	INPUT1	0226
INPUT2	022E	INPUT3	0231	LOOP	0526
LOOP1	0533	LOOP2	0558	LOOP3	055E
LOOP4	0564	LOOP5	0560	MINUS	068E
MLTCD1	0004	MLTCD2	0005	MLTPR1	0006
MLTPR2	0007	MLTPR3	0008	MULT	0347
MULT1	035F	MULT2	0377	MULT3	037D
MULT31	038D	MULT4	0397	MULT5	0388
MULTND	03D9	NEG1	03E7	NEG2	03E8
NEG3	03DC	NEG4	03E2	NEGATE	03E1
NFLAG	005E	NHEAD	063C	OUT	031B
OUT1	0329	PEND	05C3	PEND1	05AF
PRINT1	04F1	PRINT2	04FF	PRINT3	0508
PRINT4	0517	PRINT5	0520	PRINTR	04C7
PRDD1	0009	PRDD2	000A	PRDD3	000B
PRDD4	000C	PRDD5	000D	RAMPTR	0051
RCNT	0044	RORCT1	0046	RORCT2	0047
RDI	0065	RDOT1	0048	RDOT2	0049
RDOTPT	0057	RE1	004A	RE2	004B
REGS	0020	REGSND	0040	REPTR	0055
RHEAD	0631	RNHEAD	0647	ROUND	005D
ROUND1	039B	ROUND2	03B2	RTEMP	001E
SAMP	004F	SAMPND	0059	SGN	0002
SHFTR	040F	SHFTR1	0414	SKPFLG	0045
SUB	03FC	SUB2BY	0604	SUM1	0014
SUM2	0015	SUM3	0016	TSUM1	004C
TSUM2	004D	TSUM3	004E	TYPE	005C
XTEMP	0000				

APPENDIX D

The program to determine variance reduction ratio from the z transfer function is presented on the following page.

PROGRAM TO DETERMINE VARIANCE REDUCTION RATIO FROM Z TRANSFER FUNCTION

```

10 DIMN(50),D(50)
15 S=0
20 PRINT"NUMBER OF TERMS RESULT IS TO BE CARRIED OUT = ";
25 INPUT T
40 PRINT"ORDER OF NUMERATOR = ";
50 INPUT A
60 FOR J=1 TO A+1
70 PRINT"N(";J-1;")=";
80 INPUT N(J)
90 NEXT J
100 PRINT"ORDER OF DENOMINATOR = ";
110 INPUT B
120 FOR J=1 TO B+1
130 PRINT"D(";J-1;")=";
140 INPUT D(J)
150 NEXT J
155 PRINT
160 IF A>=B THEN 210
170 A=B
180 FOR J=A+2 TO B+1
190 N(J)=0
200 NEXT J
210 FOR L=0 TO T
220 M=N(1)/D(1)
230 FOR J=2 TO B+1
240 N(J)=N(J)-M*D(J)
250 NEXT J
260 S=S+M**2
270 PRINT"N(";L;")=";M;TAB(28);"SUMMATION OF H**2 =";S
280 FOR J=1 TO A
290 N(J)=N(J+1)
300 NEXT J
310 N(A+1)=0
330 NEXT L
360 END

```

READY

APPENDIX E

A method for deriving a discrete system from a continuous system is to replace the integrators in the continuous system with numerical integrators.⁹ The resulting discrete system will be different for each method of numerical integration used. The simplest method is Euler's approximation where the time derivative is approximated by

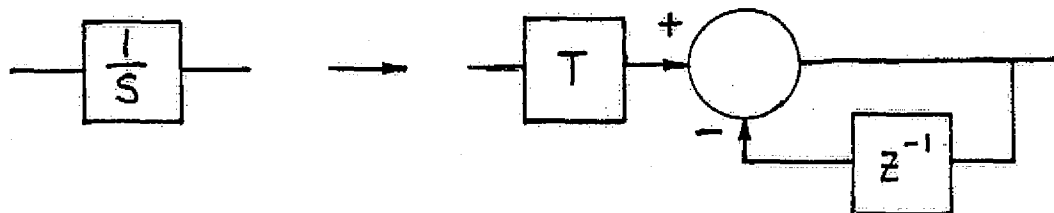
$$\frac{d}{dt}f(nT) \approx \frac{f(nT) - f((n-1)T)}{T} . \quad (37)$$

Taking the Laplace operator of the left side of (37) and equating it to the z operator of the right side of (37) yields the mapping from the s-plane to the z-plane

$$s = \frac{1-z^{-1}}{T} . \quad (38)$$

The continuous integrator can then be replaced by

$$\frac{1}{s} = \frac{T}{1-z^{-1}} . \quad (39)$$



Although with this method a discrete system can be obtained for any continuous system, the discrete system may not be physically realizable on a digital computer. Due to the finite time required for computations, every feedback loop must contain a delay element.