

AN APPLICATION OF ADAPTIVE
COMPLEX PREDICTION

by

WAYNE MICHAEL BLASI

B.S., Kansas State University, 1981

A MASTER'S THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1983

Approved by

Nasir Ahmed
Major Professor

LD
2668
.T4
1983
BS8
C.2

A11202 578018

ii

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. SOME FUNDAMENTALS	2
Demodulation Considerations	4
Signal Properties	7
III. SIGNAL MODELS	8
Target	8
Clutter	9
IV. DETECTION ALGORITHMS.	13
Standard Direction Sensitive Algorithms	13
Improved Direction Sensitive Algorithms	15
Complex Prediction	17
Direction Sensing	20
Signal Strength Equalizer	22
Integration and Threshold Detection	25
V. ROLE OF THE CLMS PREDICTOR	26
A Simplified Model	26
VI. SIMULATION RESULTS	40
Interpretation of Results	43
VII. CONCLUSIONS.	58
REFERENCES	59
ACKNOWLEDGMENTS	60
APPENDIX A: COMPLEX PREDICTOR DERIVATION	61
APPENDIX B: PROCESSOR FILTER REALIZATION	66
APPENDIX C: COMPUTER PROGRAMS	67

LIST OF FIGURES

FIGURE		PAGE
1	Location of Transmitter and Target	3
2	Inphase/Quadrature Demodulation	5
3	Target Signal Power Spectra.	10
4	Clutter Signal Power Spectra	12
5	Standard Detection Algorithm	16
6	Modified Standard Detection Algorithm	18
7	Complex Least Mean Square Adaptive Predictor	19
8	Direction Sensing Algorithm	21
9	Characteristics for Filter Realization of Kalmus Processor.	23
10	Signal Strength Equalizer Block Diagram	24
11	Triangular Target Signal with One-Sided Power Den- sity Spectrum (PDS)	27
12	a) Hypothetical Clutter Model (HCM); Case 1	31
	b) HCM; Case 1	32
	c) HCM Integrator Outputs; Case 1	33
13	a) HCM; Case 2	34
	b) HCM; Case 2	35
	c) HCM Integrator Outputs; Case 2	36
14	a) HCM; Case 3	37
	b) HCM; Case 3	38
	c) HCM: Integrator Output; Case 3	39
15	$r(t)$ Motion Function for Clutter Simulation	42
16	a) Simulation Clutter Model (SCM); Case 1	46
	b) SCM; Case 1	47
	c) SCM Integrator Output; Case 1	48

List of Figures (cont.)

FIGURE		PAGE
17	a) SCM; Case 2	.49
	b) SCM; Case 2	.50
	c) SCM Integrator Output, Case 2	.51
18	a) SCM; Case 3	.52
	b) SCM; Case 3	.53
	c) SCM Integrator Output; Case 3	.54
19	a) SCM; Case 4	.55
	b) SCM; Case 4	.56
	c) SCM Integrator Output; Case 4	.57
A1	Complex Least Mean Square Adaptive Predictor Block Diagram	.65

LIST OF TABLES

TABLE		PAGE
1	Hypothetical Clutter Model	.30
2	Simulation Clutter Model	.45

I. INTRODUCTION

The ability to detect a moving target using inphase-quadrature signals of a Doppler radar system depends on the nature of the clutter signal accompanying the signal to be detected. We consider a class of clutter signals produced by the oscillatory movement of an object or objects within the detection region. This paper introduces the notion of using adaptive complex prediction to improve target detection in this class of clutter signals, where the moving target is assumed to produce relatively low amplitude transient signals that are typically broadband. The target is also assumed to be moving in one direction -- i.e., towards or away from the radar.

The results of this development are general in the sense that they can be applied to arbitrary frequency ranges and sampling rates. The simulation results however are based on a sampling rate of 10 samples per second (sps) which implies a bandwidth of 0 to 5 hertz. This choice was made because the signal to noise (SNR) characteristics in this region are typically poor, which makes detection difficult.

II. SOME FUNDAMENTALS

The signals of interest result from the received radar signals containing Doppler frequency shift information. A simple but useful model is presented with emphasis placed on signal processing concepts with no assumptions made about transmitter/receiver characteristics other than their general form.

Referring to Figure 1, it is assumed that a radar transmitter/receiver is located at point A. A target T starts moving at time zero from location B with a velocity v which is positive for increasing and negative for decreasing distance. We let $e_1(t)$ and $e_2(t)$ denote the transmitted and received signals respectively. If

$$e_1(t) = E \sin \omega t \quad (1)$$

then

$$e_2(t) = KE \sin \omega \left(t - \frac{2D}{c} \right) \quad (2)$$

where

ω is the carrier frequency

K is the radar loop gain factor

c is the velocity of light

E is the amplitude of the transmitted signal.

From Figure 1, we have

$$D = D' \pm vt \quad (3)$$

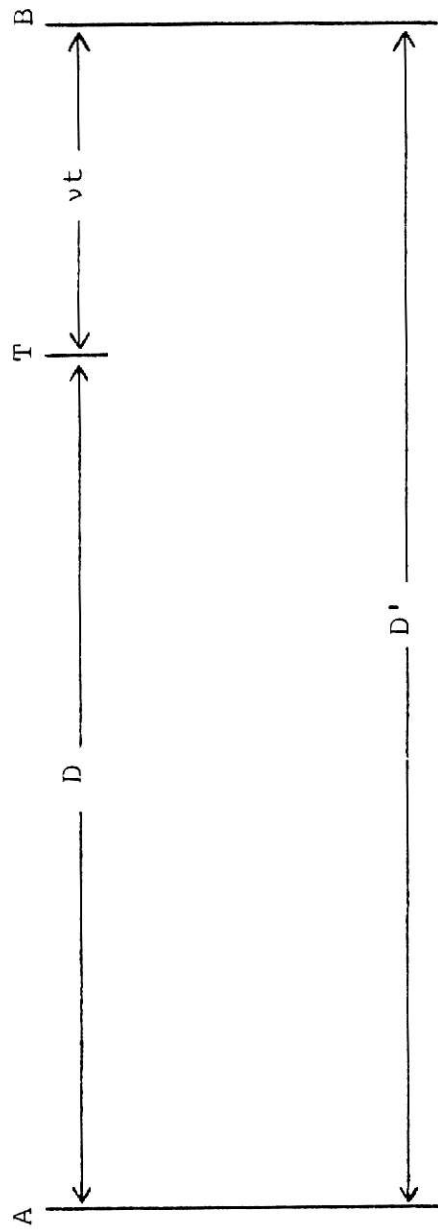


Figure 1. Location of Transmitter and target

Thus, the received signal becomes

$$e_2(t) = KE \sin w \left\{ t - \left(\frac{2D'}{c} \pm \frac{2vt}{c} \right) \right\} \quad (4)$$

or

$$e_2(t) = KE \sin \left(wt - \frac{2D'w}{c} \mp 2 \frac{v}{c} wt \right) \quad (5)$$

Defining the quantities

$$\alpha = \frac{2D'w}{c} \quad (6a)$$

which is a constant phase angle and

$$w_d = 2 \left(\frac{v}{c} \right) w \quad (6b)$$

which is the angular doppler frequency, (5) can be written

$$e_2(t) = KE \sin (wt - \alpha \mp w_d t) \quad (6)$$

Since α in (6) represents a constant phase angle, it can be ignored for analysis purposes and hence, the received signal prior to demodulation can be expressed as

$$e_2(t) = KE \sin (wt \mp w_d t) \quad (7)$$

Demodulation Considerations

The received signal in (7) is demodulated as shown in Figure 2 to obtain the inphase (I) and quadrature (Q) signals, $i(t)$ and $q(t)$, respectively as follows:

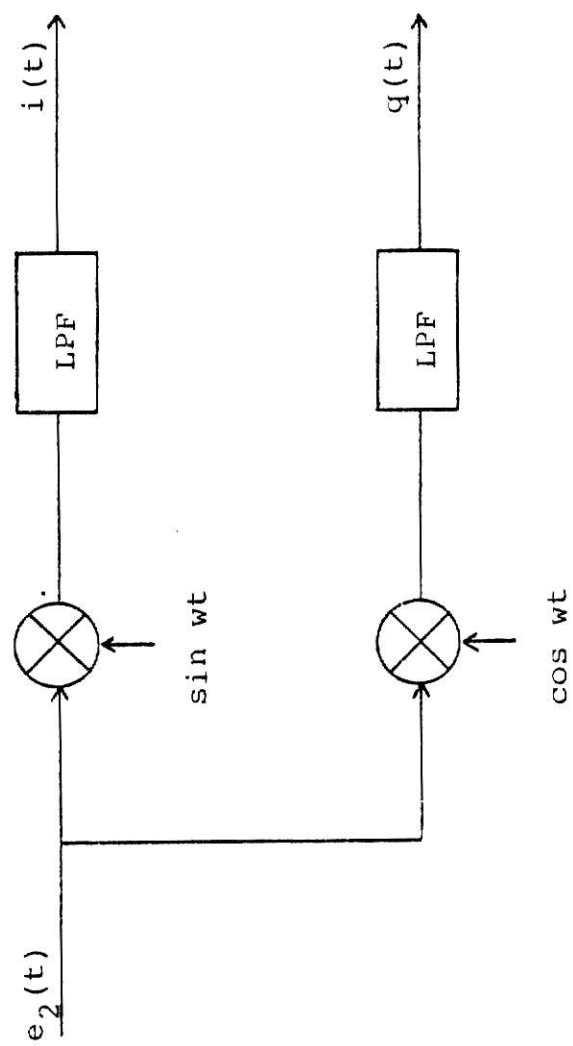


Figure 2. Inphase/Quadrature Demodulation

$$i(t) = e_2(t) \sin wt \Big|_{\text{LPF}}$$

and

(8)

$$q(t) = e_2(t) \cos wt \Big|_{\text{LPF}}$$

where LPF indicates an appropriate low-pass filter operation.

Thus, we have

$$i(t) = \frac{KE}{2} \cos (w_d t)$$

and

(9)

$$q(t) = -\frac{KE}{2} \sin (w_d t)$$

where w_d is positive for increasing and negative for decreasing distance. It should be noted that the addition of the 90° shifted or quadrature channel is to enable determination of direction of motion. Without the Q channel, motion away would appear the same as motion towards the receiver because

$$\cos (\pm w_d t) = \cos (w_d t) \quad (10)$$

With the addition of the Q channel and following the adopted sign convention, motion away from the receiver is represented as

$$i(t) = \cos (+ w_d t) \quad (11a)$$

and

$$q(t) = -\sin (+ w_d t) \quad (11b)$$

and motion towards the receiver is

$$i(t) = \cos (- w_d t) = \cos (w_d t) \quad (11c)$$

and

$$q(t) = -\sin(-w_d t) = +\sin(w_d t) \quad (11d)$$

This is the notion of direction sensitive doppler as introduced by Kalmus [1].

Signal Properties

The class of signals of interest are of the form

$$x(t) = \begin{cases} x_C(t) + x_T(t) \\ x_C(t) \\ x_T(t) \end{cases}$$

where $x_C(t)$ is due to an oscillating source and $x_T(t)$ is the target signal to be detected. The signal $x(t)$ is considered to be a complex signal of the form

$$x(t) = i(t) + jq(t) \quad (12)$$

where

$$i(t) = i_C(t) + i_T(t) \quad (13a)$$

$$q(t) = q_C(t) + q_T(t) \quad (13b)$$

and

$$j = \sqrt{-1}$$

Because the signal $x(t)$ is complex, the frequency spectrum need not be symmetric about the origin. Thus, the concept of positive and negative spectrum components is relevant and will be used extensively throughout.

III. SIGNAL MODELS

Target

The target signal of interest is the result of continuous motion in one direction, toward or away from the receiver. The demodulated I and Q channel components as expressed in (9) are

$$i(t) = \frac{KE}{2} \cos (w_d t)$$

and (14)

$$q(t) = -\frac{KE}{2} \cos (w_d t)$$

where

$$w_d = 2\left(\frac{v}{c}\right)w.$$

These equations are valid for any $v = v(t)$. However, for the time being, we will restrict our attention to the case when the velocity is constant. Therefore, we let

$$v = \hat{V}_T$$

and

$$\hat{w}_T = 2\left(\frac{\hat{V}_T}{c}\right)w.$$

The I and Q components for motion away from the receiver ($+ V_T$) are

$$i(t) = \frac{\hat{KE}}{2} \cos (\hat{w}_T t) \quad (15a)$$

and

$$q(t) = -\frac{\hat{KE}}{2} \sin(\hat{\omega}_T t) \quad (15b)$$

The complex signal $x(t)$ is thus

$$x(t) = \frac{\hat{KE}}{2} \cos \hat{\omega}_T t - j \frac{\hat{KE}}{2} \sin \hat{\omega}_T t \quad (16a)$$

or

$$x(t) = \frac{\hat{KE}}{2} e^{-j \hat{\omega}_T t} \quad (16b)$$

This signal has a single frequency spectrum component at $-\hat{\omega}_T$ as shown in Figure 3. Similarly, for motion toward the receiver ($-\hat{\omega}_T$) the complex signal can be represented as

$$x(t) = \frac{\hat{KE}}{2} e^{j \hat{\omega}_T t} \quad (17)$$

which has a single frequency spectrum component at $+\hat{\omega}_T$ as shown in Figure 3.

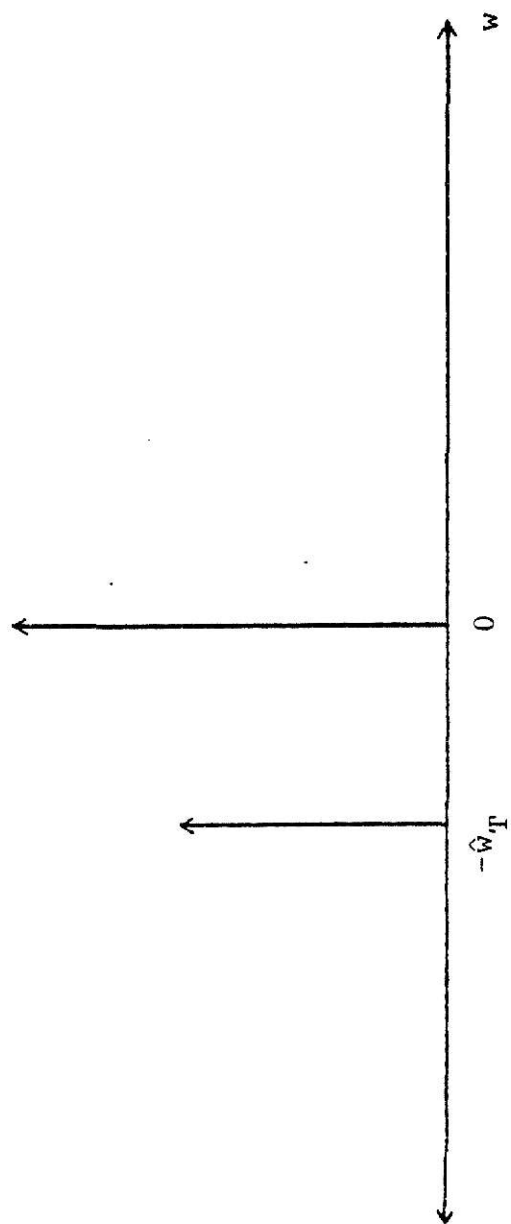
Clutter

The clutter signal is due to the oscillatory motion of one or more objects within the detection region. Because the motion is no longer in just one direction, the return signal contains positive and negative doppler components. The demodulated I and Q channel signals are represented as

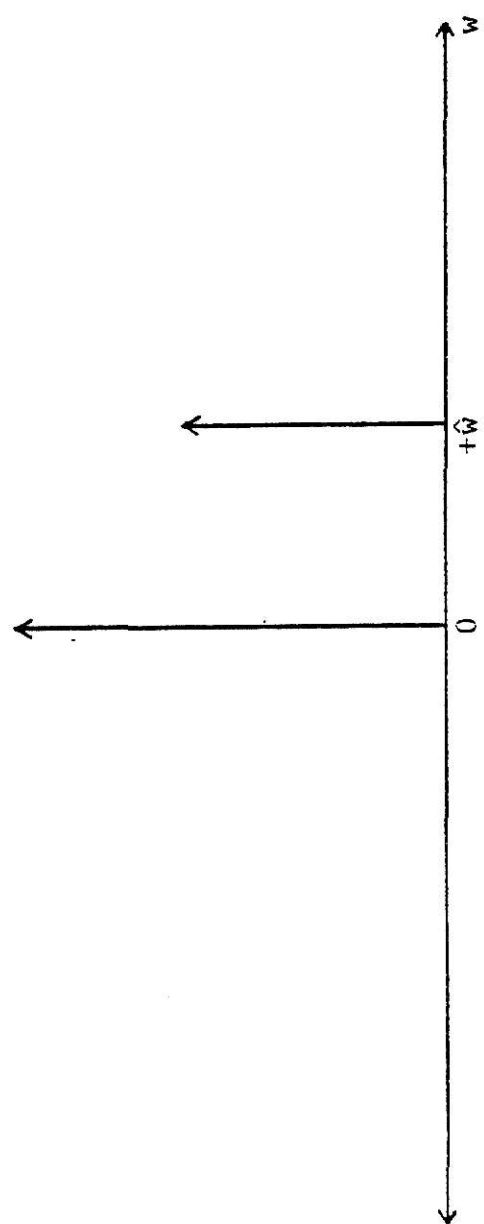
$$i(t) = \frac{KE}{2} \cos [c_1 r(t)] \quad (18a)$$

and

$$q(t) = -\frac{KE}{2} \sin [c_1 r(t)] \quad (18b)$$



a) Power Density Spectrum for Target Motion Away From Receiver



b) Power Density Spectrum for Target Motion Toward Receiver

Figure 3. Target Signal Power Spectra

where c_1 is a conversion constant and $r(t)$ is a time-varying function that represents the motion of the oscillating object. A typical example of $r(t)$ is

$$r(t) = r_1 \sin (wt) \quad (19)$$

Substituting (19) in (18), the demodulated signals become

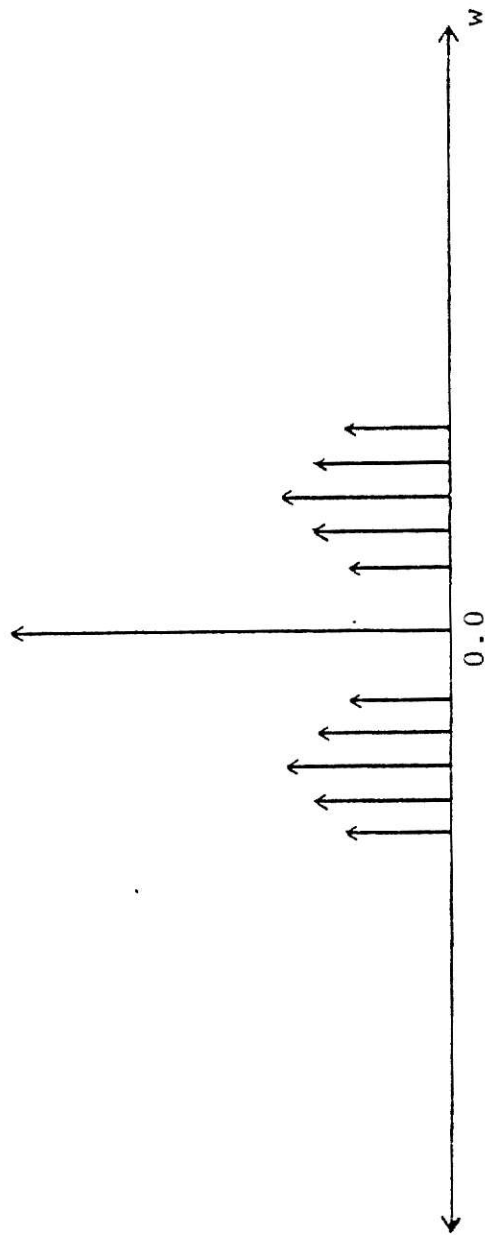
$$i(t) = \frac{KE}{2} \cos \{c_1 r_1 \sin (wt)\} \quad (20a)$$

and

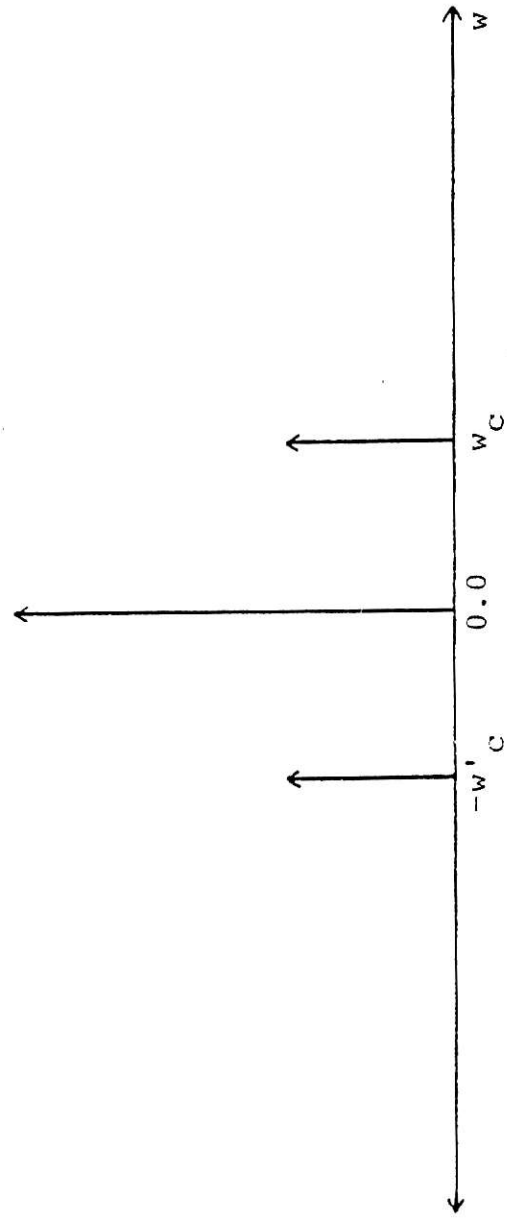
$$q(t) = -\frac{KE}{2} \sin \{c_1 r_1 \sin (wt)\} \quad (20b)$$

which is a frequency modulated (FM) signal. An example of the corresponding two-sided complex frequency spectrum for this class of clutter signals is shown in Figure 4.

For the purposes of the following discussion we assume a hypothetical two-sided clutter signal, whose spectrum contains single components only at $+w_c$ and $-w_c$ as shown in Figure 4. This simplification is made due to the complexities of manipulating equations (20) and allows a more instructive derivation of the detection algorithm. The more complex case of (20) is considered in the section on simulation results.



a) Power Density Spectrum for FM Modulated Clutter Signal



b) Power Density Spectrum for Hypothetical Clutter Signal

Figure 4. Clutter Signal Power Spectra

IV. DETECTION ALGORITHMS

Standard Direction Sensitive Algorithm

A commonly used algorithm suggested by Kalmus [2] for determining whether the signal source is moving toward or away from the radar consists in evaluating the quantity

$$g_s = \int_L \{ i(t) \frac{dq(t)}{dt} \} dt \quad (21)$$

where L denotes an appropriate integration time. The value g_s is compared with a threshold θ such that if

$$-\theta < g_s < \theta \quad (22)$$

no target is present. If the integrated output exceeds the threshold θ , an alarm condition exists. The key is to set the threshold high enough to prevent nuisance alarms due to clutter, but low enough to allow detection of weak target signals.

The integration process suggested in (21) is implemented by an appropriate low-pass operation, and (21) becomes

$$g_s = \left. i(t) \frac{dq(t)}{dt} \right|_{\text{LPF}} \quad (23)$$

Substituting the expression for the demodulated signal (9) in (23) leads to

$$g_s = \left. \left\{ \frac{KE}{2} \cos w_d t \cdot -\frac{KE}{2} w_d \cos w_d t \right\} \right|_{\text{LPF}} \quad (24a)$$

$$\text{or } g_s = -\frac{K^2 E^2}{8} w_d \quad (24b)$$

From (24) it is clear that g_s is negative if w_d is positive -- i.e., when the source moves away from the radar. Conversely, g_s is positive when w_d is negative corresponding to motion towards the radar. Thus, if g_s is integrated (smoothed), the corresponding output for the hypothetical clutter model of Figure 4 is zero because

$$g_s = g_c = \frac{K^2 E^2}{8} w_c - \frac{K^2 E^2}{8} w_c' = 0 \quad (25)$$

Next we consider the case $w_c \neq w_c'$ i.e., when the velocities of the clutter source are different when moving toward and away from the radar. Even under the assumptions of the hypothetical clutter model suggested, the output corresponding to (25) becomes

$$g_s = g_c = \frac{K^2 E^2}{8} (w_c - w_c') \quad (26)$$

where

$$w_c = 2\left(\frac{V_c}{C}\right)w \text{ and } w_c' = 2\left(\frac{V_c'}{C}\right)w$$

The above analysis can be extended to the case when the returned signals of (9) contain an additional term (signal) due to the motion of a target in one direction only. Then corresponding to (25) we obtain the integrator output to be

$$g_s = g_{c+T} = \frac{K^2 E^2}{8} (w_c - w_c') \pm \frac{\hat{K}^2 E^2}{8} \hat{w}_T \quad (27)$$

where

$$\hat{w}_T = 2\left(\frac{\hat{V}_T}{C}\right)w$$

and $\hat{V}_T$ is the velocity of the target. The sign associated with the second term in (27) is positive for an approaching target and negative when it is moving away. This second term provides the information for detection. If the first term of (27) is such that it dominates g_s , the target will be undetectable. Under these conditions, we will be unable to choose a value for the threshold θ such that detection is possible without nuisance alarms.

We will refer to the preceding derivation as the standard detection (SD) algorithm which is shown in Figure 5.

Improved Direction Sensitive Algorithm

In the application of interest, the clutter source and the target are such that:

- a) $w_c \neq w_c'$
- b) $\hat{K} \ll K$; i.e., the radar cross-section (RCS) of the target is much smaller than that of the oscillating body (clutter source);
- c) The target enters the detection region for a relatively short duration, while the clutter is continuously present.

Because of a) and b) above, it is apparent from (27) that the ability to detect targets with the SD algorithm decreases as the difference between w_c and w_c' increases. In such situations adaptive complex prediction can be used effectively to improve the detection performance of the SD algorithm. To this end

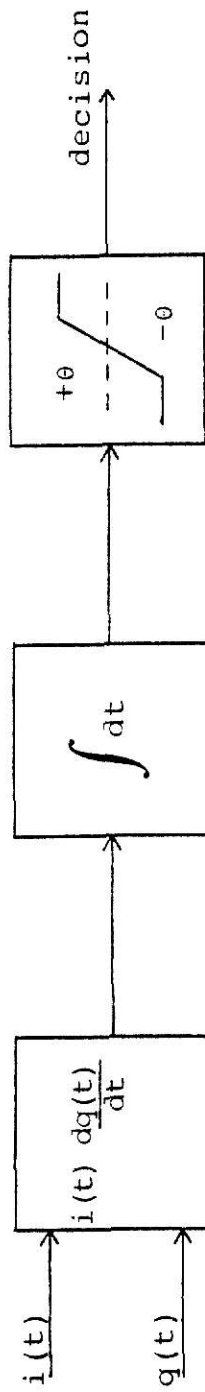


Figure 5. Standard Detection Algorithm

we introduce a complex predictor as illustrated in Figure 6 and refer to the configuration as the modified standard detection (MSD) algorithm. The structure and role of each sub-block of Figure 6 is investigated along with some overall algorithm considerations.

Complex Predictor. The prediction stage in the MSD algorithm is implemented using Widrow's complex least-mean-square (CLMS) algorithm [3]. Referring to Figure 7, the output at time n is

$$e(n) = x(n) - W_m^T X_m \quad (28)$$

where the vectors W_m and X_m are

$$W_m^T = [W_1, W_2, \dots, W_m] \quad (29)$$

and

$$X_m^T = [X(n-\Delta), x(n-\Delta-1), \dots, x(n-\Delta-(m-1))]$$

where m is the number of filter weights and Δ represents a time delay. The filter weights W are updated as [3]

$$W_i(n+1) = W_i(n) + \mu e(n) X^*(n-i) \quad (30)$$

where i represents the i th filter component, μ is a convergence parameter for controlling stability and rate of adaptation, and $X^*(n-1)$ denotes the complex conjugate of $(n-i)$ th sampled input. All operations in (28) and (30) are complex (phasor) operations and the sampled input $X(n)$ is given by (12) as

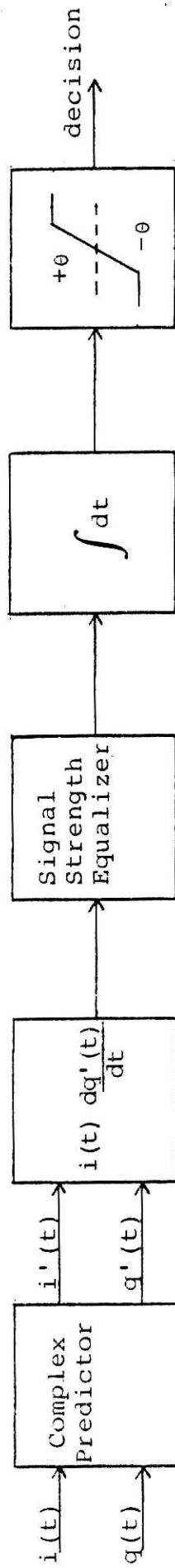


Figure 6. Modified Standard Detection Algorithm

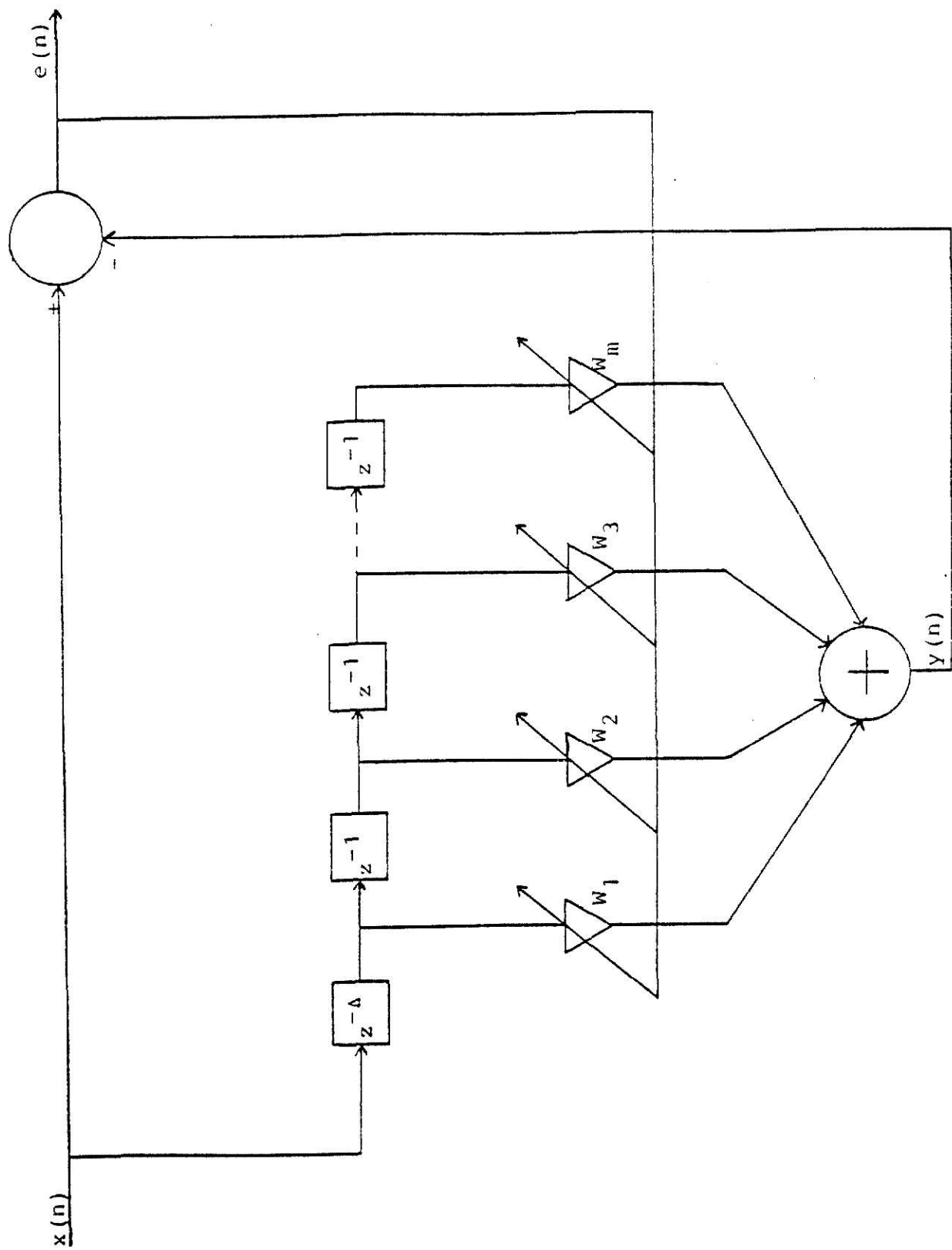


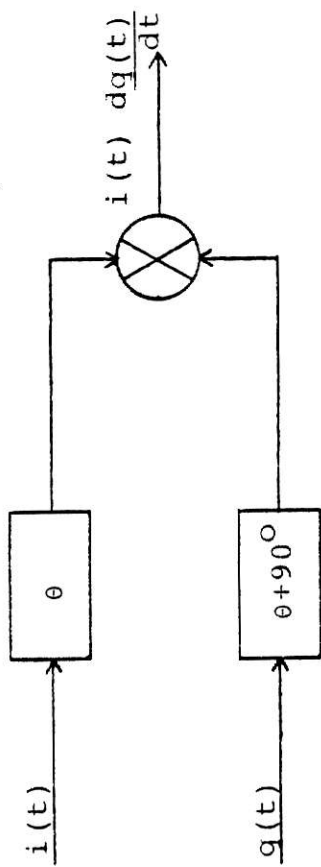
Figure 7. Complex Least Mean Square Adaptive Predictor Block Diagram

$$X(n) = i(n) + jq(n) \quad (31)$$

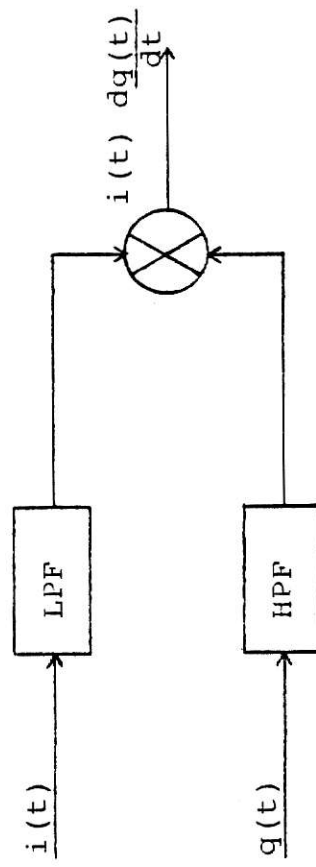
It is interesting to note that (28) and (30) are the general form of the least-mean-square predictor for the real case. A more detailed derivation of (28) and (30) is provided in Appendix A.

The sinusoidal nature of the clutter signal allows for predictability and because the exact frequency is unknown or varies over time, the ability to adapt is desirable. This algorithm is used to remove the larger amplitude clutter and leave the signal (target) portion essentially unchanged by choosing a suitable value for the convergence parameter μ . This allows the CLMS algorithm to adapt fast enough to the large amplitude clutter and leave the signal with little change since the latter is relatively small in amplitude and lasts for a short duration.

Direction Sensing. The $i \frac{dq}{dt}$ process was developed by Kalmus [1], [2] to provide the capability to distinguish between one-sided target signals and two-sided clutter. This is achieved by introducing an additional 90° phase shift to the Q channel and forming the product as shown in Figure 8. This process is implemented by applying a one-pole low-pass filter to the I channel components and a one-pole high-pass filter to the Q channel component as shown in Figure 8. If the low-pass and high-pass filters have the same cutoff, the phase response of the two differ by exactly 90° over all frequencies. The



a) Kalmus Processor



b) Filter Realization

Figure 8. Direction Sensing Algorithm

transfer characteristics for both filters are shown in Figure 9. The product formed after this process yields the output suggested by (24), that is

$$\frac{K^2 A^2}{4} \cos^2 (w_d t) \quad (32)$$

This response leads to zero dc offset for ideal clutter and positive or negative dc offset for an approaching or receding target. The transfer functions for the low-pass and high-pass filters used are given in Appendix B.

Signal Strength Equalizer. The signal strength equalizer (SSE) stage of the MSD algorithm is employed to reduce the dependency of the threshold θ on the gain contributions of the previous stages and the input signal. It is an implementation of a first-order difference equation which continuously scales its input so that its output signal strength is maintained at same prescribed level. Thus, if X_n is the input sample to the SSE at time n , the corresponding output is

$$y_n = \left(\frac{d}{S_n}\right) X_n \quad (33)$$

where S_n is the signal strength estimate which is obtained as

$$S_n = \beta S_{n-1} + (1-\beta) |X_n| \quad (34)$$

where $0 < \beta < 1$ is a smoothing parameter. The implementation of the SSE is shown in Figure 10.

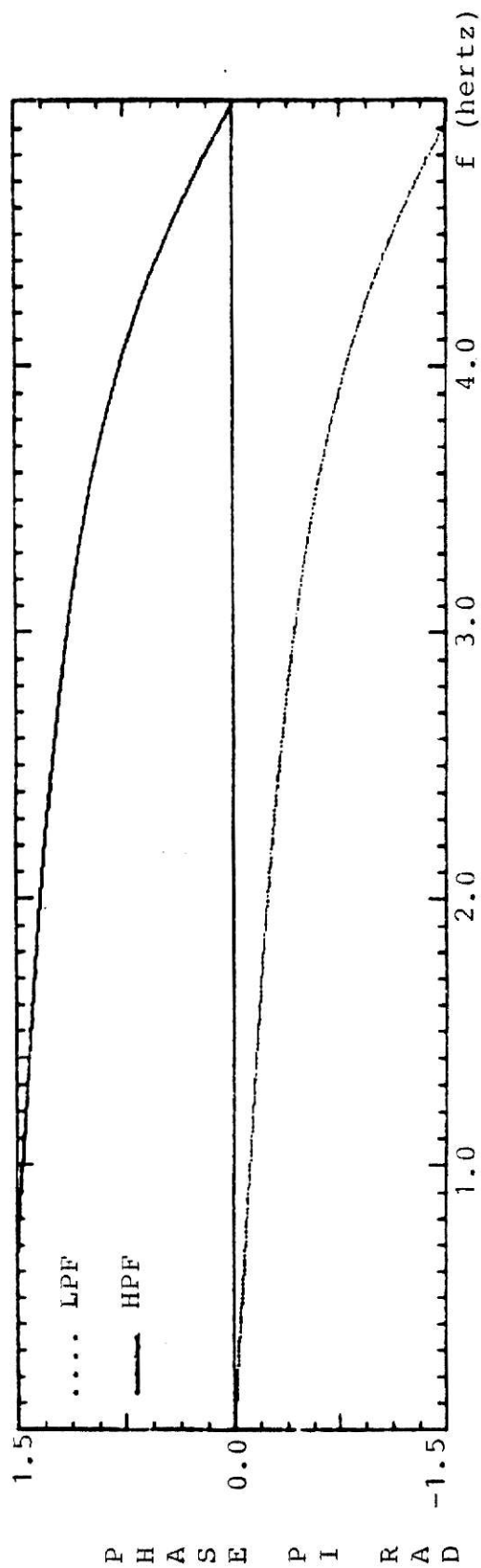
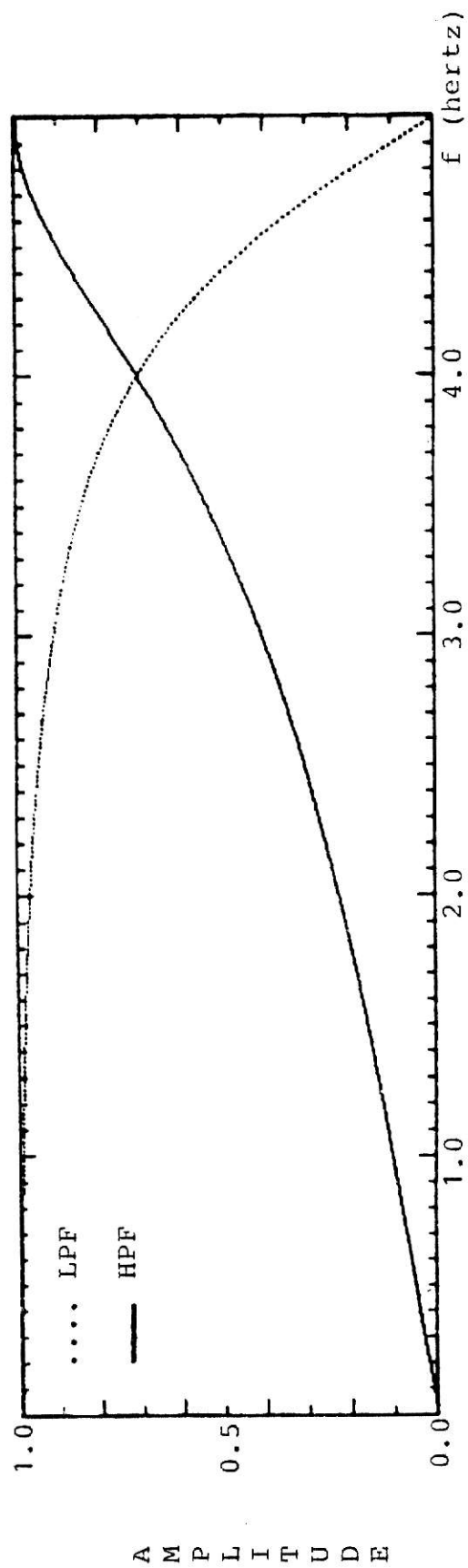


Figure 9. Characteristics for Filter Realization of Kalms Processor

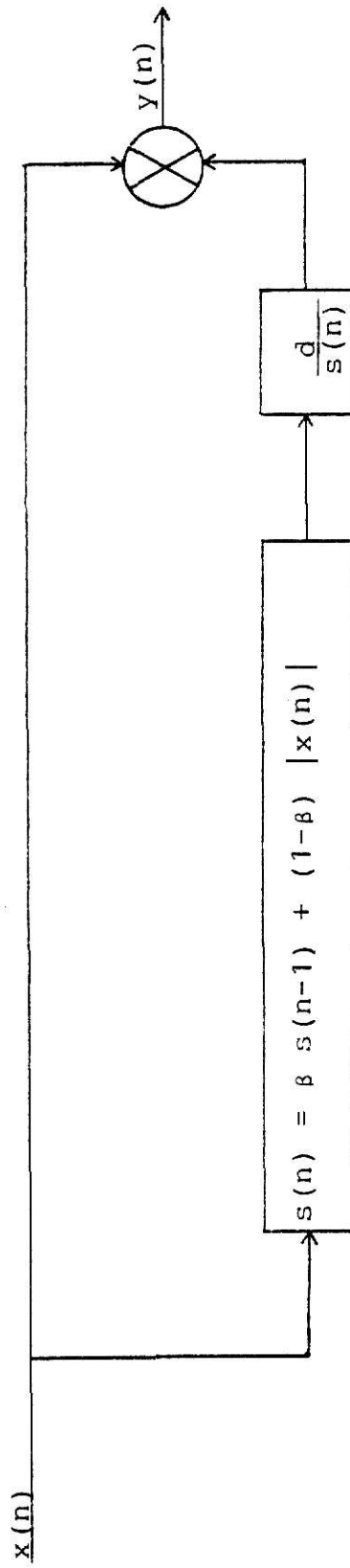


Figure 10. Signal Strength Equalizer Block Diagram

Integration and Threshold Detection. The integration process of the MSD algorithm of Figure 6 is performed by an application of a suitable low-pass filter operation which provides proper smoothing. The effect is to integrate out the positive and negative contributions of the clutter while allowing the target portion to build to a detectable level.

The threshold detection stage simply compares the integrator output with a preset threshold value $\pm \theta$ such that if the threshold is exceeded, a target is present.

V. ROLE OF THE CLMS ALGORITHM

A Simplified Model

The role of the CLMS algorithm is best explained using the hypothetical clutter source that produces a two-sided spectrum with single frequency spectrum components at $+w_c$ and $-w'_c$; see Figure 4. The following cases are considered for the clutter and target signals:

- a) $w_c \neq w'_c$ i.e., approaching and regressing velocities of clutter source are different.
- b) The radar cross section (RCS) of the clutter signal changes due to rotation; i.e., $K \neq K'$ in (27).

The final outputs of the MSD algorithm are compared with the SD algorithm of Figure 5 for clutter only conditions and for the case where the target signals of Figure 11 are added to the clutter signals. The target signal motion begins at 90 seconds and lasts for 30 seconds with a peak amplitude of 0.1 and a frequency of 1.0 hertz. If we let the demodulated signals (9) be

$$i(t) = A \cos w_d t$$

and (35)

$$q(t) = -A \sin w_d t,$$

for motion towards the receiver (35) becomes

$$i(t) = A \cos w_c t$$

and (36)

$$q(t) = A \sin w_c t$$

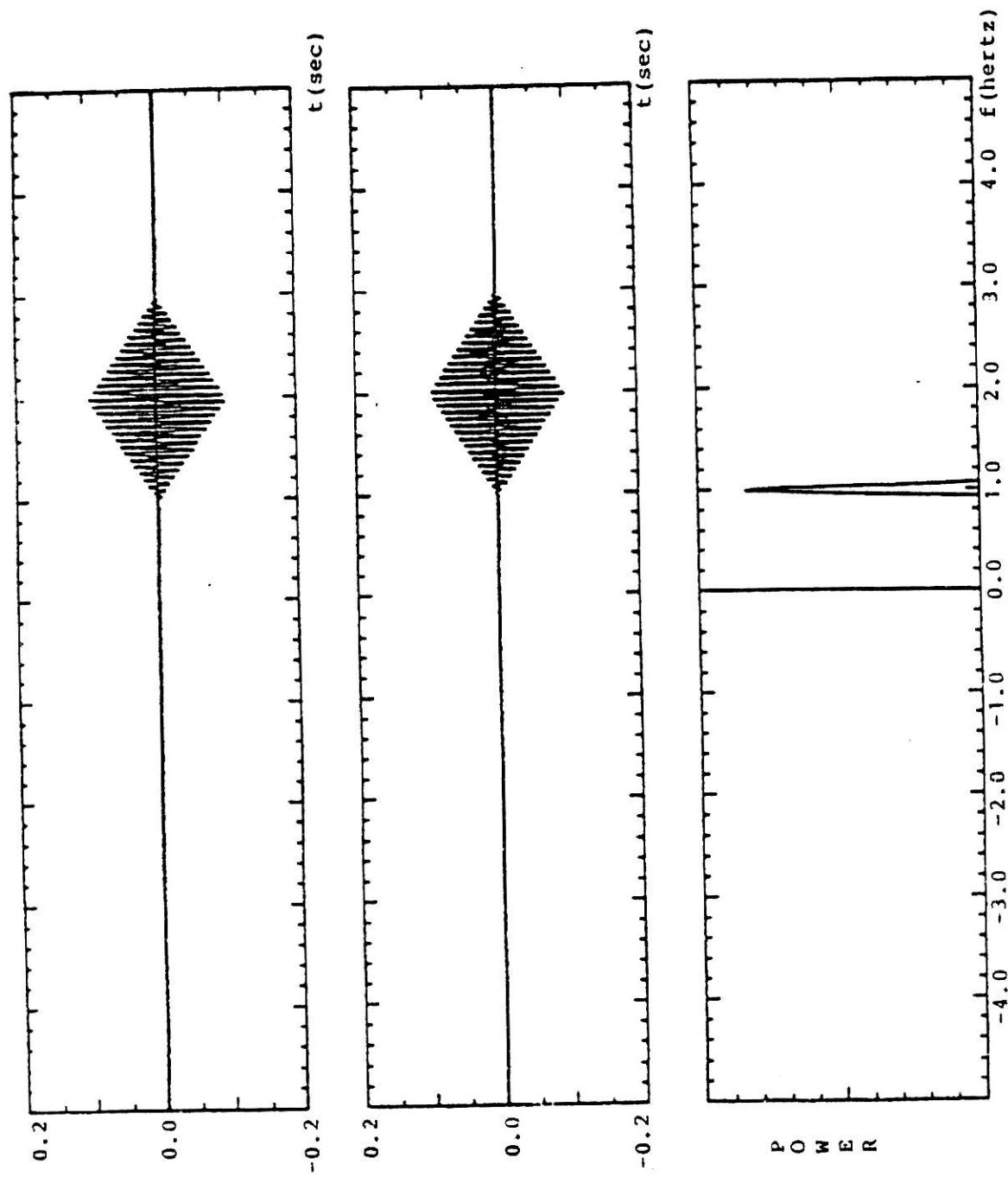


Figure 11. Triangular Target Signal with One-Sided Power Density Spectrum (PDS)

and for motion away, (35) becomes

$$i(t) = A' \cos w'_c t$$

and

(37)

$$q(t) = -A' \sin w'_c t.$$

The resulting signals due to both sources are

$$i(t) = A \cos w_c t + A' \cos w'_c t$$

and

(38)

$$q(t) = A \sin w_c t - A' \sin w'_c t$$

with appropriate choices of A , A' , w_c and w'_c , the conditions of (a) and (b) mentioned previously can be mathematically modeled. The simulations were performed on a Data General NOVA 4X minicomputer. The signal parameters of interest along with peak integrator output values are given in Table 1 with case 1 corresponding to condition (a) where $w_c \neq w'_c$, case 2 to condition (b) where $A \neq A'$, and case 3 to conditions (a) and (b) where $w_c \neq w'_c$ and $A \neq A'$.

The plots of Figures 12-14 correspond to the signals at various stages of the algorithms as labeled. The resulting final outputs of the integration stage of the SD algorithm are consistent with the results indicated in (27). Even for the hypothetical clutter case, the SD algorithm is unable to detect a weak target signal for the conditions when the oscillating source has different velocities for approaching and receding motion and/or when the RCS varies. The integrator

final outputs of the SD algorithm are slightly larger for the clutter plus target case than for the corresponding clutter only condition, but no common threshold θ exists such that detection is possible for all cases without false alarms.

The MSD algorithm however, provides reliable detection for all cases of the simplified model. The plots containing the CLMS predictor outputs show that the CLMS did in fact reduce the clutter amplitude and allowed the target signal to pass relatively unaffected. The CLMS predictor stage of the MSD algorithm used 8 filter weights with a convergence parameter μ of 0.01. Detection is indicated by clipping of the integrator outputs for the clutter plus target conditions in Figures 12c, 13c, and 14c for the MSD algorithm. The value of the threshold θ was chosen at ± 0.01 which allows for reliable detection with no false alarms for hypothetical clutter model simulations.

		SIGNAL PARAMETERS					MAXIMUM INTEGRATOR OUTPUTS			
							CLUTTER ONLY			CLUTTER + TARGET
							SD ALGORITHM	MSD ALGORITHM	SD ALGORITHM	
CASE 1	2.0	2.0	2.75	2.5	2.5	-33.8	0.1954	6.02 E-6	0.1977	3.62 E-2
CASE 2	1.0	2.0	2.5	2.5	2.5	-31.8	2.3997	3.65 E-5	2.4000	3.33 E-2
CASE 3	1.0	2.0	2.5	2.75	2.75	-31.8	1.4249	5.78 E-5	1.4300	3.38 E-2

$$*SNR_{dB} = 10 \log \frac{\sigma_T^2}{\sigma_C^2}$$

σ_T^2 = sum of variances of I and Q channel target signals

σ_C^2 = sum of variances of I and Q channel clutter signals

Table 1. Hypothetical Clutter Model

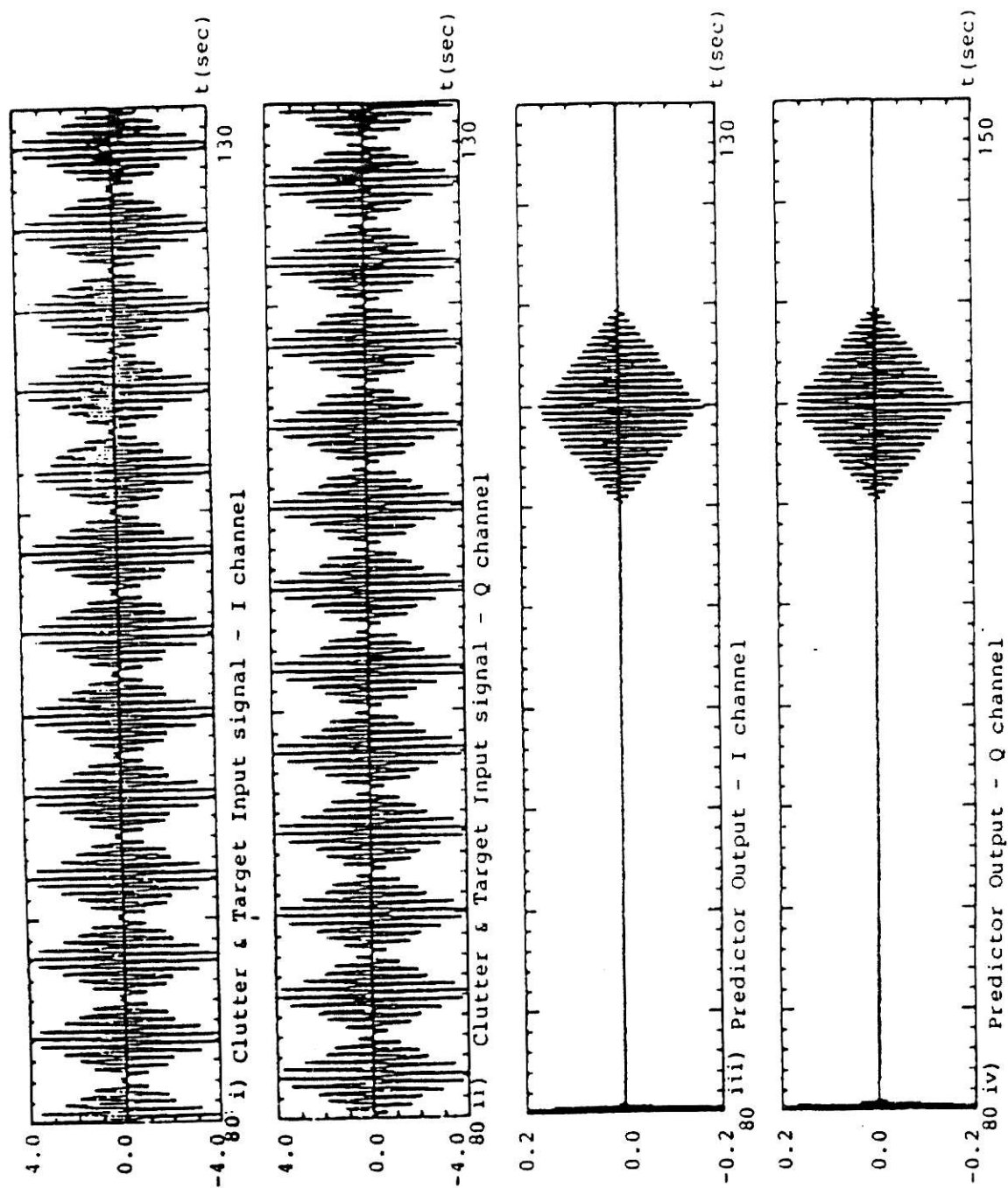
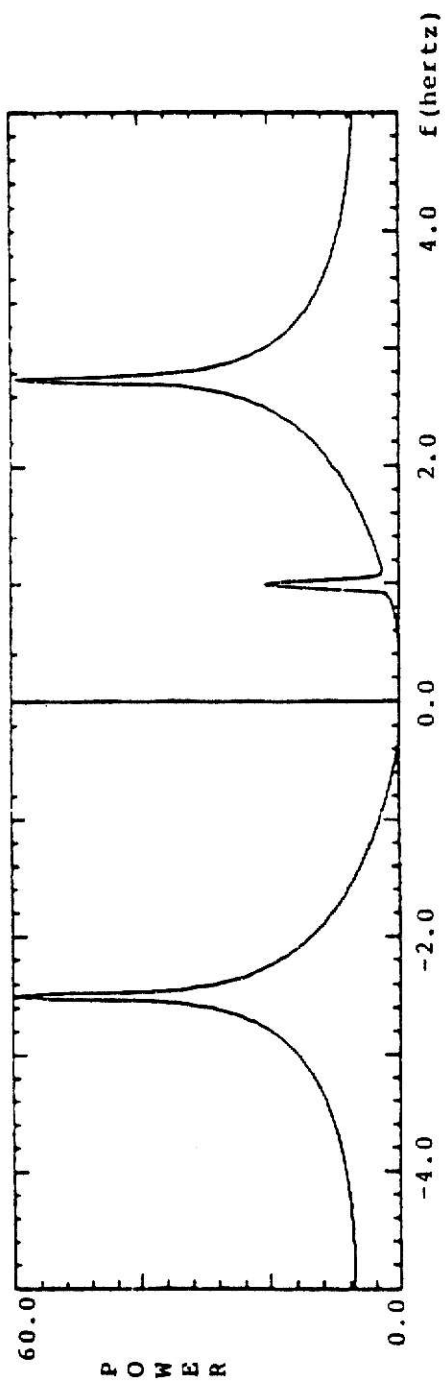
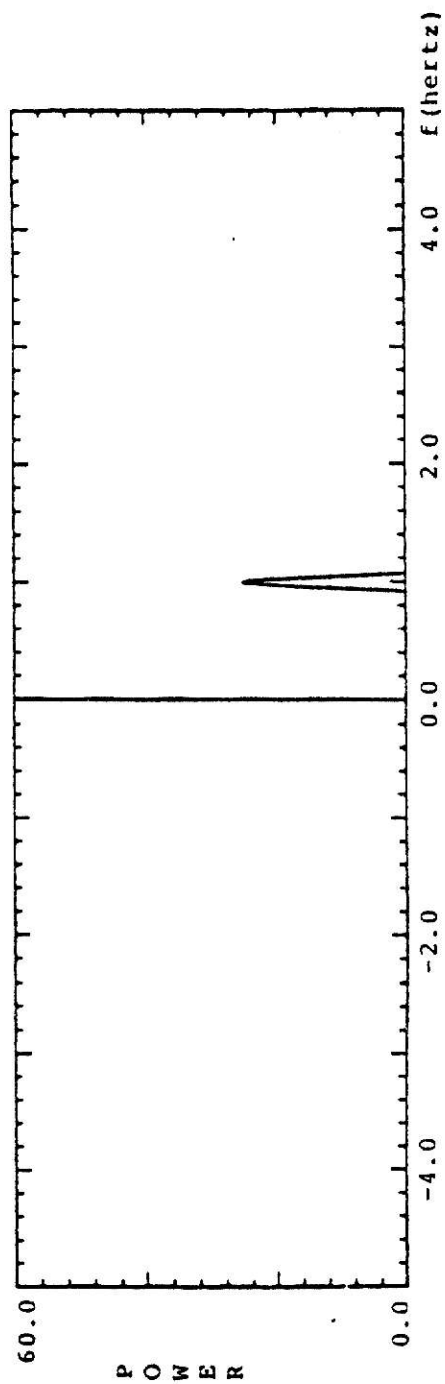


Figure 12a. Hypothetical Clutter Model (HCM); Case 1



i) PDS of Input Clutter & Target



ii) PDS of Predictor Output
Figure 12b. HCM; Case 1

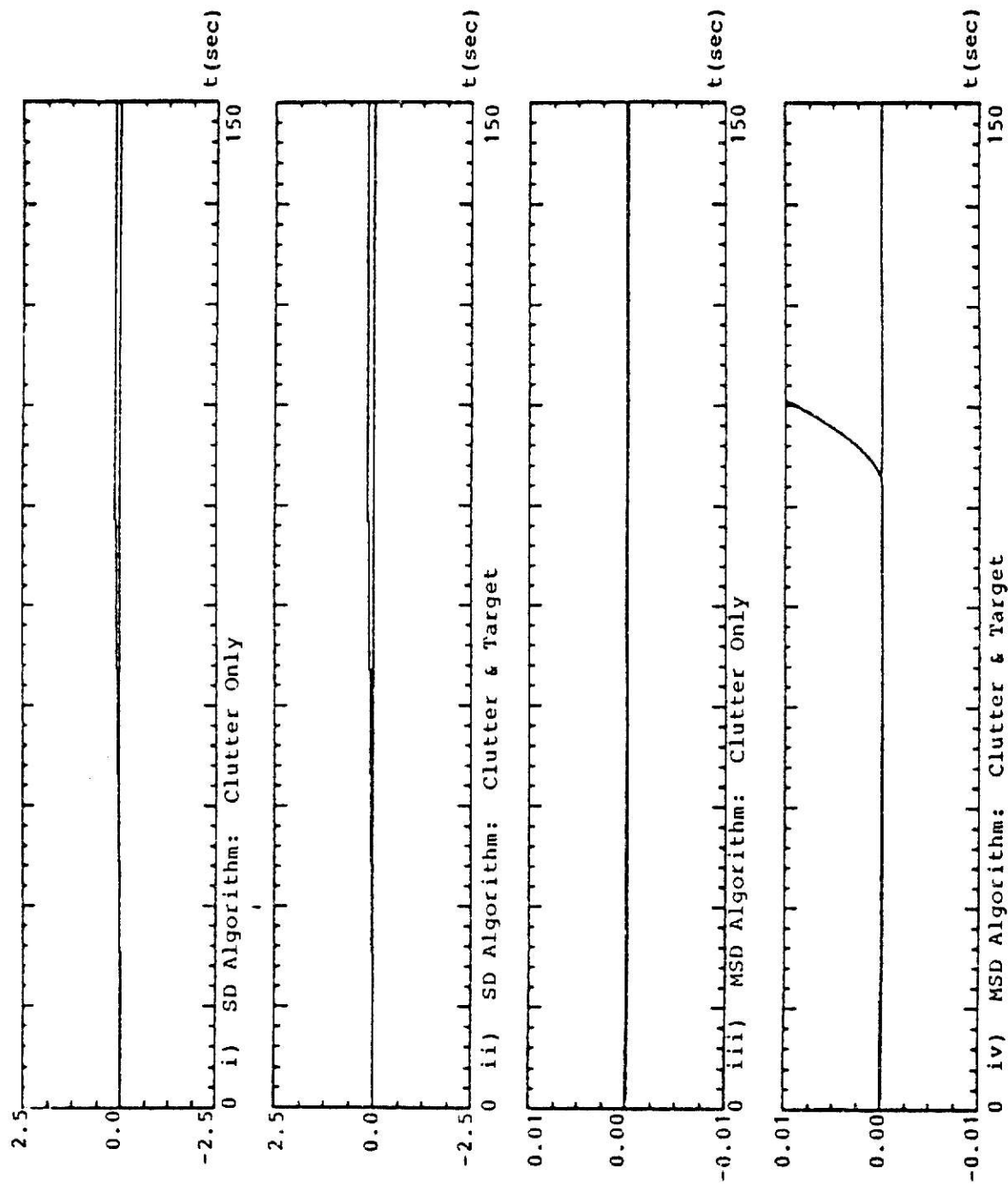


Figure 12c. HCM Integrator Outputs; Case 1

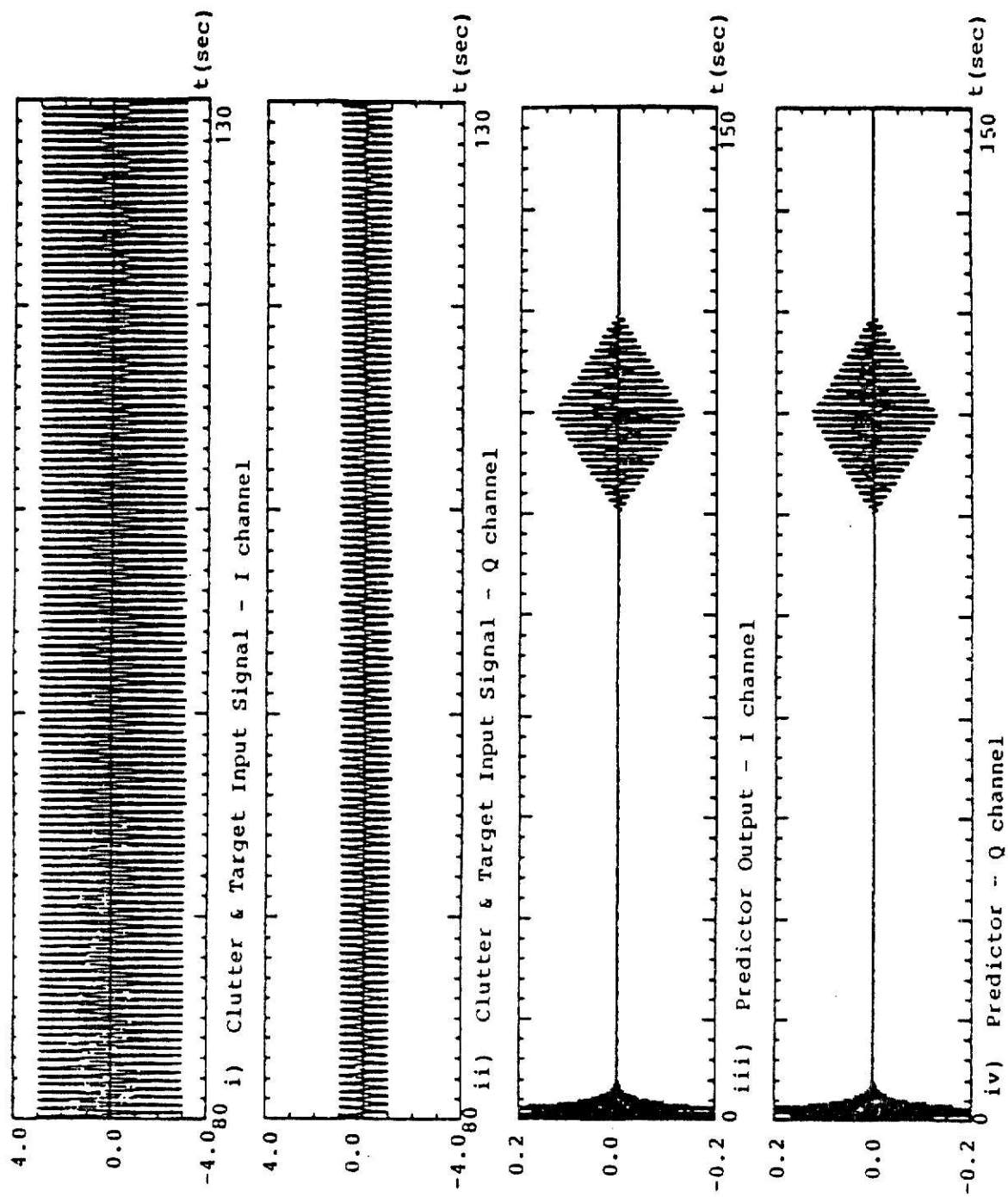
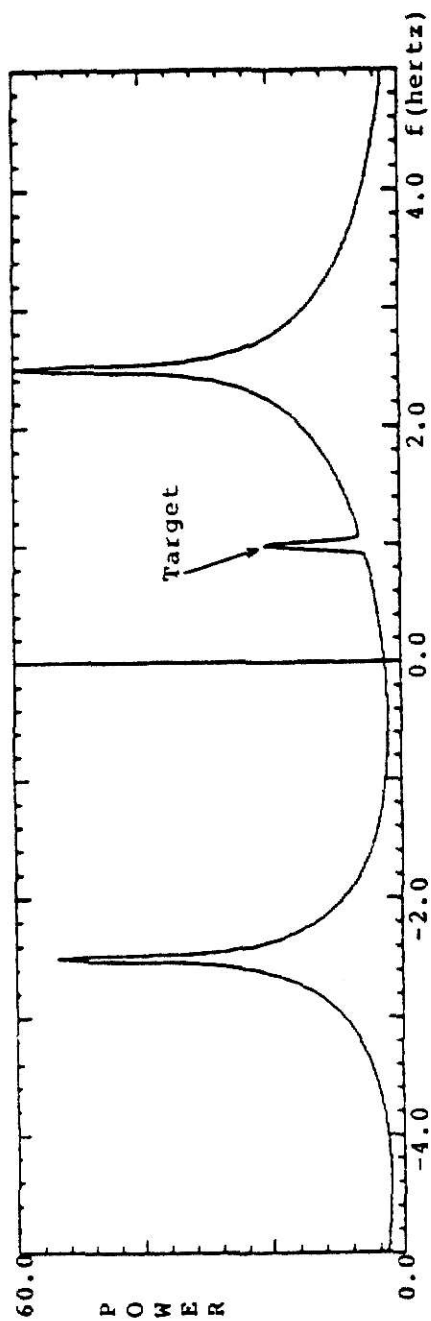
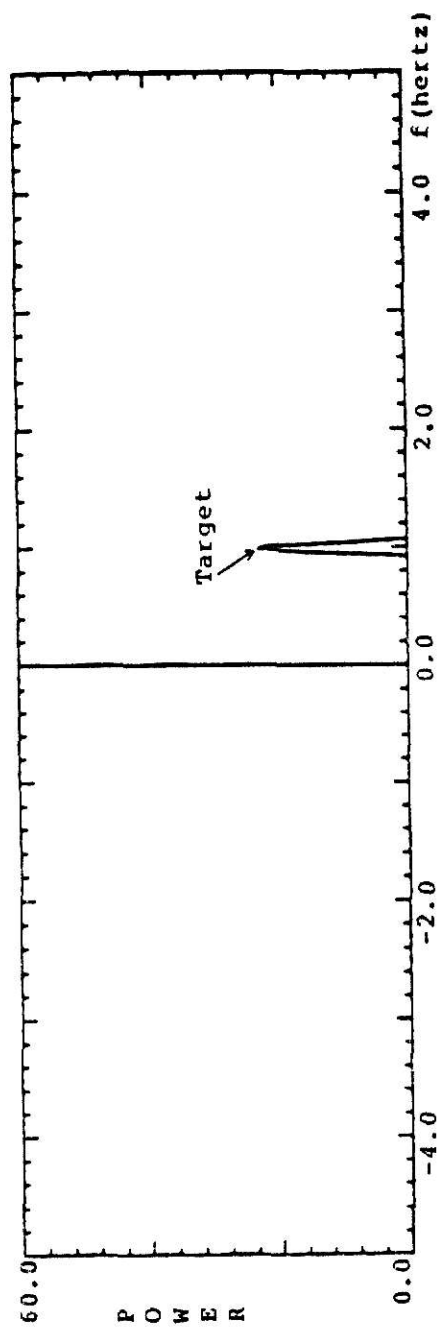


Figure 13a. HCM; Case 2



i) PDS of Input Clutter & Target



ii) PDS of Predictor Output

Figure 13b. HCM; Case 2

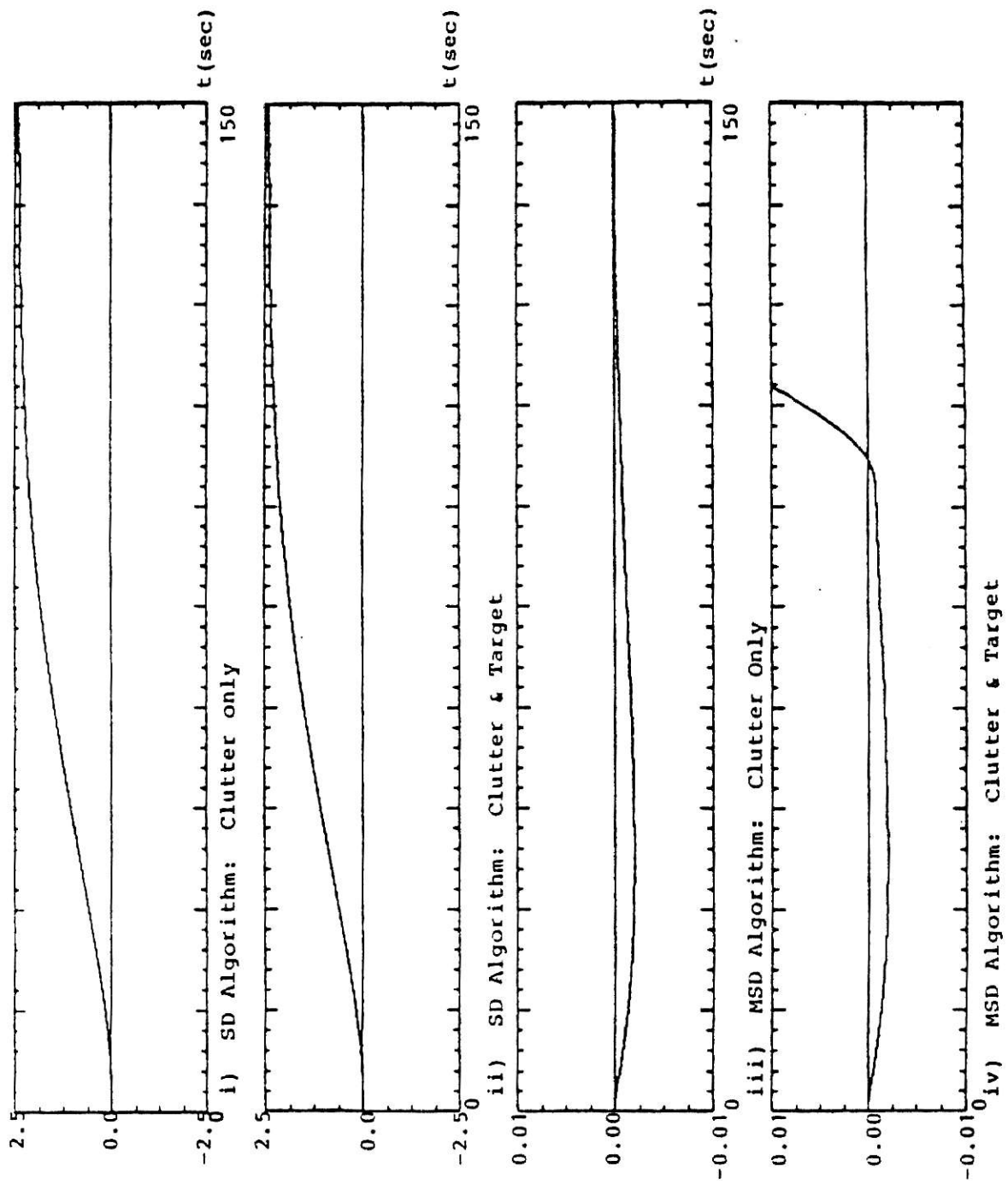


Figure 13c. HCM Integrator Outputs; Case 2

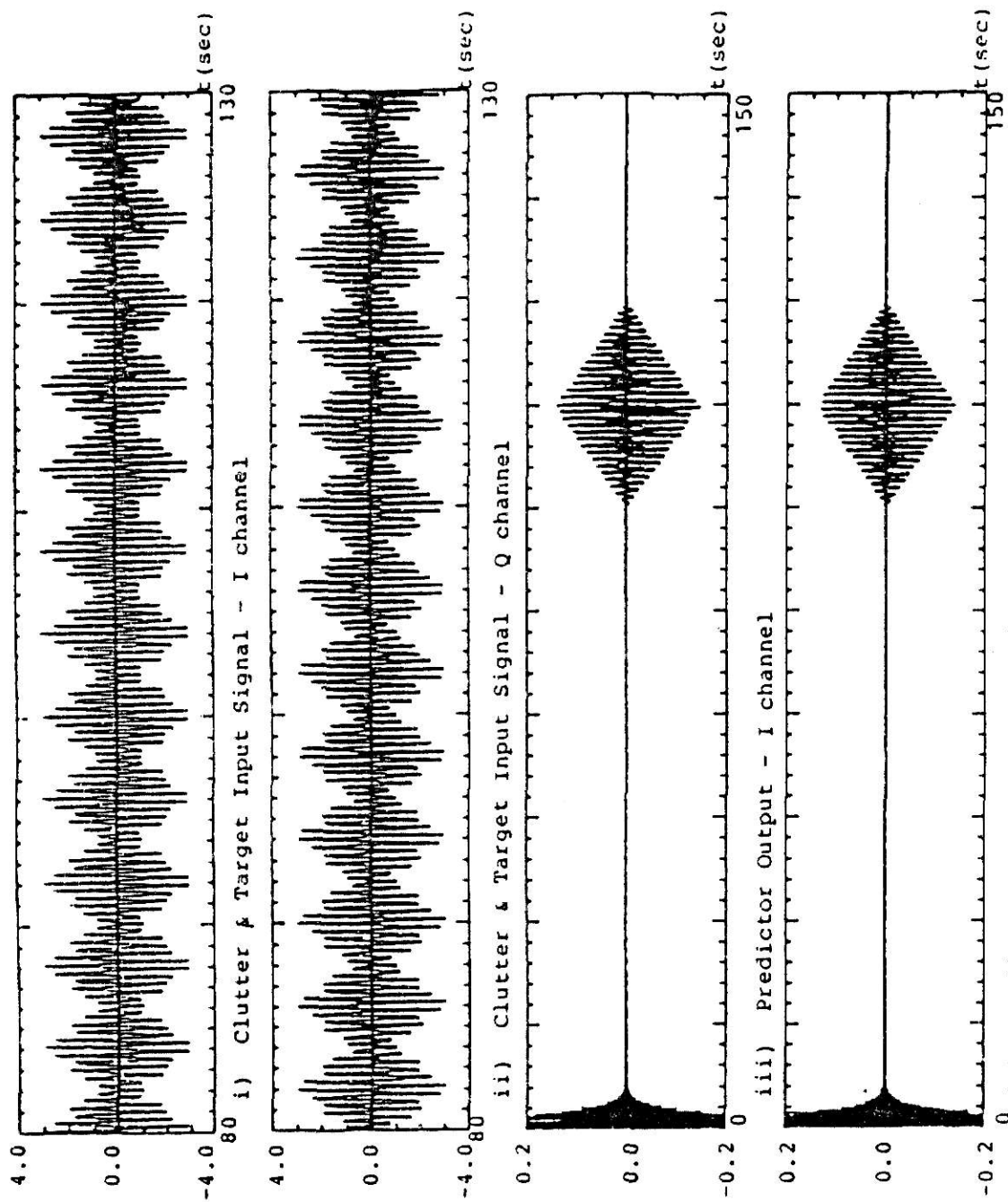


Figure 14a. HCM; Case 3

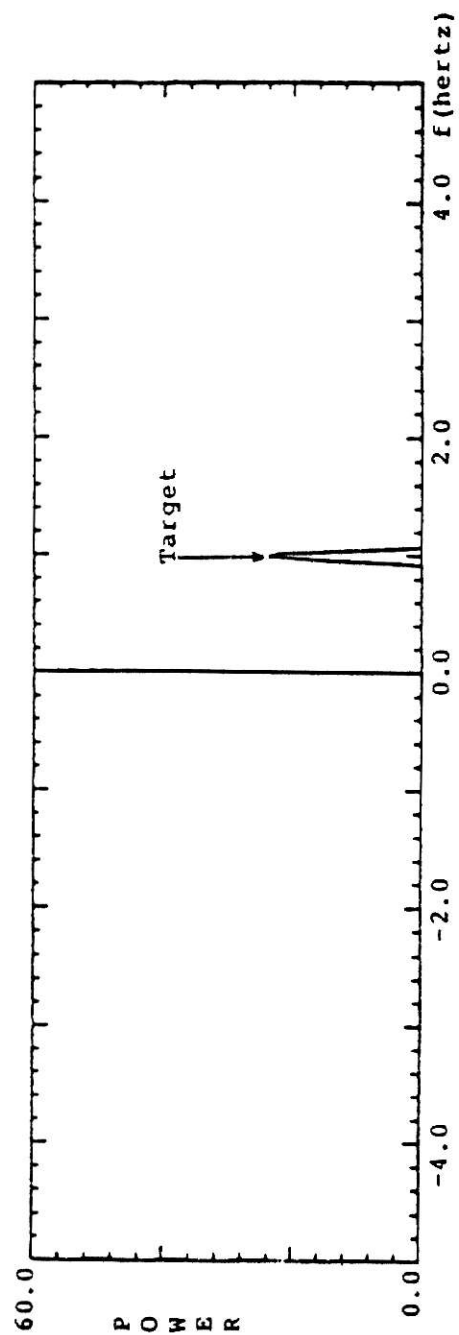
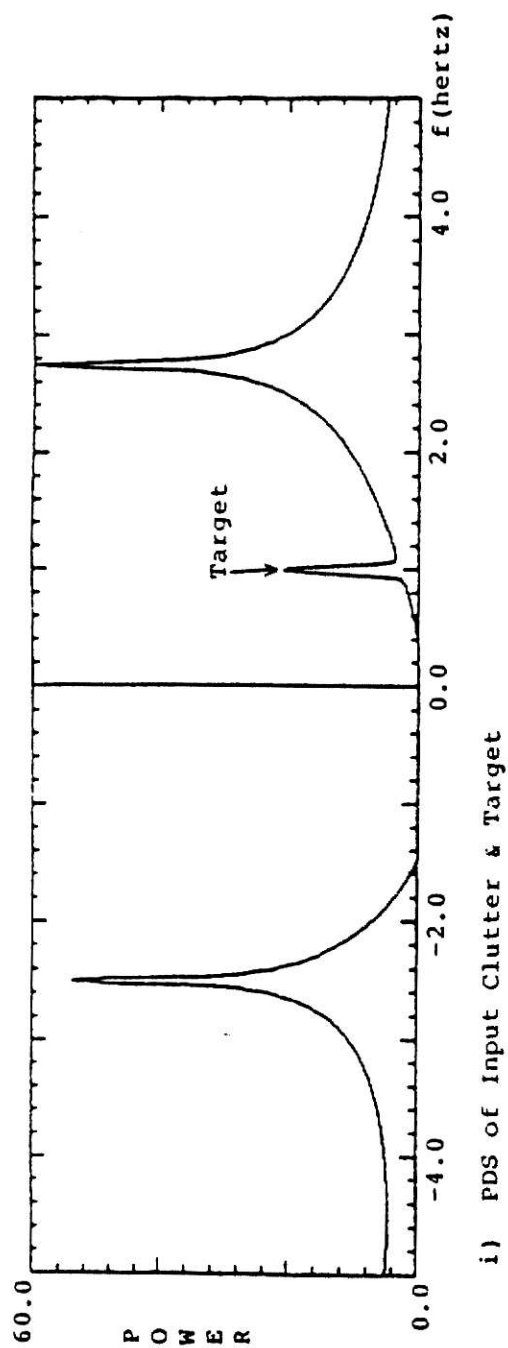


Figure 14b. HCN; Case 3

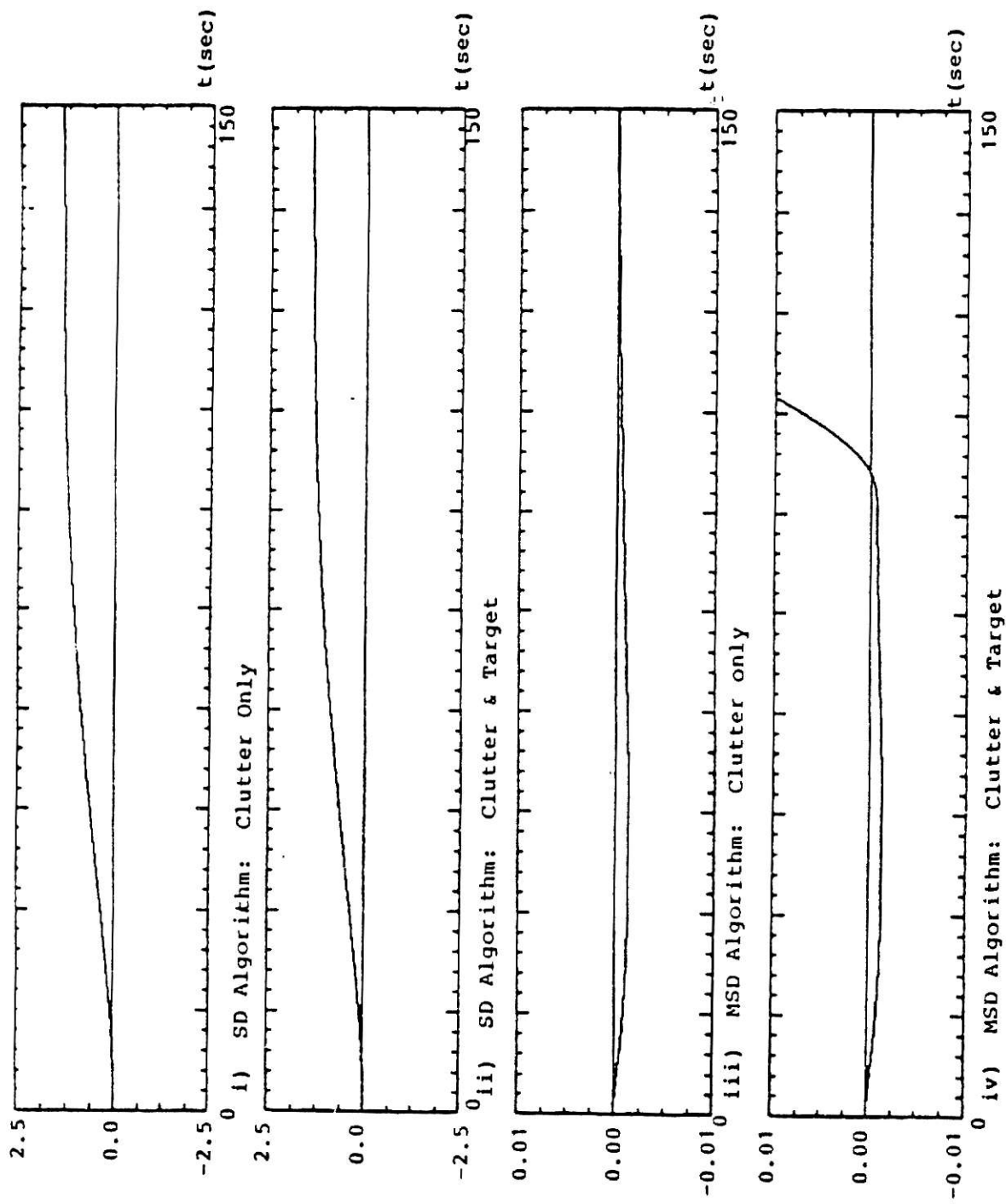


Figure 14c. HCM Integrator Output; Case 3

VI. SIMULATION RESULTS

We now consider a general class of demodulated clutter signals of the form

$$i(t) = A \cos \{ \theta_0 + [4 \pi / \lambda r(t)] \}$$

and (39)

$$q(t) = A \sin \{ \theta_0 + [4 \pi / \lambda r(t)] \}$$

where

A is the clutter amplitude

θ_0 is the clutter signal phase at its
quiescent position

$r(t)$ is the amplitude of the mechanical motion

λ is the radar wavelength.

The clutter signals of (39) have a two-sided frequency spectrum and by appropriate choices of $r(t)$, can be used to model oscillating objects of a physical system. The function $r(t)$ represents the position of the clutter source as a function of time t where the rate of change of $r(t)$ is the velocity.

The first choice of $r(t)$ is of the form

$$r(t) = r_1 \sin (2\pi f t) \quad (38)$$

where r_1 is the amplitude and f is the frequency of mechanical motion of the oscillating clutter source. Substitution of (38) in (37) yields the I and Q channel components

$$i(t) = A \cos \{ \theta_0 + [4\pi r_1 / \lambda \sin (2\pi f t)] \}$$

and (39)

$$q(t) = A \sin \{ \theta_0 + [4\pi r_1 / \lambda \sin (2\pi f t)] \}$$

The signal parameters and algorithm outputs corresponding to the signals of (39) are referred to as case 1.

The next choice of $r(t)$ is shown in Figure 15. The choices of rise-time (T_R) and fall-time (T_F) allow simulation of changes in velocity for motion towards and away and changing signal amplitude A corresponding to motion towards or away results in simulation of a change in the RCS due to rotation. Three cases using the $r(t)$ function of Figure 15 substituted in (37) are simulated and referred to as cases 2, 3, and 4. Case 2 corresponds to oscillation in which velocity in both directions is the same with the same RCS. Case 3 corresponds to a greater velocity for motion towards the receiver than for motion away with equal RCS. Case 4 corresponds to a greater RCS for approaching motion than for motion away with equal velocities in both directions.

The clutter signal parameters referring to (37) for each case are given in Table 2 along with the signal-noise-ratio (SNR) computed as

$$SNR_{dB} = 10 \log \frac{\sigma_T^2}{\sigma_c^2} \quad (40)$$

where

$$\sigma_T^2 = \sigma_{I_T}^2 + \sigma_{Q_T}^2$$

and

$$\sigma_c^2 = \sigma_{I_C}^2 + \sigma_{Q_C}^2 \quad (41)$$

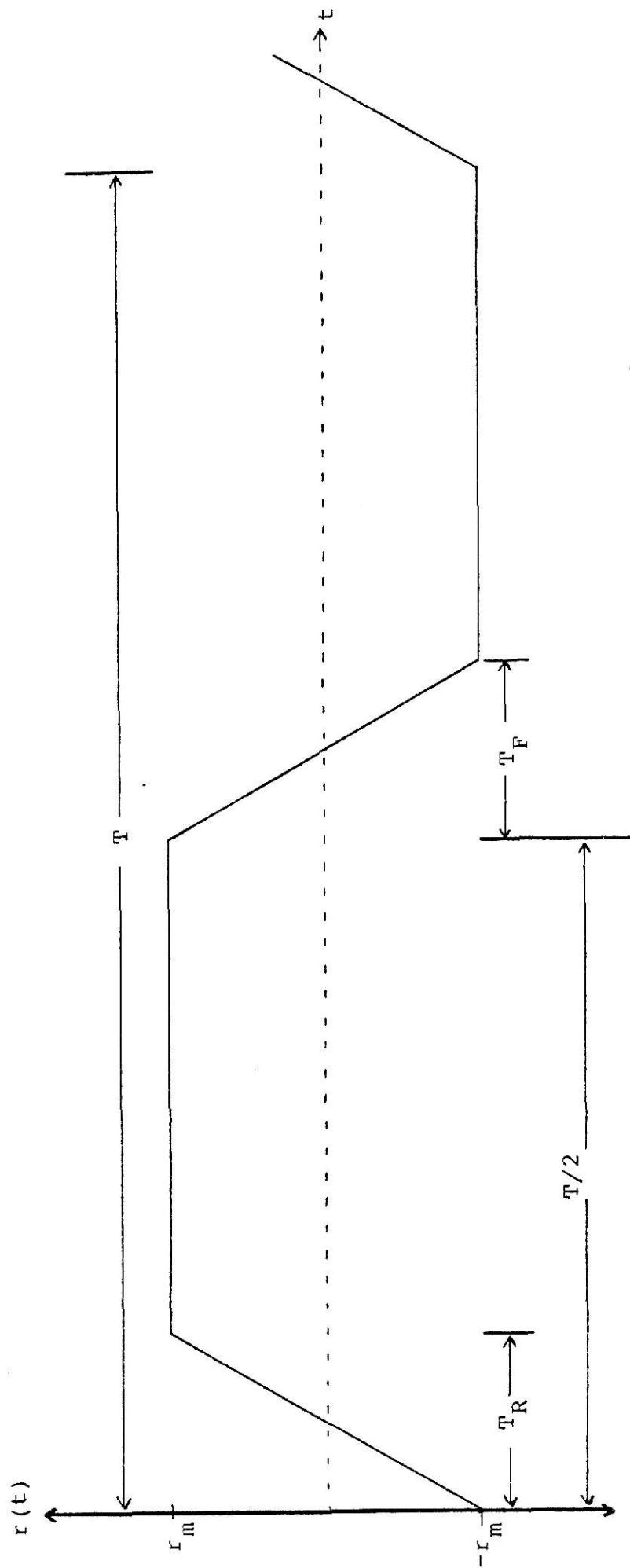


Figure 15. $r(t)$ Motion Function for Clutter Simulation

where $\sigma_{I_T}^2$ represents the variance of the I channel target component. The target signal of Figure 11 is used with motion beginning at 360 seconds and lasting for 40 seconds with a peak amplitude of 0.1.

The plots of Figures 22-33 correspond to the outputs of various stages of the MSD and SD algorithms as indicated. The integrator outputs for the clutter only cases are shown with the corresponding outputs of the clutter plus target signals. The clutter only and clutter plus target input signals were band-pass filtered with a 2-section Butterworth filter from 0.3 to 3 hertz to remove the d.c. component of the input signal.

Interpretation of Results. The SD final integrator outputs of Figures 16, 17, 18, and 19 and the peak integrator outputs of Table 2 show that no common threshold θ can be chosen which allows detection in all cases without nuisance alarms. The MSD algorithm however, allows for detection in all cases. The apparent nuisance alarm in Figure 30 for the MSD algorithm can be attributed to system adaptation time as the output moves to steady state after one cycle of the input. The plots of the power spectra of the input signals and outputs of the CLMS predictor show that clutter power was reduced with little attenuation of the target signal which allows for more reliable detection as shown in the final integrator outputs.

The predictor stage of the MSD algorithm used 16 filter weights with a convergence parameter $\mu = 0.001$. The computer

programs used for each stage of the MSD and SD algorithm are given in Appendix C along with the parameters required for computer simulation.

	SIGNAL PARAMETERS							MAXIMUM INTEGRATOR OUTPUTS			
	T_R (sec)	T_F (sec)	T (sec)	r_m	A	A'	SNR dB	CLUTTER ONLY		CLUTTER & TARGET	
								SD ALG.	MSD ALG.	SD ALG.	MSD ALG.
CASE 1	---	---	1.667	0.125 λ	5.0	5.0	-37.6	1.53 E-2 -9.26 E-4	1.84 E-3 -1.42 E-3	1.53 E-2 -9.26 E-4	1.23 E-2 -1.42 E-3
CASE 2	1.0	1.0	60.0	0.125 λ	5.0	5.0	-21.6	4.74 E-2 -3.74 E-2	3.60 E-3 -2.08 E-3	4.74 E-2 -3.74 E-2	1.92 E-2 -2.74 E-3
CASE 3	1.0	2.0	60.0	0.125 λ	5.0	5.0	-20.6	5.86 E-2 -7.32 E-3	3.58 E-3 -1.27 E-2	6.25 E-2 -7.32 E-3	1.74 E-2 -1.27 E-2
CASE 4	1.0	1.0	60.0	0.125 λ	5.0	4.0	-20.8	4.75 E-2 -2.11 E-2	3.60 E-3 -1.89 E-3	4.75 E-2 -2.11 E-2	2.41 E-2 -1.89 E-3

Table 2. Simulation Clutter Model

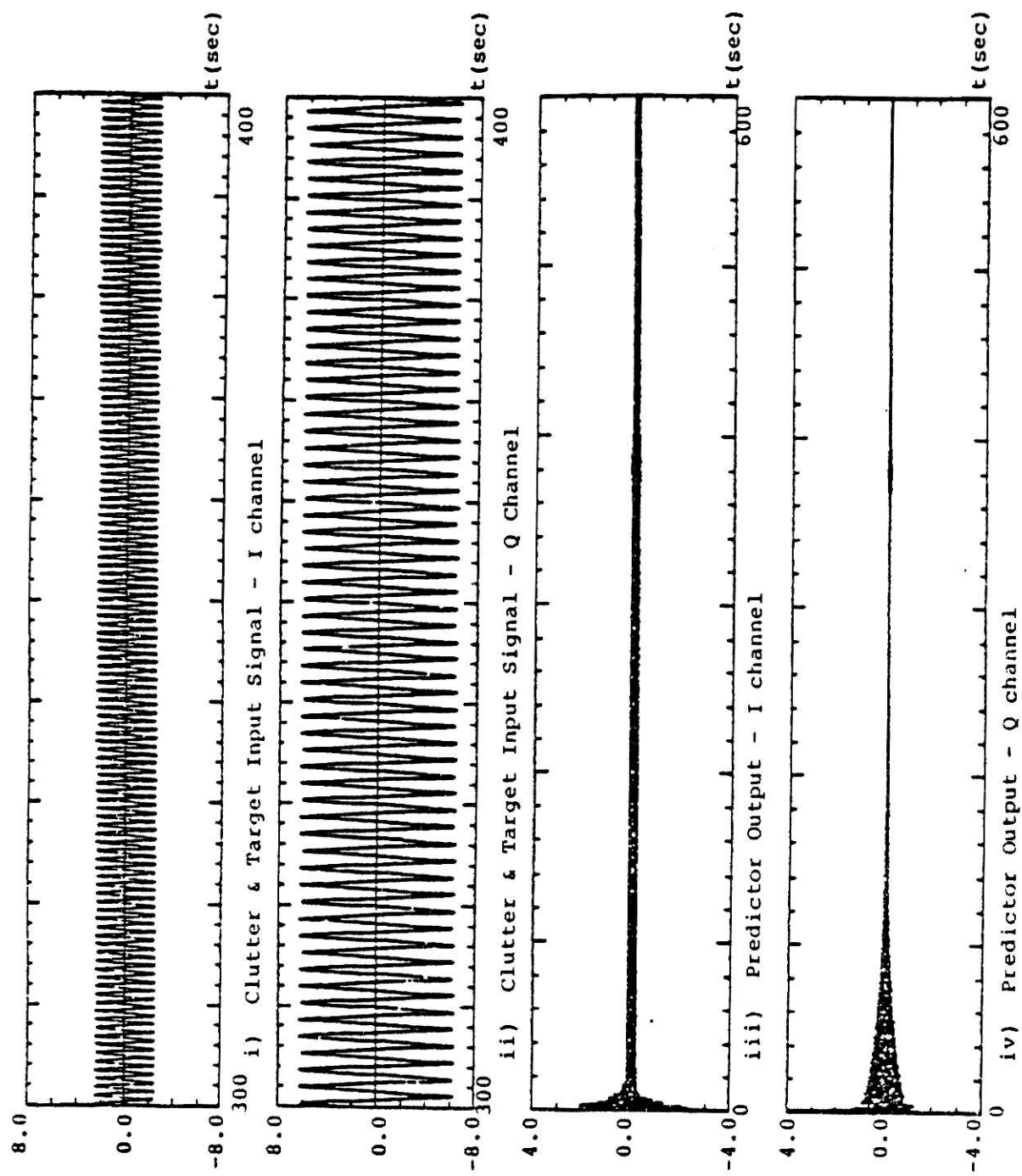
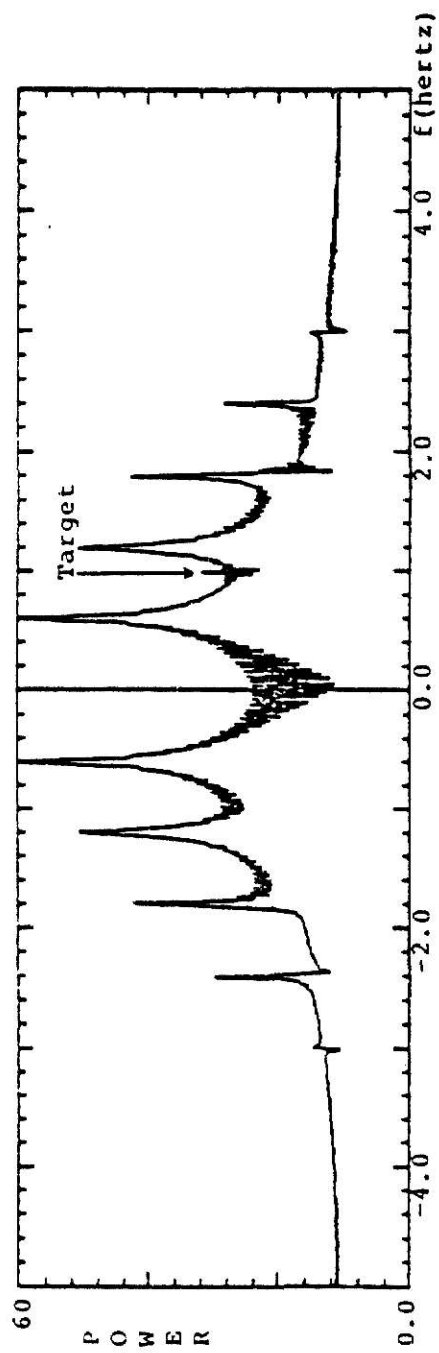
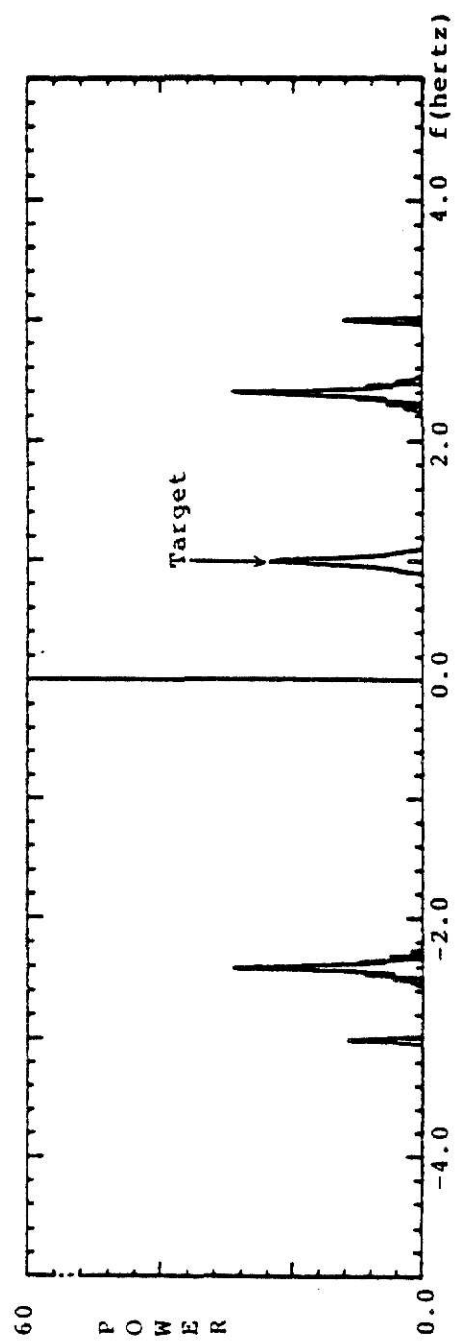


Figure 16a. Simulation Clutter Model (SCM); Case 1



i) PDS of Input Clutter & Target



ii) PDS of Predictor Output

Figure 16b. SCM; Case 1

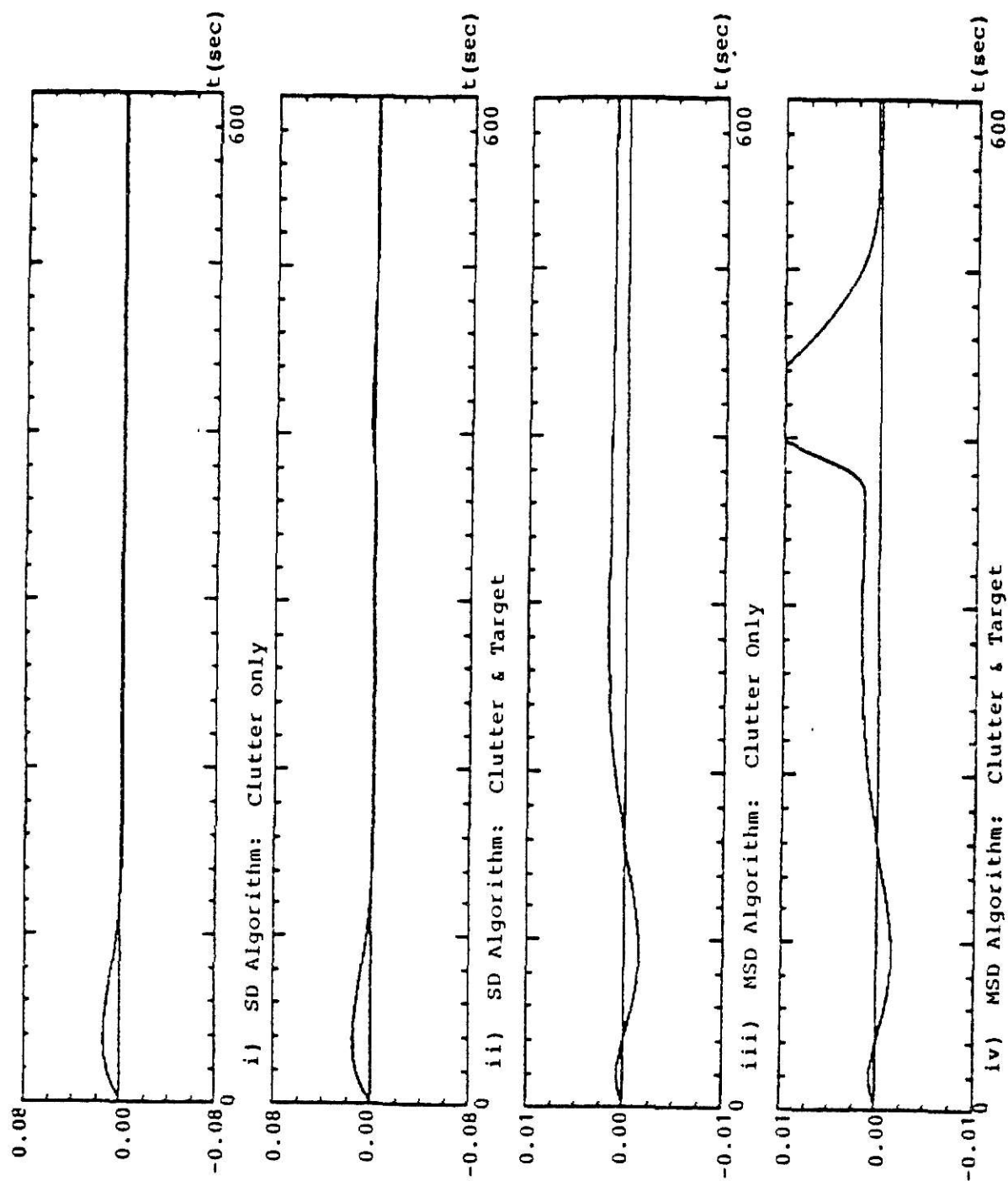


Figure 16c. SCM Integrator Output; Case 1

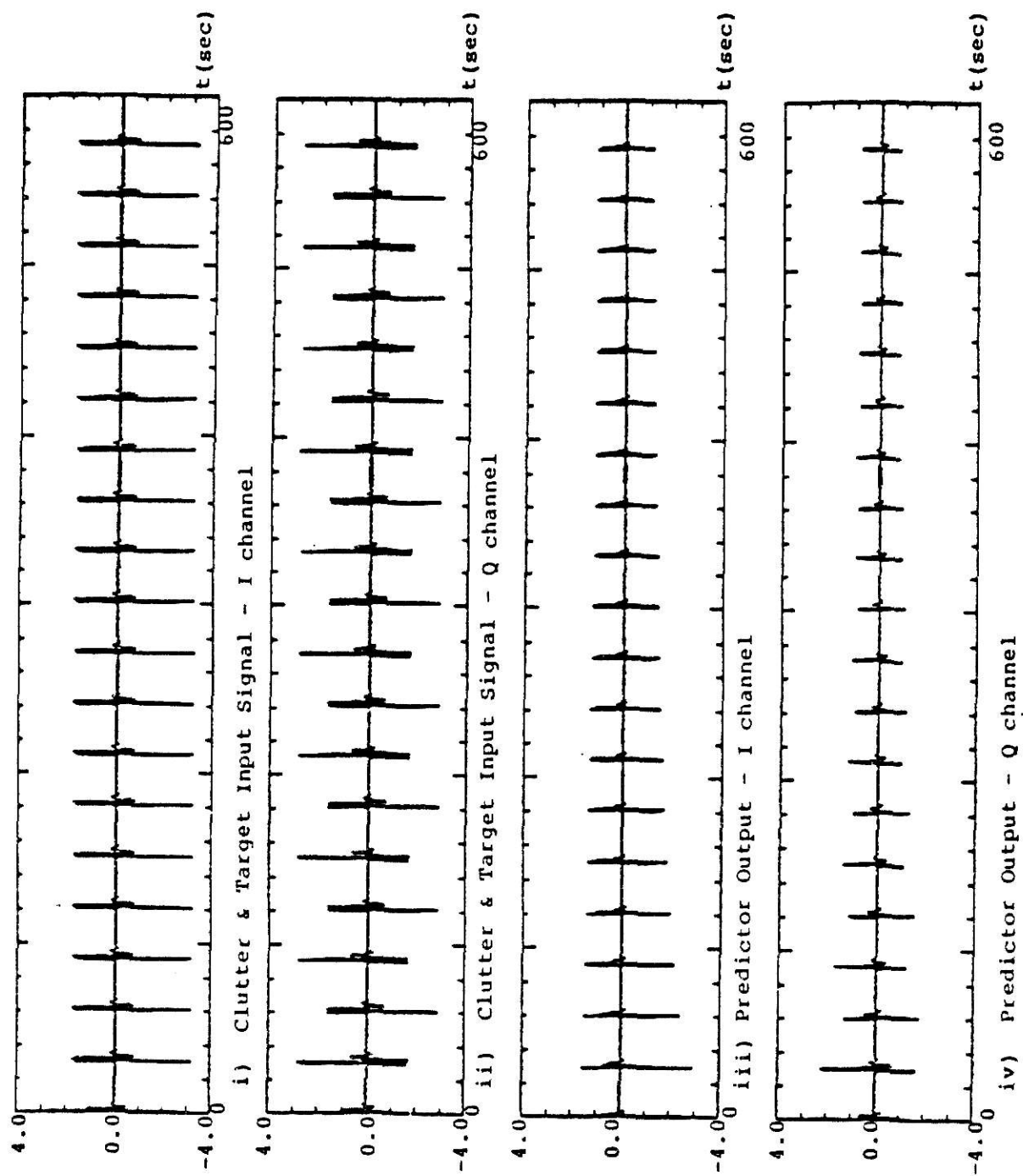
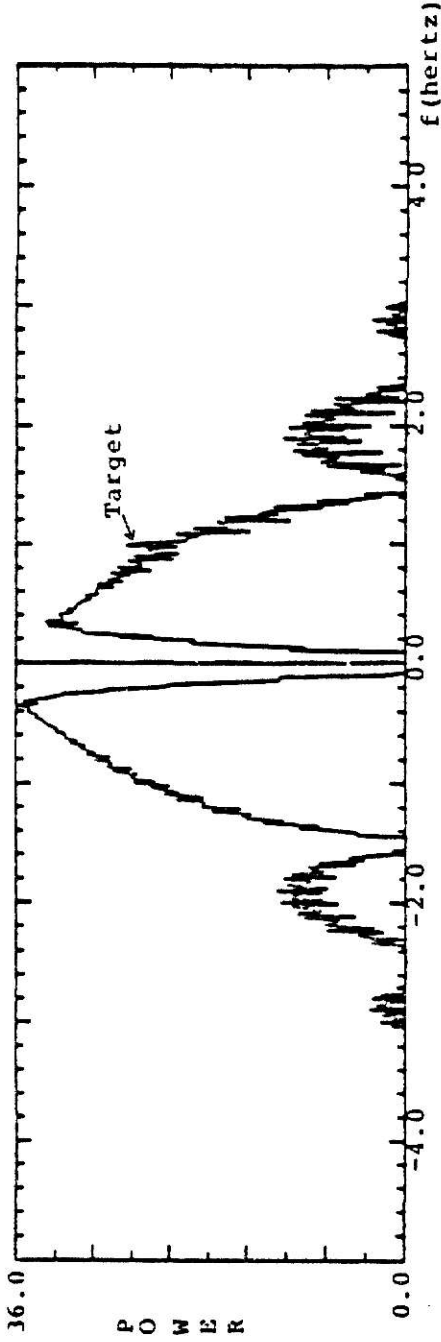
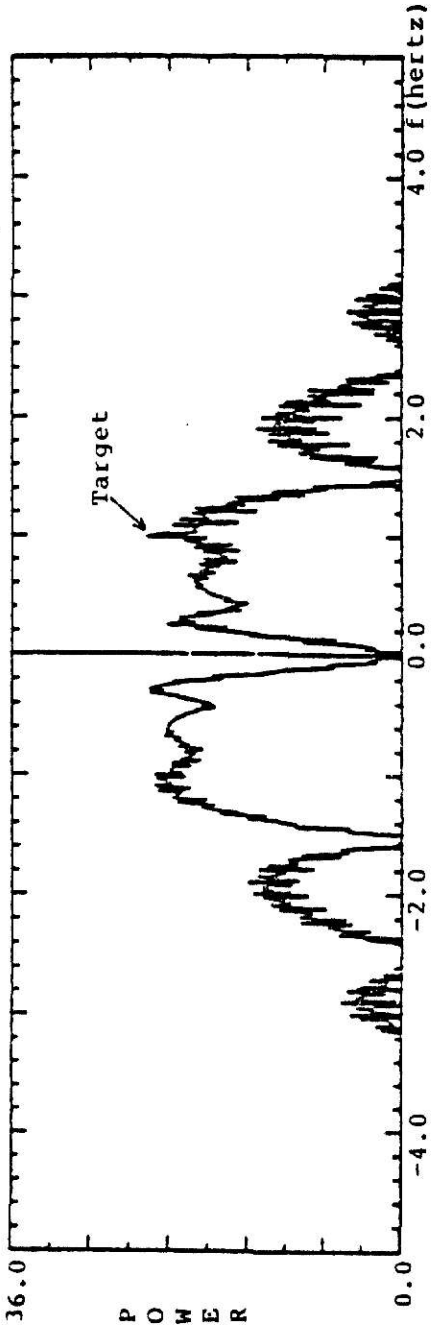


Figure 17a. SCM; Case 2



i) PDS of Input Clutter & Target



ii) PDS of Predictor Output

Figure 17b. SCM; Case 2

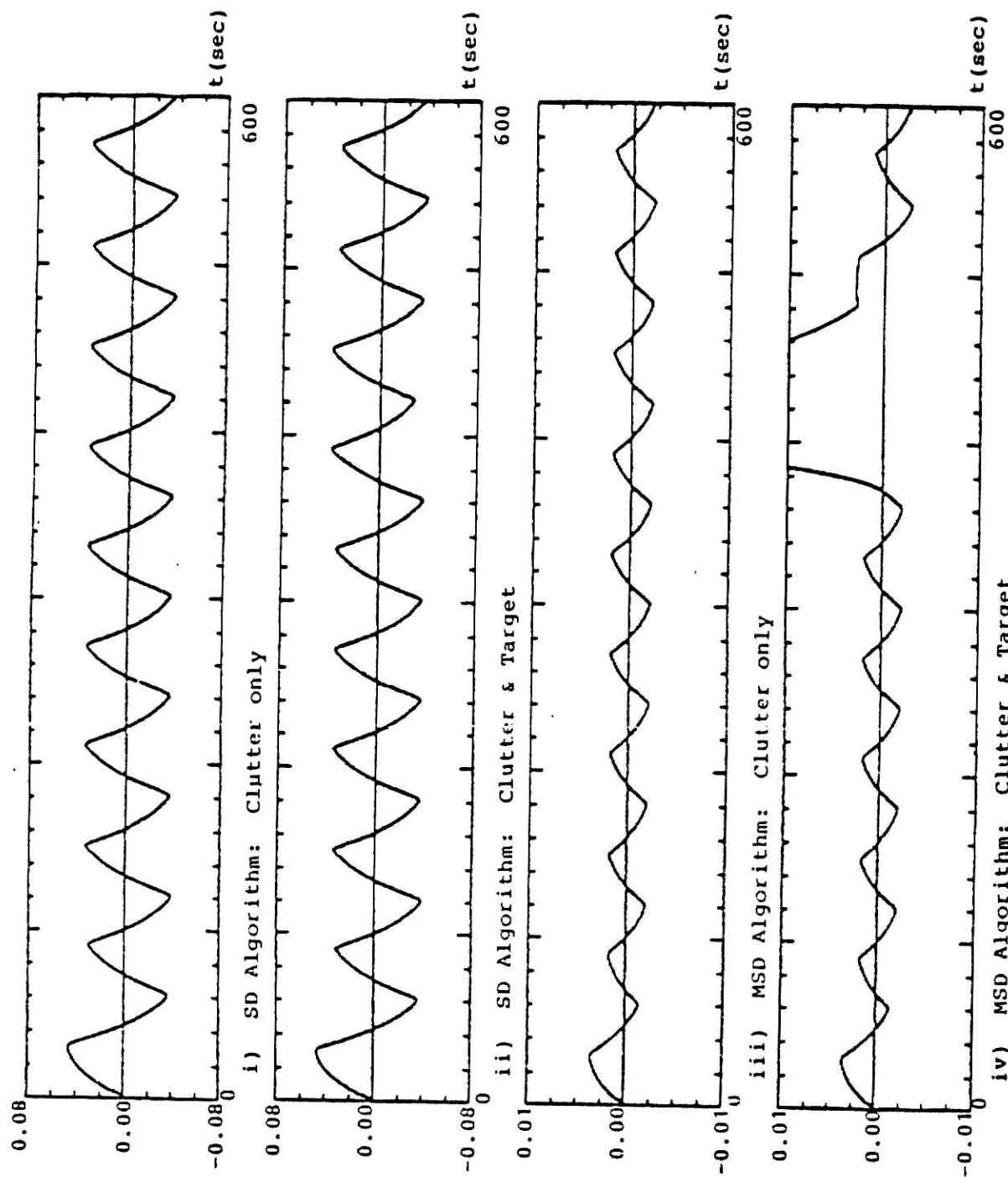


Figure 17c. SCM Integrator Output; Case 2

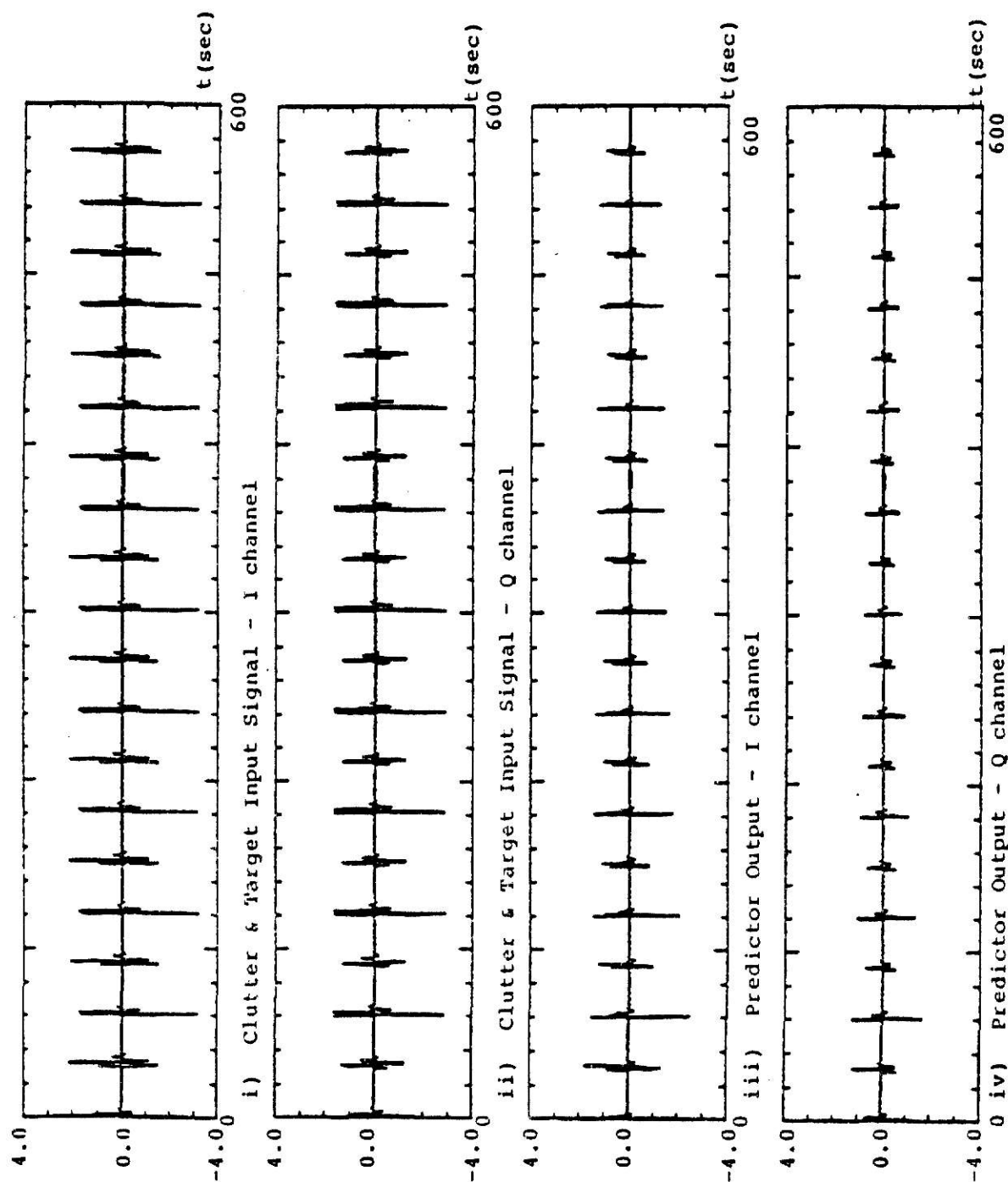
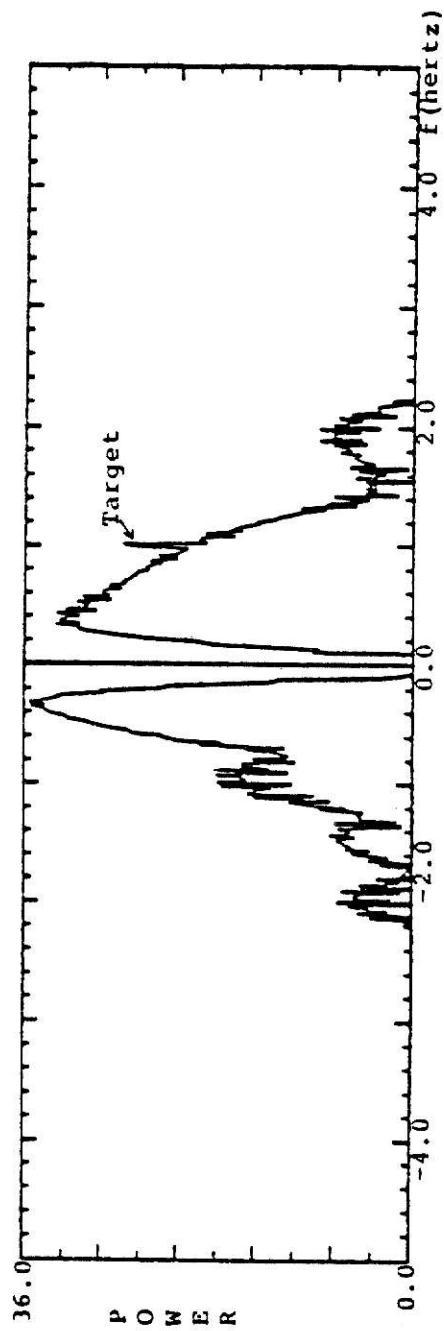
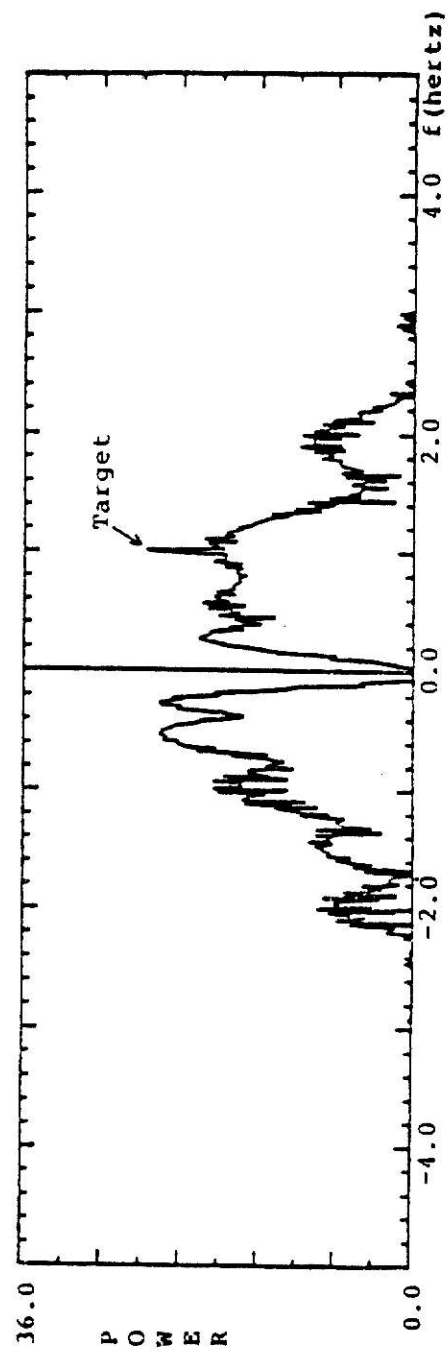


Figure 18a. SCM; Case 3



i) PDS of Input Clutter & Target



ii) PDS of Predictor Output

Figure 18b. SCM; Case 3

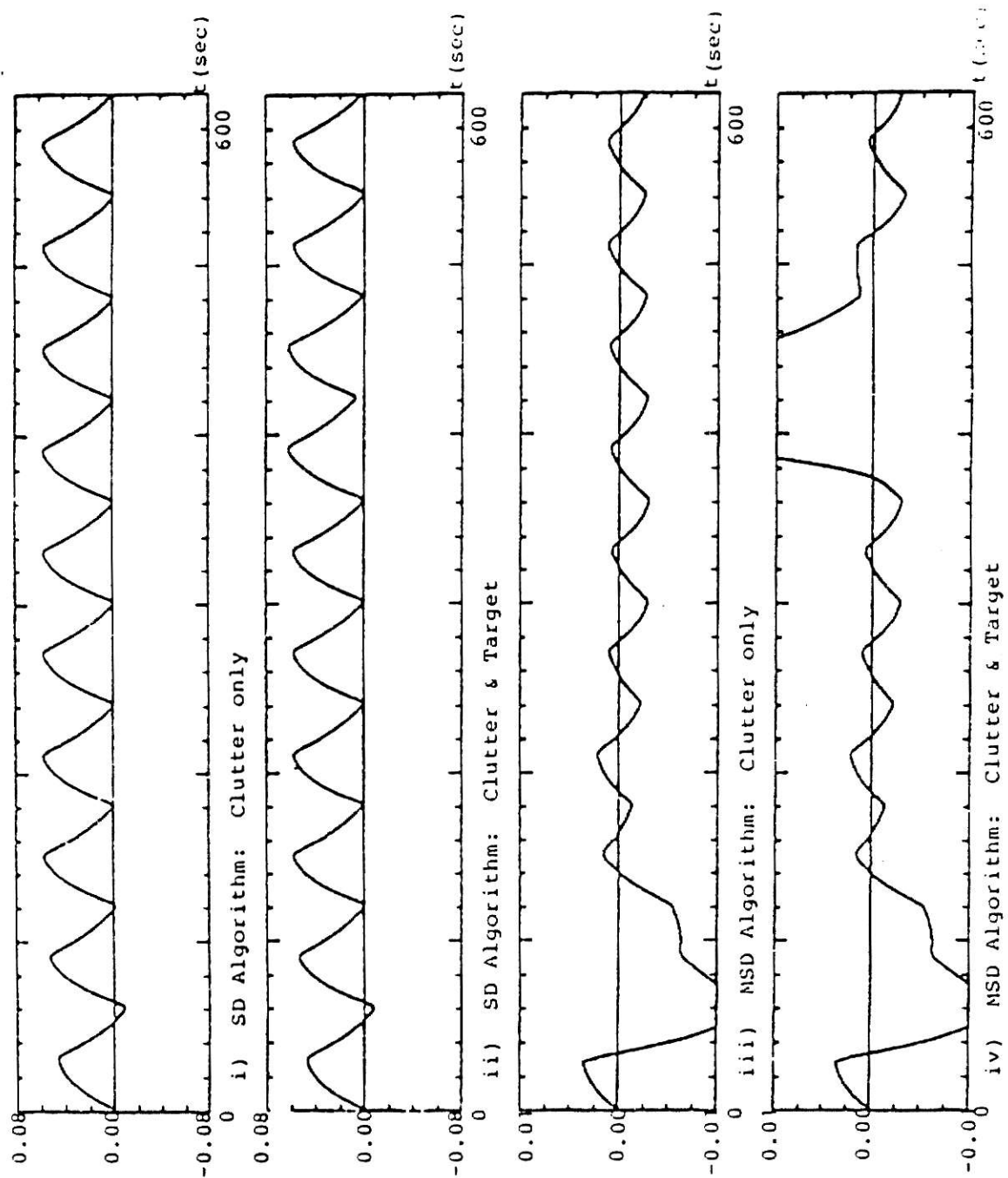


Figure 18c. SCM Integrator Output; Case 3

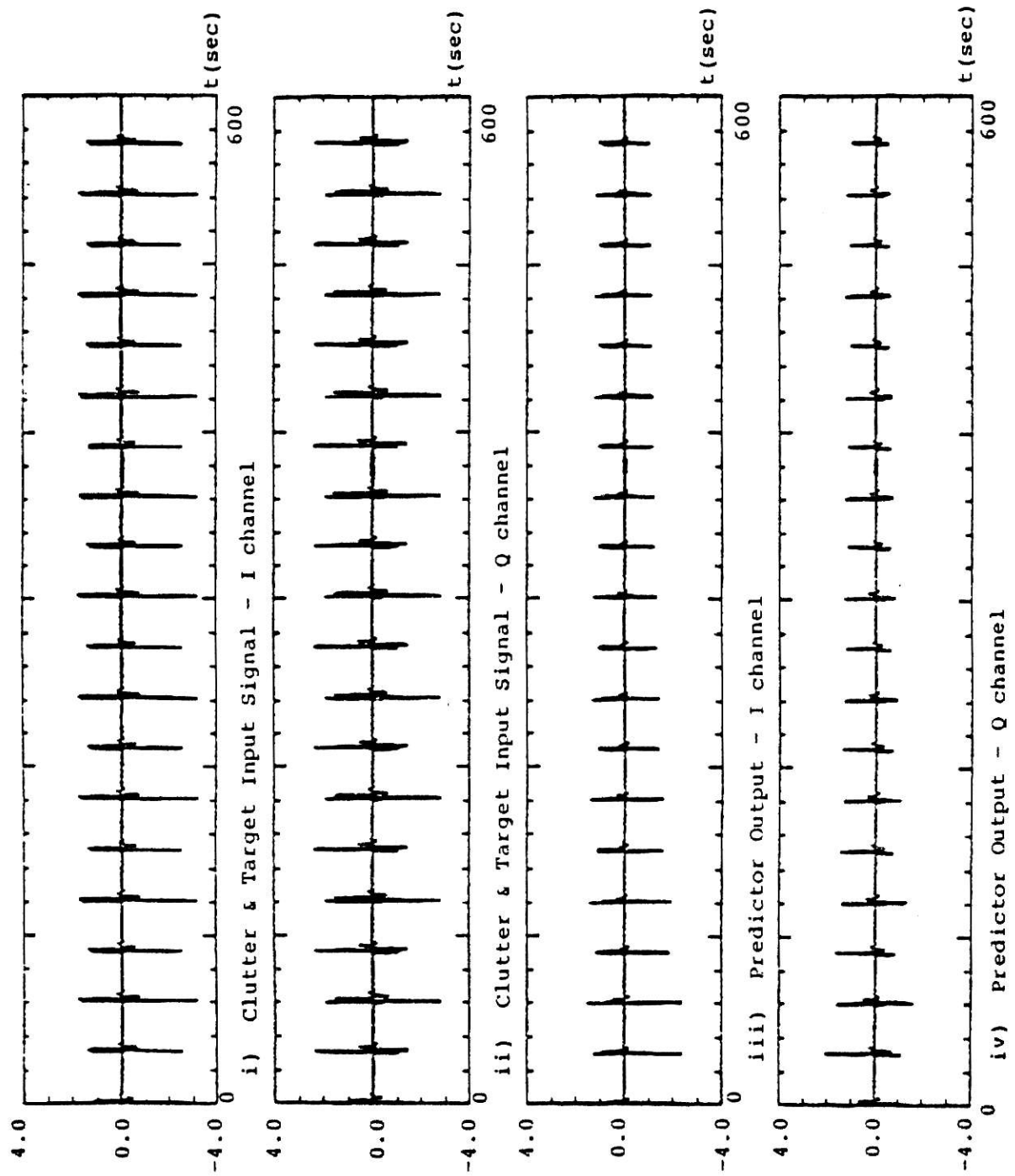
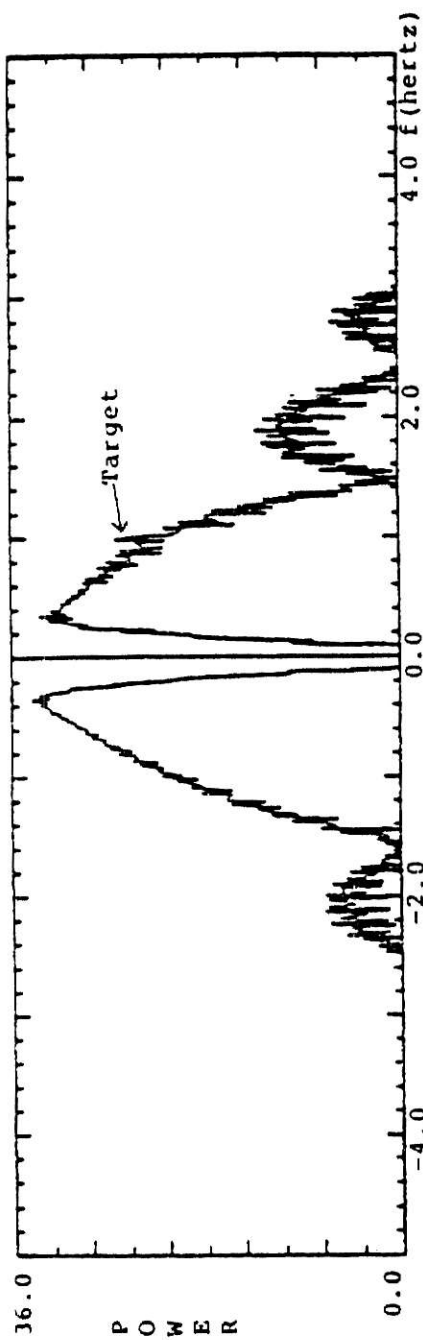
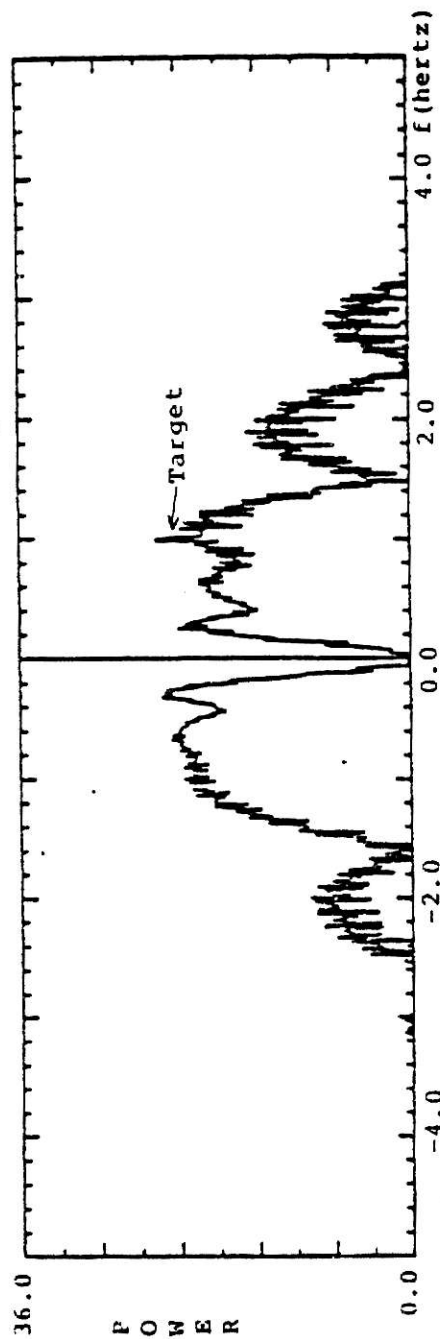


Figure 19a. SCM; Case 4



i) PSD of Input Clutter & Target



ii) PSD of Predictor Output

Figure 19b. SCM; Case 4

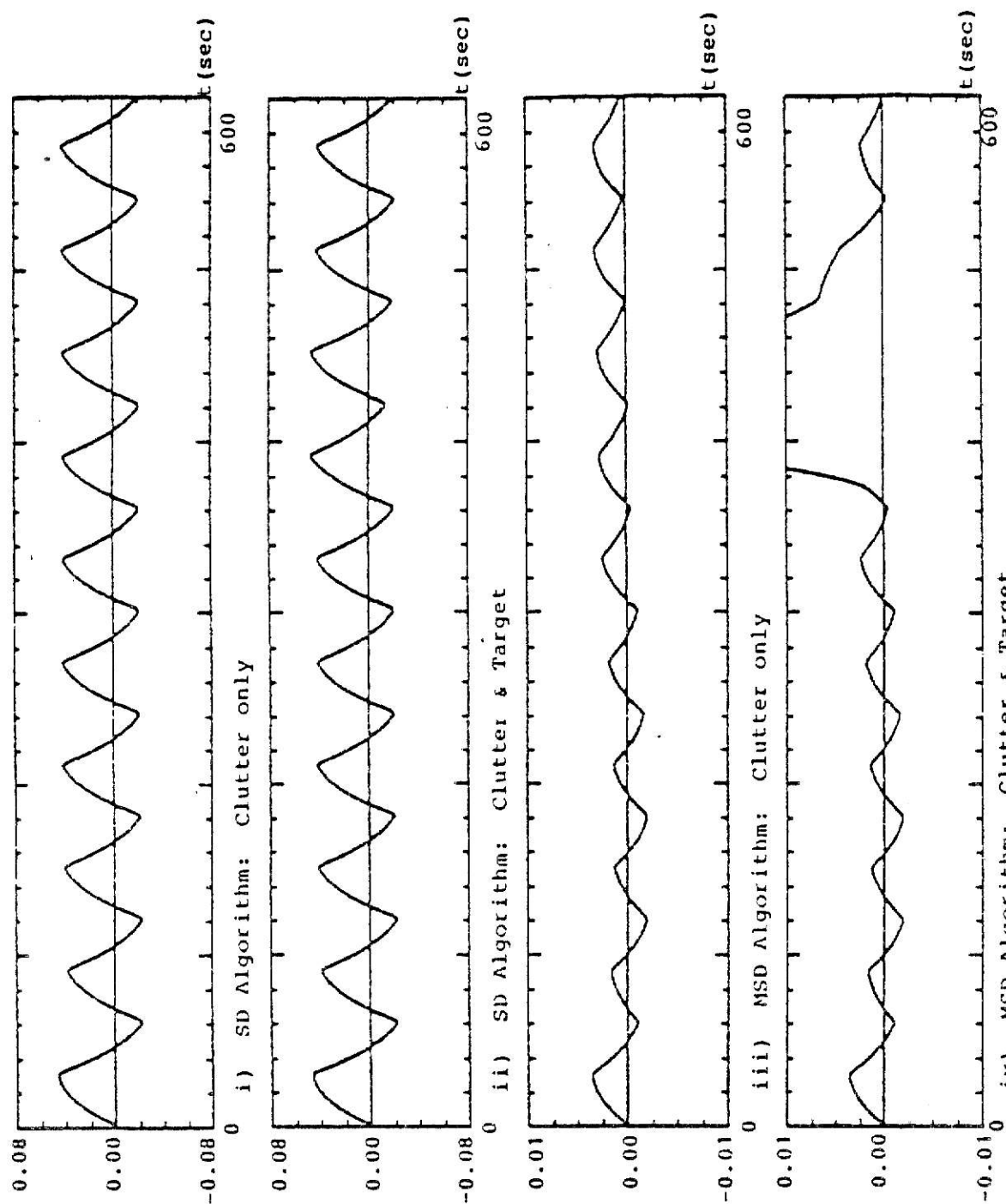


Figure 19c. SCM Integrator Output; Case 4

VII. CONCLUSIONS

This work has illustrated the effectiveness of the addition of adaptive complex prediction for improved target detection in an oscillatory clutter environment. The results of the computer simulations indicate that the ability to set a common threshold for detection using the MSD algorithm is greatly improved over the SD algorithm for which no common value exists which allows detection without nuisance alarms. The results of the simulation section can be extended to include other models of oscillating objects whose position function $r(t)$ is known. The choice of algorithm parameters is dependent on the model chosen for the clutter environment and target signal and must be chosen to enhance detection.

REFERENCES

1. H. P. Kalmus, "Direction Sensitive Doppler Device", Proceedings of the IRE, Vol. 43, pp. 698-700, 1955.
2. H. P. Kalmus, "Doppler Wave Recognition With High Clutter Rejection", IEEE Trans. Aerospace and Electronic Systems, Vol. AES-3, No. 6, pp. 334-339, Nov. 1967.
3. B. Widrow, J. McCool, and M. Ball, "The Complex LMS Algorithm", Proc. IEEE (Lett.), pp. 719-720, April, 1975.
4. M. P. Moya, Private Communication on Doppler Radar, Quadrature Demodulation, and Kalmus Processing, Apr., 1983.
5. G. Schnetzer, Private Communication on Quadrature Detection Concepts, Mar., 1983.

ACKNOWLEDGMENT

I would like to thank my friends and family, especially my parents for their encouragement and support of my academic endeavors.

I also wish to express my appreciation to Dr. Donald R. Hummels and Dr. Willard A. Parker for being members of my graduate committee. I particularly wish to thank my major advisor, Dr. Nasir Ahmed for his invaluable support and guidance. Finally, I wish to thank Mary Patterson Moya and George Schnetzer of Sandia National Laboratories, Albuquerque, N.M., for their suggestions and support.

APPENDIX A

DERIVATION OF CLMS ALGORITHM

The complex least mean square (CLMS) predictor is shown in Figure A1. The input signal is a sampled (discrete) function and the weights are variable. The output of the filter stage at time n is

$$y(n) = X_m^T W_m = W_m^T X_m \quad (A1)$$

where

$$X_m^T = [X(n-\Delta) \ X(n-\Delta-1) \ \dots X(n-\Delta-(m-1))] \quad (A2)$$

$$W_m^T = [W_1, W_2, \dots W_m] \quad (A3)$$

and m is the number of filter weights.

The error signal at time n is defined as

$$e(n) = X(n) - W_m^T X_m = X(n) - X_m^T W_m \quad (A4)$$

For complex inputs, each term in (A1-4) contains real and imaginary components, that is

$$y(n) = y_R(n) + j y_I(n) \quad (A5)$$

$$e(n) = e_R(n) + j e_I(n) \quad (A6)$$

and

$$X_m = \begin{bmatrix} X_R(n-\Delta) \\ X_R(n-\Delta-1) \\ \vdots \\ X_R(n-\Delta-(m-1)) \end{bmatrix} + j \begin{bmatrix} X_I(n-\Delta) \\ X_I(n-\Delta-1) \\ \vdots \\ X_I(n-\Delta-(m-1)) \end{bmatrix} = X_{m_R} + j X_{m_I} \quad (A7)$$

$$W_m = \begin{bmatrix} W_{1R} \\ W_{2R} \\ \vdots \\ W_{mR} \end{bmatrix} + j \begin{bmatrix} W_{1I} \\ W_{2I} \\ \vdots \\ W_{mI} \end{bmatrix} = W_{mR} + jW_{mI} \quad (A8)$$

where R denotes the direct or real signal component, I the 90° shifted quadrature or imaginary component, and $j = \sqrt{-1}$.

The CLMS must adapt to both the real and imaginary parts simultaneously, while minimizing in a sense both the real and imaginary components of the error $e(n)$. This is equivalent to

$$\nabla_w E\{e^2\} = \nabla_w E\{e \bar{e}\} \quad (A9)$$

where E represents the expected value operator, ∇ is the gradient operator, and $\bar{e}$ is the complex conjugate of e .

Since the two components of the error are in quadrature relationship, they cannot be minimized independently. Because the operation in (A9) is difficult to perform, it is modified as

$$\nabla_w \{e^2\} = \nabla_w (e \bar{e}) \quad (A10)$$

which is the instantaneous gradient of the squared error. The justification for this process is similar to that for the original LMS derivation and experimental results verify the validity of the simplification.

The conjugate of the complex error (A4) is

$$\overline{e(n)} = \overline{X(n)} - \overline{W}_m^T \overline{X}_m = \overline{X(n)} - \overline{X}_m^T \overline{W}_m \quad (A11)$$

The instantaneous gradient of $e(n) \overline{e(n)}$ with respect to the real component of the weight vector is

$$\begin{aligned} \nabla_{W_R} \{e(n) \overline{e(n)}\} &= \begin{bmatrix} \frac{\partial (e(n) \overline{e(n)})}{\partial W_{1R}} \\ \frac{\partial (e(n) \overline{e(n)})}{\partial W_{2R}} \\ \vdots \\ \frac{\partial (e(n) \overline{e(n)})}{\partial W_{mR}} \end{bmatrix} = e(n) \nabla_{W_R} (\overline{e(n)}) + \overline{e(n)} \nabla_{W_R} (e(n)) \\ &= e(n) (-\overline{X}_m) + \overline{e(n)} (-X_m) \end{aligned} \quad (A12)$$

Similarly, the instantaneous gradient with respect to the imaginary component of the weight vector is

$$\begin{aligned} \nabla_{W_I} \{e(n) \overline{e(n)}\} &= e(n) \nabla_{W_I} (\overline{e(n)}) + \overline{e(n)} \nabla_{W_I} (e(n)) \\ &= e(n) (j\overline{X}_m) + \overline{e(n)} (-jX_m) \end{aligned} \quad (A13)$$

Applying the method of steepest descent to the real and imaginary parts of the weight vector which is the same as changing them along their respective negative gradient estimates, we obtain

$$W_{R_i}(n+1) = W_{R_i}(n) - \mu \nabla_{W_R} \{e(n) \overline{e(n)}\} \quad (A14)$$

$$W_{I_i}(n+1) = W_{R_i}(n) - \mu \nabla_{W_I} \{e(n) e(n)\} \quad (A15)$$

where $W_{R_i}(n)$ implies the real part of i th weight at time n . Since the weight vector W is complex, it can be expressed as

$$W_i(n) = W_{R_i}(n) + jW_{I_i}(n) \quad (A16)$$

$$W_i(n+1) = W_i(n) - \mu [\nabla_{W_R} \{e(n) \overline{e(n)}\} + j \nabla_{W_I} \{e(n) \overline{e(n)}\}] \quad (A17)$$

If the expressions for the gradients in (A12) and (A13) are substituted in (A17), we obtain the update equation for the i th filter weight as

$$\begin{aligned} W_i(n+1) = W_i(n) - \mu [e(n) \{-\overline{X(n-\Delta-(i-1))}\} + \overline{e(n)} \{-X(n-\Delta-(i-1))\}] \\ + j e(n) \{j \overline{X(n-\Delta-(i-1))}\} + j \overline{e(n)} \{-j X(n-\Delta-(i-1))\}] \end{aligned}$$

$$W_i(n+1) = W_i(n) + \mu 2 e(n) \overline{X(n-\Delta-(i-1))}$$

$$W_i(n+1) = W_i(n) + 2\mu e(n) \overline{X(n-\Delta-(i-1))}$$

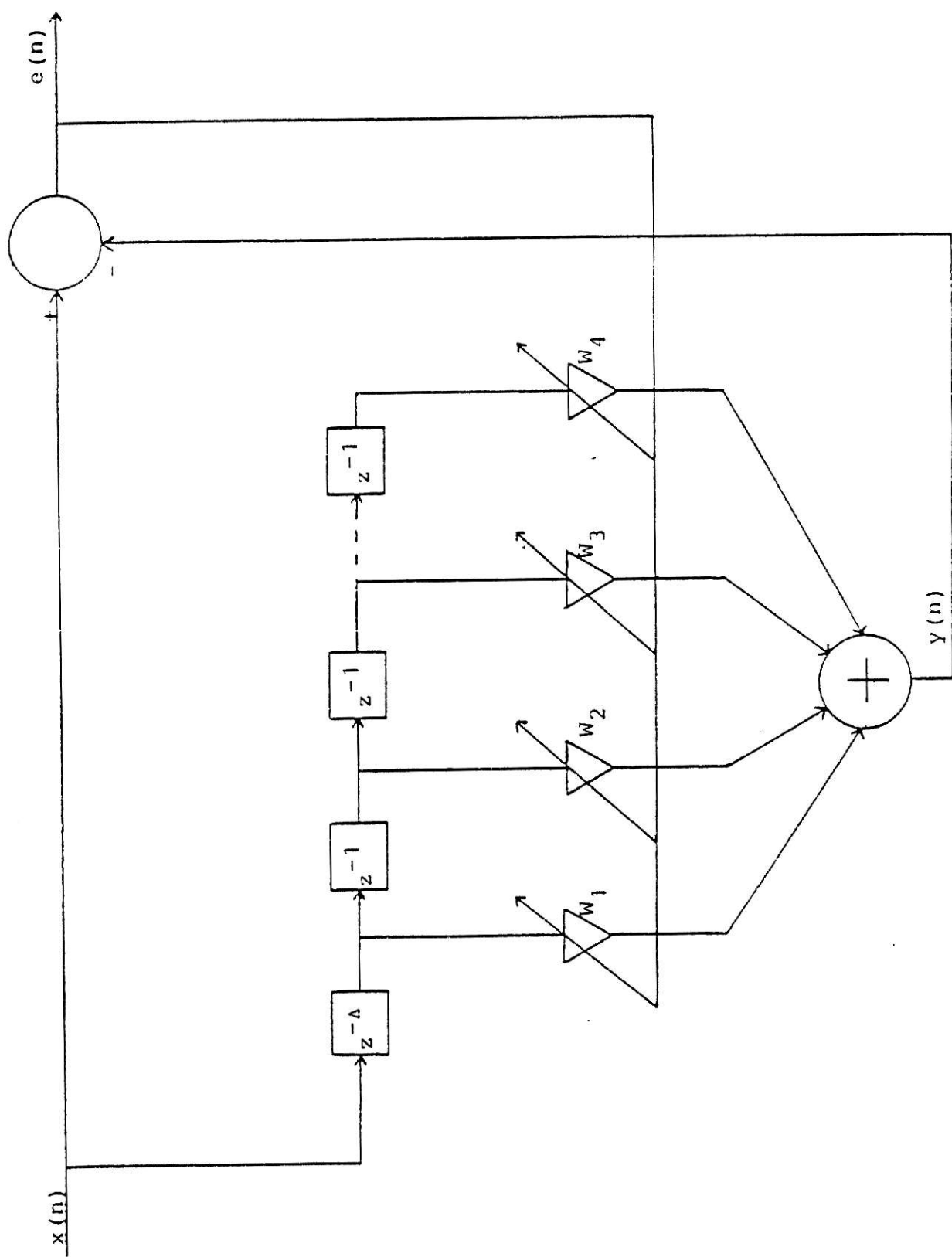


Figure A1. Complex Least Mean Square Adaptive Predictor Block Diagram

APPENDIX B

FILTER TRANSFER FUNCTIONS

The following equations were used to perform the Kalman Process of Figure 8b.

$$\text{LPF:} \quad H(z) = \frac{0.754763 (1 + z^{-1})}{1 + 0.509526 z^{-1}} \quad (\text{B1})$$

$$\text{HPF:} \quad H(z) = \frac{0.245237 (1 - z^{-1})}{1 + 0.509526 z^{-1}} \quad (\text{B2})$$

with

$$H(z) = \frac{Y(z)}{X(z)}$$

where $X(z)$ is the input to the filter and $Y(z)$ is the output.

APPENDIX C

COMPUTER PROGRAMS

The computer programs used for implementing the MSD and SD algorithms along with those for data generation are presented. The parameters used for data generation for the hypothetical and simulation results are given in Tables 1 and 2. The number of filter weights as well as the value for the convergence parameter μ for the CLMS are given in the appropriate sections. The following values were used in all cases for implementing the Kalmus Processor and SSE.

Kalmus Processor: Input gain = 18 dB

SSE: Desired Signal Strength = 0.1

Minimum Signal Strength = 0.001

Beta = 0.95

```

C*****
C
C      COMPLEX LEAST MEAN SQUARE PREDICTOR
C
C      DG FORTRAN 5 SOURCE FILENAME:          CLMS.FR
C
C      DEPARTMENT OF ELECTRICAL ENGINEERING    KANSAS STATE UNIVERSITY
C
C      REVISION          DATE          PROGRAMMER
C      -----          ----          -
C      00.0              OCT 27, 1982    WAYNE M. BLASI
C
C*****
C
C      PURPOSE
C
C      THIS ROUTINE IMPLEMENTS WIDROW'S COMPLEX LEAST
C      MEAN SQUARE PREDICTOR. SEE <3>
C
C      ROUTINE(S) CALLED BY THIS ROUTINE
C
C      OPENR
C      OPENW
C      READR
C      WRITR
C
C*****
C
C
C      REAL I(256),Q(256)
C      COMPLEX W(64)/64*(0.0,0.0)/,G,E(256)
C      COMPLEX X(320)/320*(0.0,0.0)/
10  CONTINUE
C      ACCEPT'NUMBER OF FILTER WEIGHTS (1-64)? ',N
C      IF(N.LT.1.OR.N.GT.64)GO TO 10
C      ACCEPT'VALUE OF MU= ? ',U
C      CALL OPENR(0,'INPUT REAL FILENAME ? ',1024,SIZEI)
C      CALL OPENR(1,'INPUT IMAG. FILENAME ? ',1024,SIZEQ)
C      CALL OPENW(2,'OUTPUT REAL FILENAME ? ',1024,SIZE)
C      CALL OPENW(3,'OUTPUT IMAG. FILENAME ? ',1024,SIZE)
C      T=AMIN1(SIZEI,SIZEQ)
C      T=T/1024
C      M=IFIX(T)
C      DO 90 J=1,M
C      CALL READR(0,J,I,1,IER)
C      CALL READR(1,J,Q,1,IERR)
C
C      SET UP ARRAYS TO ALLOW NEGATIVE INDICES (n-i)
C
C      DO 30 L=1,N
C      X(L)=X(256+L)
30  CONTINUE
C      DO 40 L=N+1,256+N
C      X(L)=CMPLX(I(L-N),Q(L-N))
40  CONTINUE
C
C      PERFORM FILTER OPERATION

```

```
      DO 70 K=N+1,256+N
          G=(0.0,0.0)
          DO 50 L=1,N
              G=W(L)*X(K-L)+G
50          CONTINUE
C
C      COMPUTE ERROR AT TIME K
C
          E(K-N)=X(K)-G
C
C      UPDATE FILTER WEIGHTS
C
          DO 60 LL=1,N
              W(LL)=W(LL)+U*E(K-N)*CONJG(X(K-LL))
60          CONTINUE
70      CONTINUE
          DO 80 MI=1,256
              I(MI)=REAL(E(MI))
              Q(MI)=AIMAG(E(MI))
80          CONTINUE
          CALL WRITR(2,J,I,1,IERR)
          CALL WRITR(3,J,Q,1,IERR)
90      CONTINUE
          CALL RESET
          STOP***NORMAL TERMINATION***
          END
```

```

C*****
C
C      KALMUS PROCESSOR
C
C      DG FORTRAN 5 SOURCE FILENAME:          IDQDT.FR
C
C      DEPARTMENT OF ELECTRICAL ENGINEERING    KANSAS STATE UNIVERSITY
C
C      REVISION          DATE          PROGRAMMER
C      -----          ----          -
C      00.0              AUG 26, 1982    SIRAVARA VIJAYENDRA
C      01.0              APR 10, 1983    WAYNE M. BLASI
C
C*****
C
C      PURPOSE
C
C          THIS ROUTINE IMPLEMENTS THE I*dQ/dt OPERATION
C          OF THE KALMUS PROCESSOR. SEE <1>,<2>.
C
C      ROUTINE(S) CALLED BY THIS ROUTINE
C
C          ONEPOL
C          OPENR
C          OPENW
C          READR
C          WRITR
C
C*****
C
C
C      REAL INPHAS(256),QUAD(256),IQOUT(256)
C      REAL I1(2),I2(2),Q1(2),Q2(2),AHP(3),ALP(3)
C      INTEGER PTR
C
C      NB = 1024
C      NBYTE = 4
C      CALL OPENR(0,'INPHASE FILE NAME ? : ',NB,F1)
C      CALL OPENR(1,'Q-PHASE FILE NAME ? : ',NB,F2)
C      CALL OPENR(2,'FILTER COEFICIENT FILE NAME ? : ',NBYTE,F3)
C      CALL OPENW(3,'I * DQ/DT OUTPUT FILE NAME ? : ',NB,F4)
C
C      PARAMETER FETCH
C
C      ACCEPT ' INPUT GAIN IN dB ? : ',DBGAIN
C      GAIN = 10 ** (DBGAIN/20.0)
C
C      READ LOW-PASS FILTER COEFFICIENTS
C
C      PTR = 0.0
C      DO 10 J = 1,3
C      PTR = PTR + 1
C      CALL READR(2,PTR,ALP(J),1,IER)
10  CONTINUE
C
C      READ HIGH-PASS FILTER COEFFICIENTS
C

```

```

DO 20 J = 1,3
PTR = PTR + 1
CALL READR(2,PTR,AHP(J),1,IER)
20 CONTINUE
C
DO 30 J=1,2
I1(J)=0.0
I2(J)=0.0
Q1(J)=0.0
Q2(J)=0.0
30 CONTINUE
C
C   ALGORITHM EXECUTION
C
ITER = AMIN1(F1,F2)/1024
DO 100 K = 1,ITER
CALL READR(0,K,INPHAS,1,IER)
CALL READR(1,K,QUAD,1,IER)
DO 50 J = 1,256
I1(1)=INPHAS(J) * GAIN
Q1(1)=QUAD(J) * GAIN
C
C   LOW PASS FILTER THE INPHASE CHANNEL AND HIGH PASS FILTER THE
C   QUADRATURE CHANNEL
C   EACH ONE POLE FILTER HAS 0 dB GAIN AND 4 Hz CUTOFF
C
CALL ONEPOL(I1,I2,ALP)
CALL ONEPOL(Q1,Q2,AHP)
C
C   MULTIPLY BY GAIN FACTORS AND LIMIT
C
I = I2(1)
Q = Q2(1) * 6.3
C
IF(ABS(I).GT.2.8) I = SIGN(2.8,I)
IF(ABS(Q) .GT.4.0) Q = SIGN(4.0,Q)
I = I * 2.0
C
C   FORM PRODUCT OF I AND Q CHANNELS
C
IQOUT(J) = I * Q/10.0
C
C   SAVE I AND Q CAHNNEL VALUES FOR FILTER OPERATION
C
I1(2)=I1(1)
Q1(2)=Q1(1)
50 CONTINUE
CALL WRITR(3,K,IQOUT,1,IER)
100 CONTINUE
CALL RESET
STOP
END

```

```

C*****72
C
C      ONEPOLE FILTER
C
C      DG FORTRAN 5 SOURCE FILENAME:      ONEPOL.FR
C
C      DEPARTMENT OF ELECTRICAL ENGINEERING  KANSAS STATE UNIVERSITY
C
C      REVISION      DATE      PROGRAMMER
C      -----      ----      -
C      00.0          AUG 26, 1982      SIRAVARA VIJAYENDRA
C      .
C*****
C
C      CALLING SEQUENCE
C
C          CALL ONEPOL(X,Y,A)
C
C      PURPOSE
C
C          THIS SUBROUTINE IMPLEMENTS A ONE POLE FILTER. THE
C          TYPE,i.e. HP OR LP IS DETERMINED BY THE COEFFICIENTS
C          PASSED AS PARAMETERS.
C
C      ROUTINE(S) CALLED BY THIS ROUTINE
C
C          NONE
C
C      ARGUMENT(S) REQUIRED FROM THE CALLING ROUTINE
C
C          X - ARRAY CONTAINING PAST AND PRESENT INPUT VALUES
C          Y - ARRAY CONTAINING PAST FILTER OUTPUTS
C          A - FILTER COEFFICIENT VALUES
C
C      ARGUMENT(S) SUPPLIED TO THE CALLING ROUTINE
C
C          Y - PRESENT FILTER OUTPUT
C*****
C
C      NOTE 1: This subroutine makes no checks on the validity
C              of the data supplied by the calling routine.
C*****
C
C      SUBROUTINE ONEPOL(F,G,A)
C      REAL F(2),G(2),A(3)
C      G(2) = G(1)
C      G(1) = A(1)*F(1) + A(2) * F(2) - A(3) * G(2)
C      RETURN
C      END

```

```

C*****
C
C          SIGNAL STRENGTH EQUALIZER
C
C          DG FORTRAN 5 SOURCE FILENAME:          SSE.FR
C
C          DEPARTMENT OF ELECTRICAL ENGINEERING    KANSAS STATE UNIVERSITY
C
C          REVISION          DATE          PROGRAMMER
C          -----          ----          -
C          00.0              OCT 10, 1982      SIRAVARA VIJAYENDRA
C
C*****
C          PURPOSE
C
C          THIS ROUTINE IMPLEMENTS THE SIGNAL STRENGTH
C          EQUALIZER OF THE MSD ALGORITHM.
C
C          ROUTINE(S) CALLED BY THIS ROUTINE
C
C          OPENR
C          OPENW
C          READR
C          WRITR
C
C*****
C
C          DIMENSION X(256)
C          ACCEPT'DESIRED SIGNAL STRENGTH =:',DSTR
C          ACCEPT'BETA =:',BETA
C          ACCEPT'MIN SIG. STRENGTH =:',SMIN
C          NB = 1024
C          CALL OPENR(0,' INPUT FILE NAME ? : ',NB,F)
C          CALL OPENW(1,' OUTPUT FILE - AFTER GAIN : ',NB,F1)
C          ITER = IFIX(F/1024)
C          SXX = 0.0
C          DO 10 J = 1,ITER
C          CALL READR(0,J,X,1,IER)
C          DO 20 K = 1,256
C
C          ESTIMATE SIGNAL STRENGTH
C
C          SXX = SXX * BETA + (1.0 - BETA) * ABS(X(K))
C          IF(SXX.LE.SMIN) SXX = 1.0
C          W = DSTR/(SXX + .0001)
C
C          SCALE INPUT
C
C          X(K) = X(K)*W
20      CONTINUE
C          CALL WRITR(1,J,X,1,IER)
10      CONTINUE
C          CALL RESET
C          STOP***NORMAL TERMINATION***
C          END

```

```

C*****
C
C      TRIANGULAR TARGET SIGNAL GENERATOR
C
C      DG FORTRAN 5 SOURCE FILENAME:          TRITARGET.FR
C
C      DEPARTMENT OF ELECTRICAL ENGINEERING    KANSAS STATE UNIVERSITY
C
C      REVISION          DATE          PROGRAMMER
C      -----          ----          -
C      00.0              JAN 25, 1983    WAYNE M. BLASI
C
C*****
C
C      PURPOSE
C
C          THIS ROUTINE GENERATES A ONE-SIDED TRIANGULAR
C          TARGET SIGNAL.
C
C      ROUTINE(S) CALLED BY THIS ROUTINE
C
C          OPENW
C          WRITR
C
C*****
C
C      THIS ROUTINE WAS ADAPTED FROM SOFTWARE FROM SANDIA NATIONAL
C      LABORATORIES
C
C*****
C
C      REAL IOUT(256),QOUT(256)
C      PI=3.14159
C      TWOPI=2.0*PI
C      ACCEPT'SAMPLING RATE ? ',FS
C      ACCEPT'FREQUENCY OF TARGET MOTION ? ',FREQ
C      ACCEPT'DELAY PRECEEDING TARGET ENTRANCE (SEC) ? ',DELAY
C      ACCEPT'TARGET DURATION ? ',DURATN
C      ACCEPT'TARGET AMPLITUDE ? ',AMP
C      ACCEPT'NUMBER OF BLOCKS (256 PTS./BLK) ? ',NBLOCK
C      CALL OPENW(0,'INPHASE OUTPUT FILENAME? ',1024,F1)
C      CALL OPENW(1,'Q-PHASE OUTPUT FILENAME? ',1024,F2)
C      ISTART=IFIX(DELAY*FS)
C      LENGTH=IFIX(DURATN*FS)
C      C=0.0
C      DO 100 J=1,NBLOCK
C
C      GENERATE TRIANGULAR SIGNAL
C
C      DO 50 L=1,256
C      TRIANG=0.0
C      IF(C.GE.ISTART.AND.C.LE.ISTART+LENGTH/2)TRIANG=
*  FLOAT((C-ISTART)*AMP)/FLOAT(LENGTH/2)
C      IF(C.GE.ISTART+LENGTH/2.AND.C.LE.ISTART+LENGTH)
*  TRIANG=FLOAT(-1*(C-LENGTH-ISTART)*AMP)/FLOAT(LENGTH/2)
C      IOUT(L)=TRIANG*COS(TWOPI*C*FREQ/FS)
C      QOUT(L)=TRIANG*SIN(TWOPI*C*FREQ/FS)

```

```
      C=C+1.0
50    CONTINUE
      CALL WRITR(0,J,IOUT,1,IER)
      CALL WRITR(1,J,ROUT,1,IERR)
100   CONTINUE
      CALL RESET
      STOP***NORMAL TERMINATION***
      END
```

```

C*****
C
C SINUSOIDAL CLUTTER SIGNAL GENERATOR
C
C DG FORTRAN 5 SOURCE FILENAME: CLUTTER1.FR
C
C DEPARTMENT OF ELECTRICAL ENGINEERING KANSAS STATE UNIVERSITY
C
C REVISION DATE PROGRAMMER
C -----
C 00.0 APR 10, 1983 WAYNE M. BLASI
C
C*****
C
C PURPOSE
C
C THIS ROUTINE GENERATES A TWO-SIDED CLUTTER SIGNAL.
C THE POSITION FUNCTION IS A SINUSOID.
C
C ROUTINE(S) CALLED BY THIS ROUTINE
C
C OPENW
C WRITR
C
C*****
C
C Note: THIS ROUTINE GENERATES THE CLUTTER SIGNALS FOR THE
C SIMULATION RESULTS - CASE 1.
C
C*****
C REAL OUTI(256),OUTQ(256)
C PI=3.14159
C TWOPI=2.0*PI
C ACCEPT'SAMPLING FREQUENCY = ',FS
C ACCEPT'FREQUENCY OF MECHANICAL MOTION = ',FREQ
C ACCEPT'PHASE OF SIGNAL = ',PHA
C ACCEPT'# OF BLOCKS(256 PTS./BLOCK) = ',N
C ACCEPT'AMPLITUDE = ',AMP
C CALL OPENW(0,'INPHASE OUTPUT FILENAME? ',1024,SIZEI)
C CALL OPENW(1,'Q-PHASE OUTPUT FILENAME? ',1024,SIZEQ)
C K=0
C DO 99 I=1,N
C DO 10 J=1,256
C
C GENERATE POSITION FUNCTION
C
C ANG=TWOPI*FREQ*FLOAT(K)/FS
C A=SIN(ANG)
C ARG=(TWOPI*PHA)+(PI/2.0*A)
C
C GENERATE OUTPUT SIGNALS
C
C OUTI(J)=AMP*COS(ARG)
C OUTQ(J)=AMP*SIN(ARG)
C K=K+1
10 CONTINUE
C CALL WRITR(0,I,OUTI,1,IER)

```

99

```
CALL WRITR(1,I,OUTQ,1,IERR)  
CONTINUE  
CALL RESET  
STOP***NORMAL TERMINATION***  
END
```

77

```

C*****
C
C      TRAPEZOIDAL CLUTTER SIGNAL GENERATOR
C
C      DG FORTRAN 5 SOURCE FILENAME:          CLUTTER2.FR
C
C      DEPARTMENT OF ELECTRICAL ENGINEERING    KANSAS STATE UNIVERSITY
C
C      REVISION          DATE          PROGRAMMER
C      -----          ----          -
C      00.0              APR 11, 1983    WAYNE M. BLASI
C
C*****
C
C      PURPOSE
C
C          THIS ROUTINE GENERATES A TWO-SIDED CLUTTER SIGNAL.
C          THE POSITION FUNCTION IS GIVEN IN FIGURE 21.
C
C      ROUTINE(S) CALLED BY THIS ROUTINE
C
C          OPENW
C          WRITR
C
C*****
C
C      Note: THIS ROUTINE GENERATES THE CLUTTER SIGNALS FOR THE
C            SIMULATION RESULTS - CASES 2,3,4.
C
C*****
C
      REAL R(256),IDUT(256),QOUT(256)
      ACCEPT'SAMPLING FREQUENCY : ',FS
      TYPE'THESE QUERIES RELATE TO r(t) FUNCTION'
      ACCEPT'DURATION OF HALF CYCLE? : ',T1
      ACCEPT'AMPLITUDE OF MECHANICAL MOTION? ',A1
      ACCEPT'RISE TIME? : ',TR
      ACCEPT'FALL TIME? : ',TF
      TR=TR/2.0
      TF=TF/2.0
      CALL OPENW(0,'OUTPUT r(t) FILENAME? : ',1024,SIZE)
      CALL OPENW(1,'INPHASE OUTPUT FILENAME? : ',1024,SIZEI)
      CALL OPENW(2,'Q-PHASE OUTPUT FILENAME? : ',1024,SIZEQ)
      ACCEPT'# OF RECORDS (256 PTS/REC)? : ',NUMREC
      TYPE'THESE QUERIES RELATE TO I & Q SIGNALS'
      ACCEPT'AMP. OF POS. CYCLE? : ',AMP1
      ACCEPT'AMP. OF NEG. CYCLE? : ',AMP2
      ACCEPT'PHASE OF OUTPUT SIGNAL? : ',PHA
      PI=3.14159
      TWOPI=2.0*PI
      SLOPER=A1/TR
      SLOPEF=-A1/TF
      T=TR
      T3=0.0
      AIN=0.0
      K=0
      N=1

```

```

DO 999 I=1,NUMREC
DO 100 J=1,256
CONTINUE

5
C
C   GENERATE THE POSITION FUNCTION  $r(t)$ 
C

IF(N.EQ.2)GO TO 30
IF(K.GT.T*FS)GO TO 10
R(J)=AIN+SLOPER*FLOAT(K)/FS
K=K+1
GO TO 99
10 CONTINUE
IF(K.GT.(T1-TF+T3)*FS) GO TO 20
R(J)=A1
K=K+1
GO TO 99
20 CONTINUE
K=0
N=2
30 CONTINUE
IF(K.GT.2.0*TF*FS)GO TO 40
R(J)=A1+SLOPEF*FLOAT(K)/FS
K=K+1
GO TO 99
40 CONTINUE
IF(K.GT.(T1+TF-TR)*FS)GO TO 50
R(J)=-A1
K=K+1
GO TO 99
50 CONTINUE
K=0
N=1
T=2.0*TR
T3=TR
AIN=-A1
GO TO 5
99 CONTINUE
C
C   GENERATE THE CLUTTER SIGNALS
C

ARG=TWOPI*PHA+4.0*PI*R(J)
AMP=AMP1
IF(N.EQ.2)AMP=AMP2
IOUT(J)=AMP*COS(ARG)
QOUT(J)=AMP*SIN(ARG)
100 CONTINUE
CALL WRITR(0,I,R,1,IERR)
CALL WRITR(1,I,IOUT,1,IER)
CALL WRITR(2,I,QOUT,1,IR)
999 CONTINUE
CALL RESET
STOP***NORMAL TERMINATION***
END

```

AN APPLICATION OF ADAPTIVE
COMPLEX PREDICTION

BY

WAYNE MICHAEL BLASI
B.S., Kansas State University, 1981

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1983

ABSTRACT

This work introduces the notion of employing adaptive complex prediction to improve target detection in a class of radar clutter signals due to an oscillating source. The moving target is assumed to produce relatively low amplitude transient signals that are typically broadband. The motion of the target is assumed to be in one direction -- i.e., towards or away from the radar. Computer simulations are presented which illustrate the effectiveness of adaptive complex prediction for improved target detection.