Transient Scattering Response of an Airborne Target

J. H. Schukantz, Jr. D. W. S. Tam J. B. McGee L. B. Koyama J. W. Rockway S. T. Li

Technical Document 2796
April 1995

Naval Command, Control and
Ocean Surveillance Center
RDT&E Division

San Diego, CA
92152-5001

Technical Document 2796
April 1995

Transient Scattering Response of an Airborne Target

J. H. Schukantz, Jr.
D. W. S. Tam
J. B. McGee
L. B. Koyama
J. W. Rockway
S. T. Li

NAVAL COMMAND, CONTROL AND OCEAN SURVEILLANCE CENTER RDT&E DIVISION
San Diego, California 92152–5001

K. E. EVANS, CAPT, USN
Commanding Officer

R. T. SHEARER
Executive Director

ADMINISTRATIVE INFORMATION

This document is submitted in partial fulfillment of the Theater Surveillance Project (RU11G54) of the Ocean Surveillance Technology Program (program element 0602111N). The work described herein was sponsored by the Office of Naval Research (ONR 313). It was performed by the Naval Command, Control and Ocean Surveillance Center, RDT&E Division, Applied Electromagnetics Branch, San Diego, CA 92152–5001.

Released by
L. C. Russell, Head
Applied Electromagnetics
Branch

Under authority of
R. J. Kochanski, Head
Communications Systems
Engineering and Integration
Division

SB

CONTENTS

EXECUTIVE SUMMARY iii

INTRODUCTION 1

TARGET ID MEASUREMENTS 3

RESULTS 7

DISCUSSION 8

REFERENCES 9

APPENDIX A: TIME DOMAIN MEASUREMENT RANGE (TDMR) EQUIPMENT AND PROCEDURE A-1

APPENDIX B: RADAR CROSS SECTION THEORY AND CALCULATIONS B-1

APPENDIX C: RCS PROGRAM SUMMARY AND LISTING C-1

APPENDIX D: NRAD RADAR CROSS SECTION PROGRAM D-1

FIGURES

1. NRaD time domain measurement range 2

2. Sketch of a target at (a) 0° and (b) 45° rotation 3

3. Transient scattering reponse measurements for target rotation of 0° 4

4. Transient scattering response measuerements for target rotation of 45° 5

5. Results of transient scattering measurements: 0° rotation 6

6. Results of transient scattering measurements: 45° 6

7. RCS of noise and of target 7

8. Comparison of measured of NEC-calculated RCS results 8

A-1. TDMR measurement equipment A-1

B-1. Circuit diagram with BWS as transmitting antenna B-1

B-2. Circuit diagram of BWS as receive antenna B-2

EXECUTIVE SUMMARY

OBJECTIVE

The research team performed measurements in a bounded wave simulator to obtain the radar cross section (RCS) of a scale model of a particular airborne target. Measurements of the airborne target were obtained from two different flying orientations. The transient scattering response data were scaled to allow comparison with the RCS obtained by numerical methods for a full-size target. This document also presents the measurement procedure and data processing techniques that were used by the research team.

METHOD

A target was placed within the bounded wave simulator (BWS) area and was hit with an electromagnetic impulse, returning the scattered signal through the BWS for measurement in the control room. Measurements were also made of the background signal, as well as the incident signal. As the signals return to the control room below the BWS structure, the waveforms are captured on an oscilloscope, then processed using Tektronix's Signal Processing and Display Package and Radar Cross Section program developed at NRaD. These programs convert the waveforms from time domain representations to frequency through the Fast Fourier Transform process. Further waveform processing allowed the calculation of RCS, which is the basis for target identification efforts using resonant frequencies. The measurement procedure and signal processing techniques are discussed in detail in the appendices.

RESULTS

The RCS of the scale model of an airborne target from two different flying orientations was fairly independent of "flying" orientation, or rotation about its central axis. The research team established the frequency range for which these measurements are valid by determining the point at which the noise overcomes the signal reflected by the target, i.e., where the RCS of the noise spectrum reaches the level of the target RCS. A test was performed to determine how well the RCS data obtained by the scale model measurements agreed with those predicted by numerical analysis using the Numerical Electromagnetic Code (NEC). At low frequencies, the computed curve rose steeply to cross the measured curve at about 12 MHz. Between 13 and 60 MHz, the computed curve followed the shape of the scale model measurements, but exceeded it by approximately 5 dB.

INTRODUCTION

NRaD's Time Domain Measurement Range, a transient electromagnetic test facility, was originally built for measuring the effects of a simulated high-altitude electromagnetic pulse on shipboard antenna systems using scale brass models of Navy ships [1]. The Time Domain Range's bounded wave simulator (BWS) has been expanded and can now obtain the scattering response of an airborne target due to a transient pulse [2, 3]. Figure 1 shows an NRaD Time Domain Range BWS diagram.

A target was placed inside the BWS and was hit with an electromagnetic impulse. The scattered signal returned through the BWS for measurement in the control room. The background signals were also measured. As the signals returned to the control room below the BWS structure, the waveforms were captured on an oscilloscope. The research team then used the Tektronix Signal Processing and Display Package and NRaD's Radar Cross Section program to process the signals. These programs use the fast Fourier transform to convert the wave forms from time domain representations to frequency. Further waveform processing was used to calculate the radar cross section (RCS), which is the basis for target identification efforts using resonant frequencies. The appendices discuss the measurement procedure and signal processing techniques in detail.

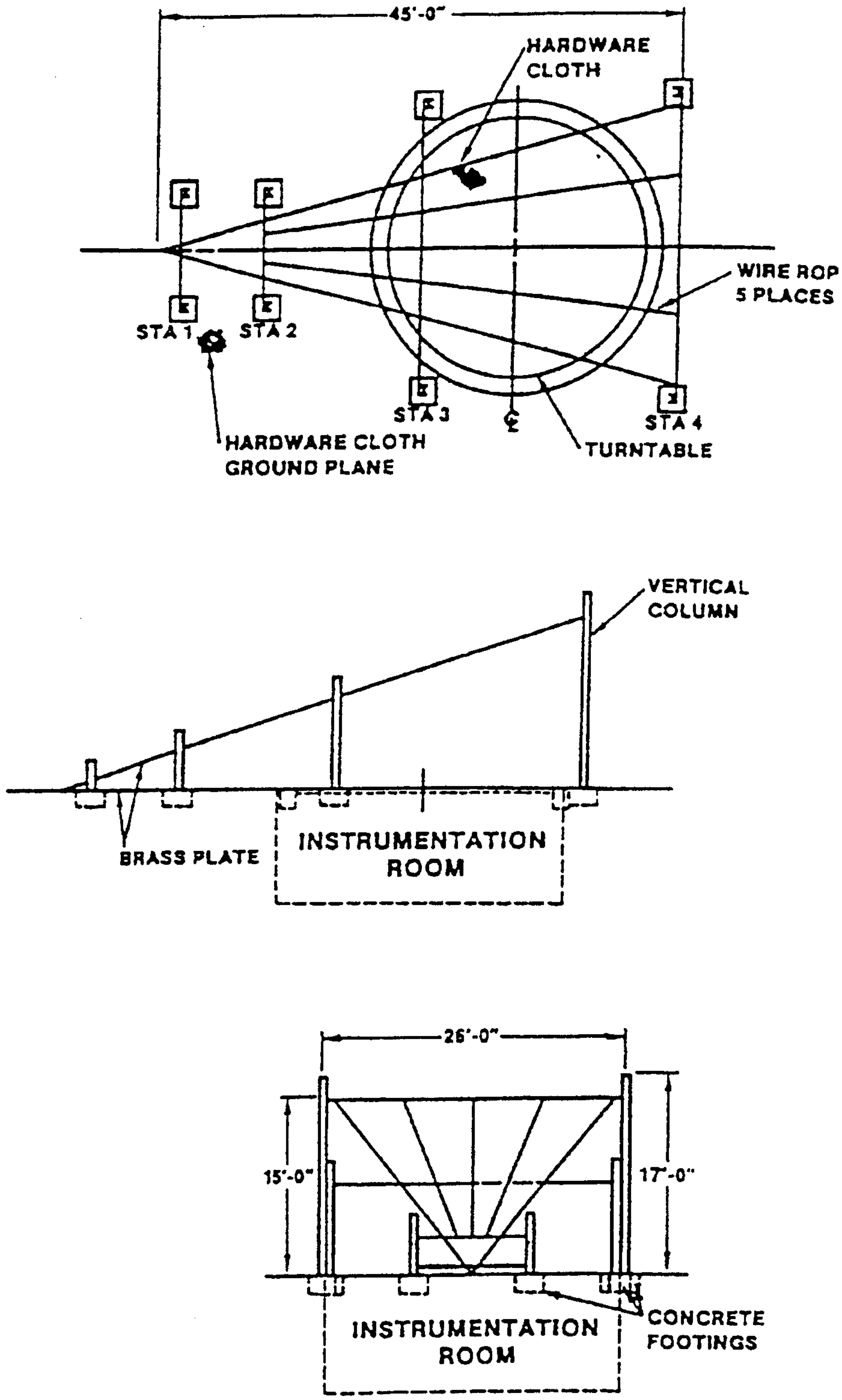
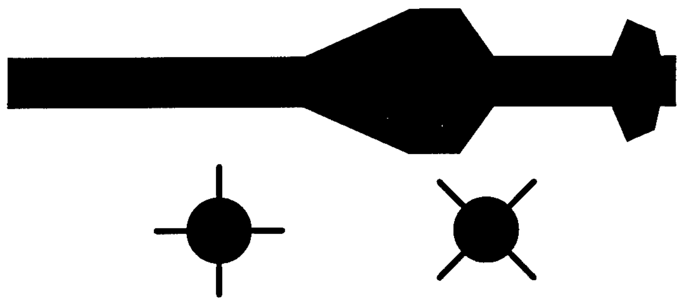


Figure 1. NRaD Time Domain Measurement Range.

TARGET ID MEASUREMENTS

The BWS structure creates some "background" reflections even when no target is in place. Therefore, the response (due to the target alone) was isolated by subtracting the background waveform from the (target + background) waveform. The isolated target signal and the incident waveform were then input into a program that calculates and scales the RCS to full size.

The target used for these runs (see figure 2) was a 1:2.93 scale brass model of a previously tested full-size target. For the first measurements, the target was suspended in the center of the BWS with its fins at a 0-degree rotation. The measurements were then repeated with the fins at 45 degrees.



(a) (b)

Figure 2. Sketch of target at (a) 0° and (b) 45° rotation.

The research team took measurements at each orientation:

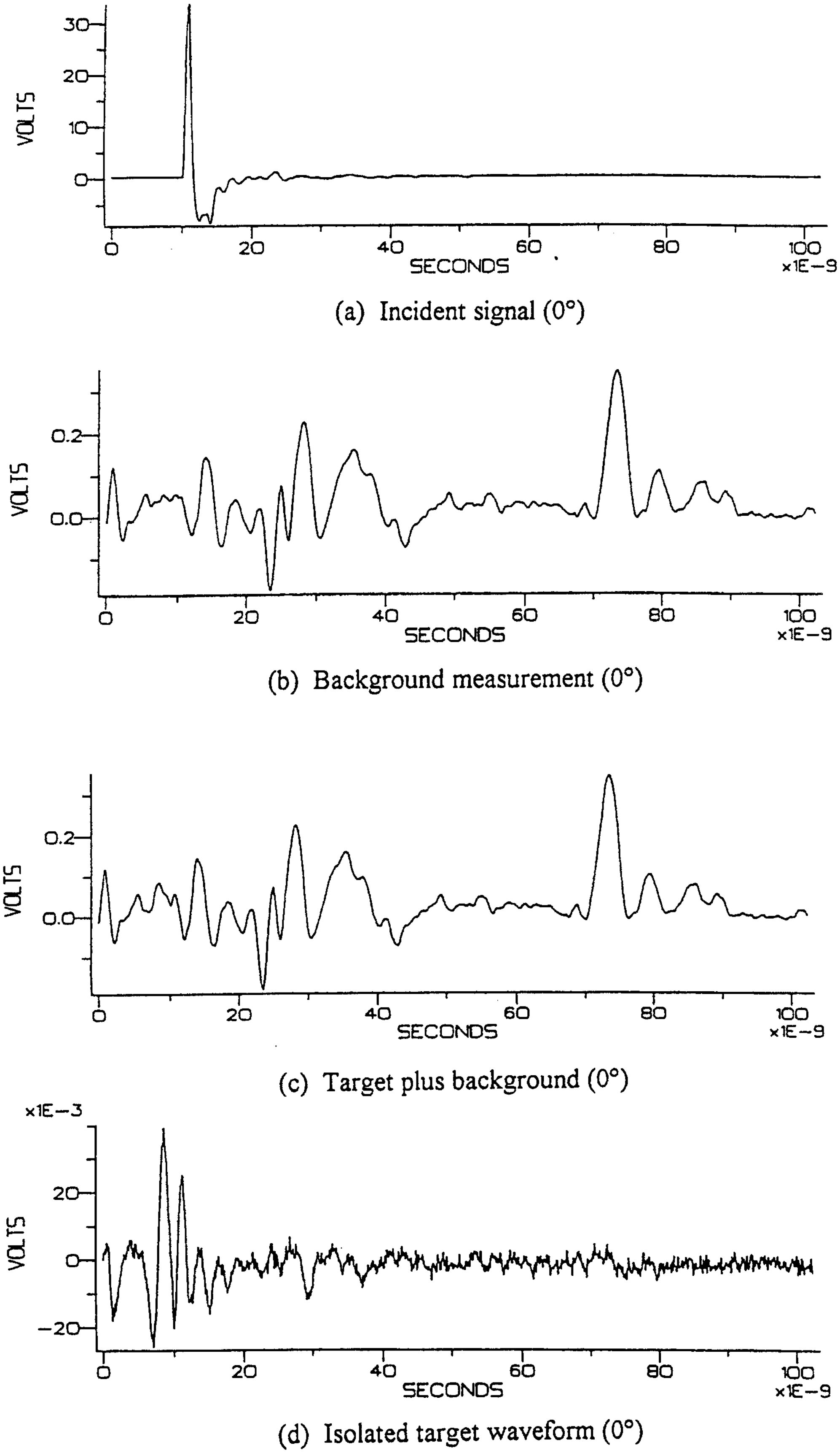
1. Background waveform

2. Target plus background waveform

3. Incident waveform

Figure 3 shows the waveform obtained for the 0-degree rotation. Figure 3(a) displays the incident waveform. The background measurement appears in Figure 3(b). Figure 3(c) shows the target (including background) measurement. The "target only" waveform in figure 3(d) was obtained by subtracting the two previous waveforms. Similarly, figures 4(a) through 4(d) display the same information for a target rotated by 45 degrees.

"Target only" waveforms were time-gated, or zeroed out before and after the appearance of the target-scattered signal. This approach prevented excess noise from entering the calculations for the RCS plots. Figure 5 displays the gated-target waveform and its corresponding RCS pattern for the 0-degree case. Figure 6 shows the same information for the 45-degree case.



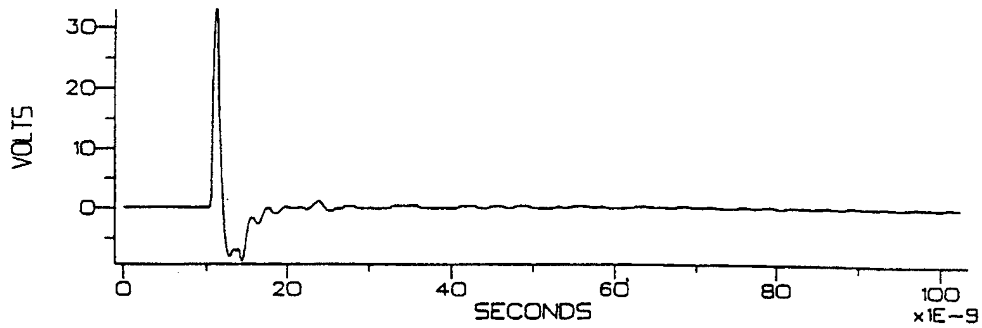
(a) Incident signal (0°)

(b) Background measurement (0°)

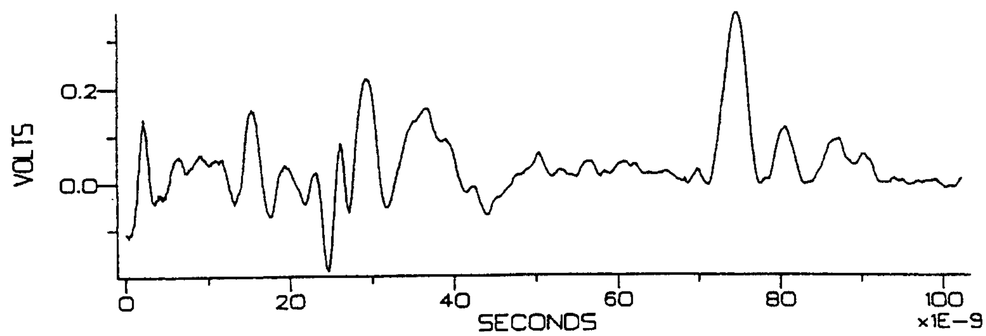
(c) Target plus background (0°)

(d) Isolated target waveform (0°)

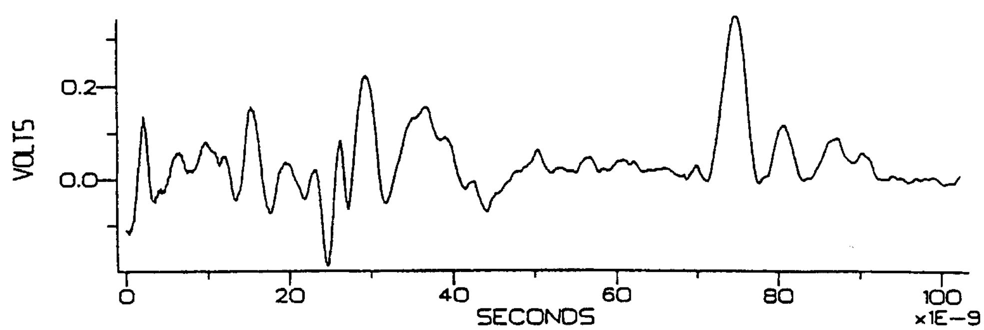
Figure 3. Transient scattering response measurements for target rotation of 0°.



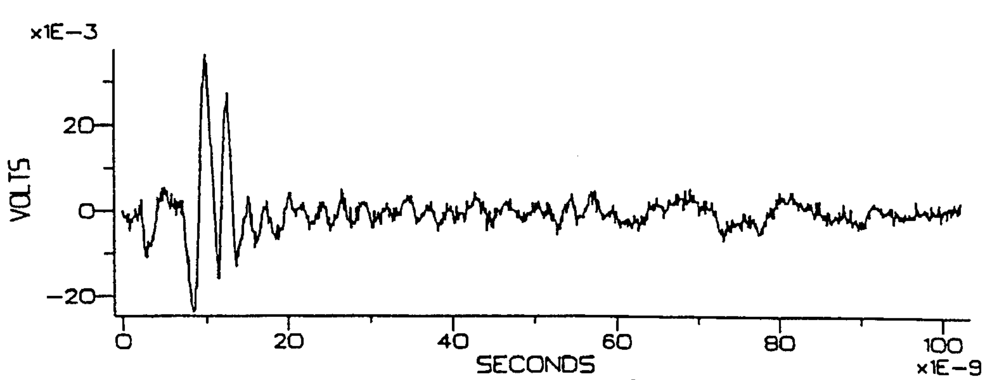
(a) Incident signal (45°)



(b) Background measurement (45°)

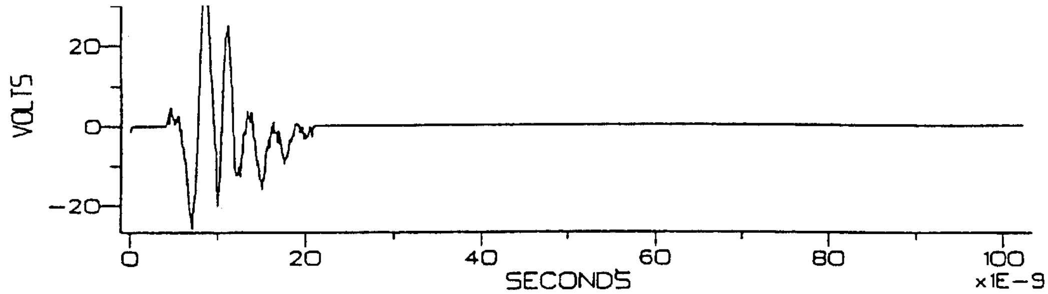


(c) Target plus background (45°)

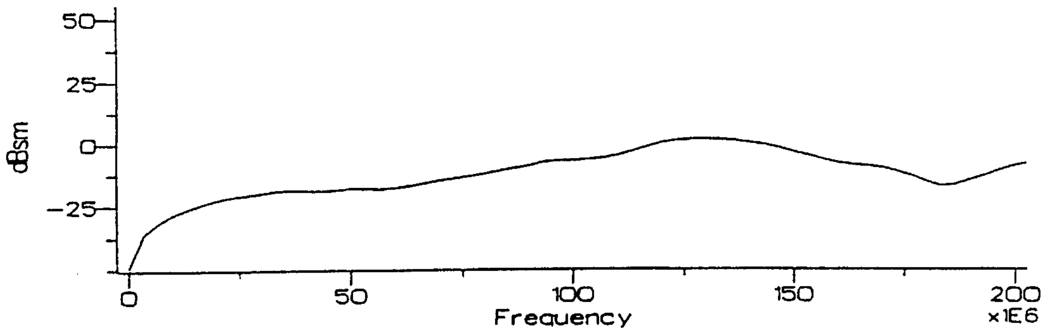


(d) Isolated target waveform (45°)

Figure 4. Transient scattering response measurements for target rotation of 45°.

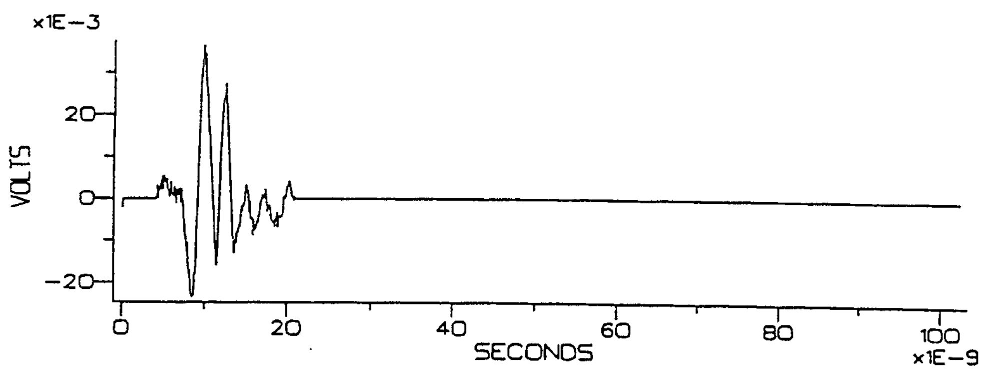


(a) Time-gated target signal (0°)

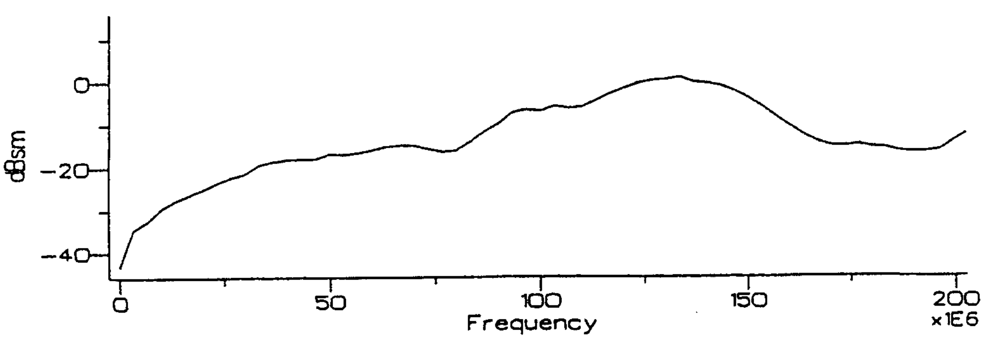


(b) RCS of gated target signal (0°)

Figure 5. Results of transient scattering measurements: 0° rotation.



(a) Time-gated target signal (45°)



(b) RCS of gated target signal (45°)

Figure 6. Results of transient scattering measurements: 45° rotation.

RESULTS

A comparison of figure 5(b) and 6(b) reveals that this target's RCS is fairly independent of "flying" orientation, or rotation about its central axis.

The research team established the valid frequency range for these measurements by determining the point at which the noise overcomes the signal reflected by the target, i.e., where the RCS of the noise spectrum reaches the level of the target RCS. The noise measurement chosen for this example was a subtraction of two background measurements taken moments apart. The difference between these background waveforms represents the "noise" that might remain after a [(target + background) - background] subtraction is performed.

Figure 7 superimposes the RCS of the noise and of the target. Around 300 MHz (full scale), these levels become equal, indicating the upper frequency limit of the RCS measurement.

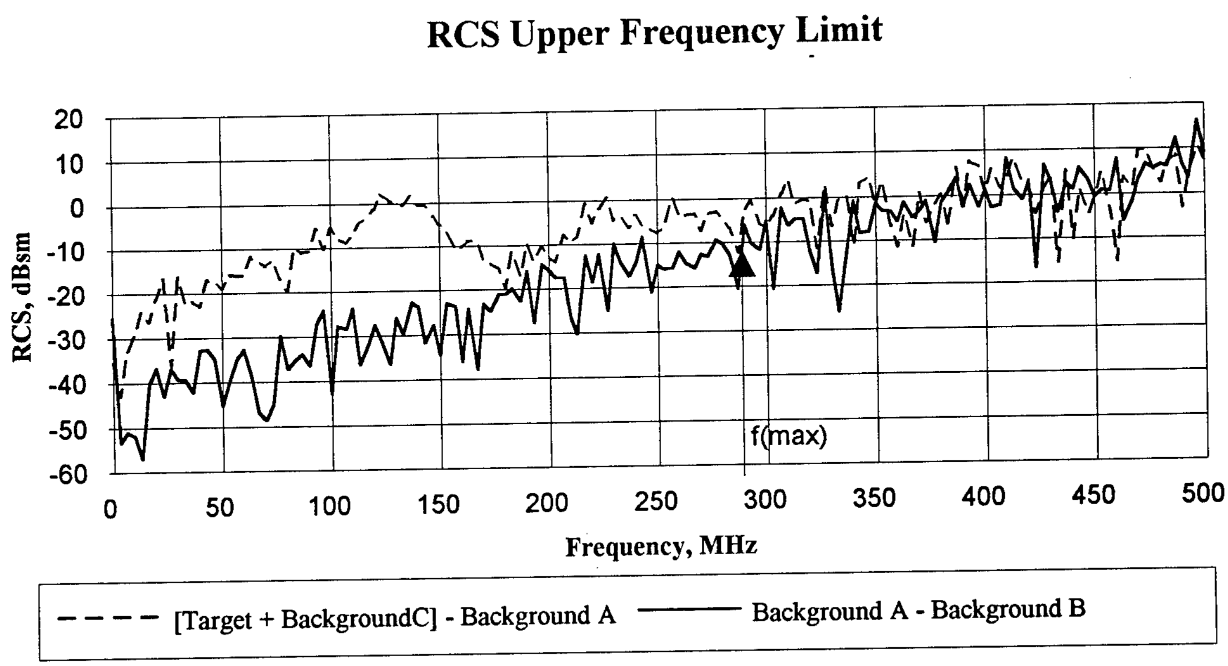


Figure 7. RCS of noise and of target.

Figure 8 compares plots of measured and Numerical Electromagnetic Code (NEC) [4, 5] RCS test results. The measured data in figure 8 comes from the 45-degree fin rotation. At low frequencies, the computed curve rises steeply to cross the measured curve at about 12 MHz. Between 13 and 60 MHz, the computed curve follows the shape of the scale model measurements, but exceeds it by approximately 5 dB.

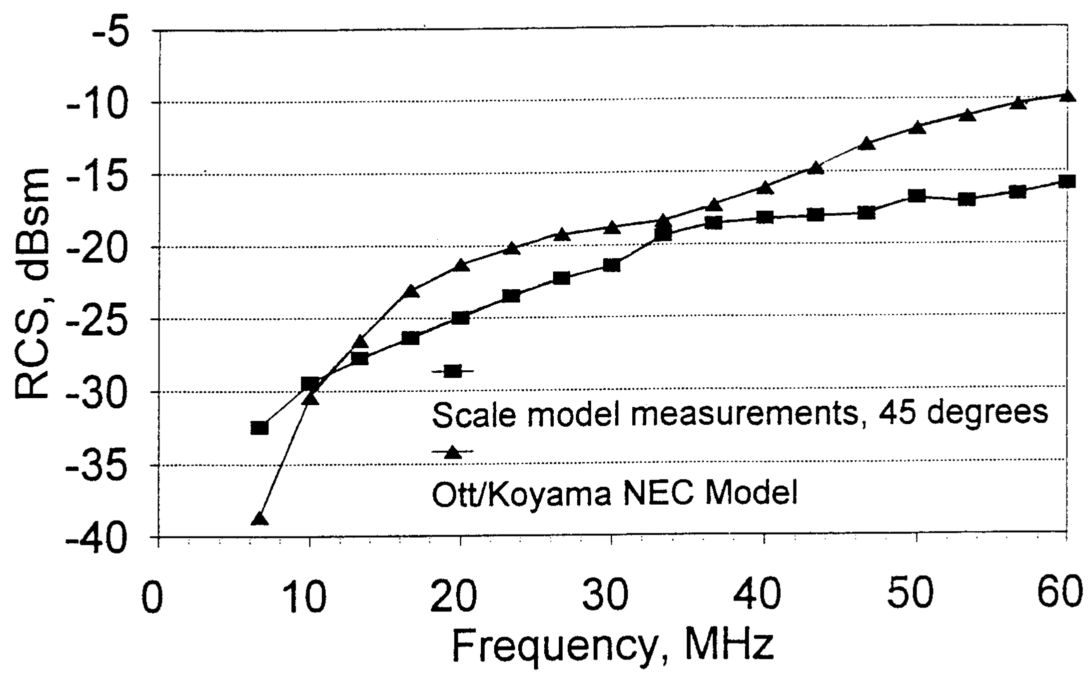


Figure 8. Comparison of Measured and NEC-Calculated RCS Results.

DISCUSSION

The data presented in this report, consisting of one test at a 0-degree fin orientation and one test at a 45-degree fin orientation, were selected from several tests that were unusable due to high levels of background noise. One source of difficulties with these measurements is the increase in noise that occurs in the afternoons, preventing a good subtraction of background waveforms. This increase may be caused by several factors. Winds may have vibrated the BWS structure, the temperature may have risen, or there may have been increased transmissions from external sources. Monitoring background noise level over several days might help establish the best time for low-interference testing. When testing, it is important to obtain a clean background - background subtraction to establish validity.

RCS measurements on the full-size target were performed on the Time Domain Measurement Range, but were partly affected by target and range interaction. The equation for calculating a target's RCS is taken as the distance to the target (R) approaches infinity. The incident field strength must be uniform over the length of the target. A large target within the confines of the BWS did not entirely meet this requirement. However, the scale model in this study was sufficiently small with respect to the BWS to provide acceptable results.

Further studies would be necessary to fully characterize the limits of allowable target size for accurate measurements.

Other topics for further study include RCS dependency on target orientation. Rotation about the central axis had little effect on the RCS of this target, but positioning has influenced the RCS results of past measurements and could factor into this case as well. Finally, a good demonstration of the target identification capabilities of the NRaD BWS system would be to perform RCS measurements on other targets of similar size and shape to attempt to distinguish between them. This has been successfully performed with model ships as targets [6, 7].

REFERENCES

Burke, G. J. and A. J. Poggio, "Numerical Electromagnetics Code (NEC) - Method of Moments", NOSC Technical Document 116, January 1981. Naval Ocean Systems Center, San Diego, CA.

Burke, G. J., "Numerical Electromagnetics Code - NEC-4, Method of Moments, Part 1: User's Manual", Lawrence Livermore National Laboratory UCRL-MA-109338, January 1992.

Deneris, C. A., J. W. Rockway, J. H. Schukantz, Jr., "Transient Scattering Response of Several Canonical Ship Targets", NRaD Technical Document 2408, February 1993. Naval Command, Control, and Ocean Surveillance Center RDT&E Division, San Diego, CA.

Li, S. T., J. C. Logan, J. W. Rockway, J. H. Schukantz, D. W. S. Tam, L. C. Martin, and E. J. Bogdan, "Shipboard Antenna Systems EMP Design Algorithm (EMPAL)", The IEEE 1983 International Symposium on EMC, August 23-25, 1983, Arlington, VA.

Li, S. T., J. C. Logan, A. G. Page, J. W. Rockway, J. H, Schukantz, and D. W. S. Tam, "Preliminary Report on the Transient Scattering Response of Several Airborne Targets", NOSC Technical Note 1643, December 1990. Naval Ocean Systems Center, San Diego, CA.

Page, A. G., C. A. Deneris, J. W. Rockway, and J. H. Schukantz, Jr., "Transient Scattering Response of Several Airborne Targets", NRaD Technical Document 2249, March 1992. Naval Command, Control, and Ocean Surveillance Center RDT&E Division, San Diego, CA.

Thowless, E. A., S. T. Li, L. B. Koyama, J. H. Schukantz, Jr., H. W. Guyader, D. W. S. Tam, J. C. Logan, "Target Identification by Use of Reflections of an Electromagnetic Impulse", NOSC Technical Note 1594, September 1989. Naval Ocean Systems Center, San Diego, CA.

APPENDIX A:

TIME DOMAIN MEASUREMENT RANGE (TDMR) EQUIPMENT AND PROCEDURE EQUIPMENT

Tektronix 11802 Oscilloscope

Pulse Generator (Picosecond Pulse Labs)

High Voltage Pulse Source (Grant Physics Associates)

Directional Coupler

Bounded Wave Simulator

Target

See diagram for equipment connections.

Note that the Grant High Voltage Pulse Source is externally triggered by the Picosecond pulse generator.

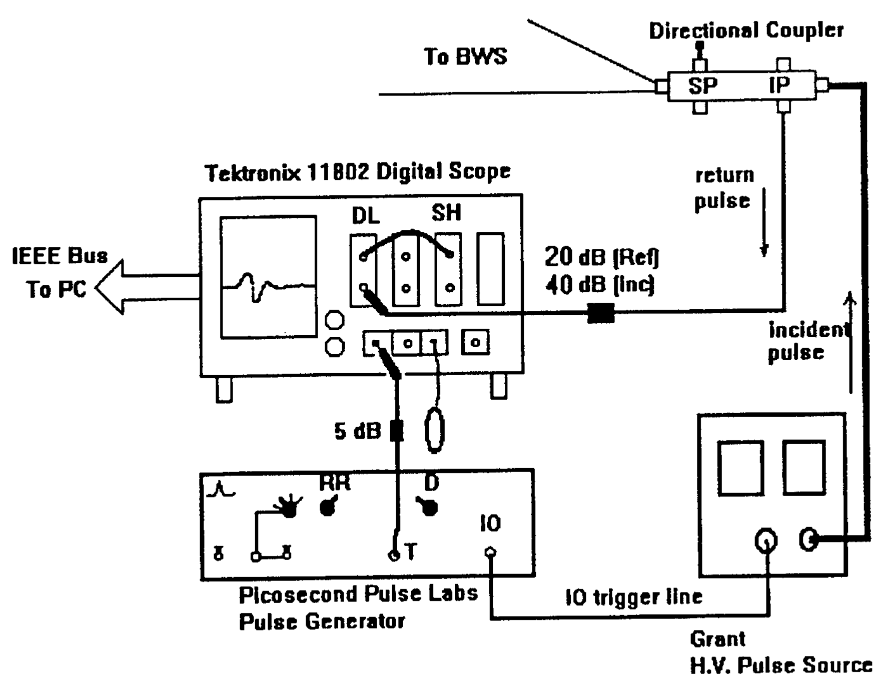


Figure A-1. TDMR measurement equipment.

PROCEDURE

BACKGROUND MEASUREMENT:

1. Allow equipment to warm up before beginning measurements (1 hour).

2. Since the pulse life of the Grant pulser is limited, it is best not to apply the external trigger until measurements are to be made. To disable triggering, disconnect the cable from the impulse output (**IO** on diagram) of the Picosecond pulse generator.

3. Wearing grounding wrist strap, place 20-dB attenuator at the delay line input to the oscilloscope.

4. Connect coaxial cable to scattering port (**SP** on diagram) of directional coupler, and 50-ohm load to incident port (**IP**). Check that the feed to the BWS is securely soldered to the coupler.

5. When ready to perform measurements, connect the IO trigger line from the Picosecond pulse generator.

6. Obtain the reflected waveform on the oscilloscope in continuous mode. When adjustments to scale have been made, take an average of 32 waveforms using 1024 sample points.

   Press "Acquire Desc"

   "Average On" (NAve = 32)

   "Average Complete"

7. Read the waveform into the computer using the SPD software. Save to disk.

8. Repeat measurement and perform waveform subtraction until measurements are stable and repeatable.

BACKGROUND PLUS TARGET MEASUREMENTS:

1. Disconnect external triggering by unplugging IO line from Picosecond pulse generator.

2. Place (or suspend) target in BWS. Check that return cable is still connected to scattering port of directional coupler.

3. Check that 20-dB attenuator is still in place on input to scope's delay line.

4. Reconnect IO trigger line to Picosecond pulse generator.

5. Obtain waveform on scope, and take average of 32 traces using 1024 sample points.

6. Read waveform into the computer, then write to disk.

7. Repeat measurement. Perform waveform subtraction on SPD in order to check for stability.

8. When good subtraction is obtained, take one more measurement with target, then quickly remove target and take a background measurement. After this, make incident wave measurements.

INCIDENT WAVE MEASUREMENTS:

1. Disconnect IO trigger.

2. With grounding wrist strap, increase attenuation on oscilloscope delay line input to 40 dB.

3. Change the return signal cable from the scattering port to the incident port on the directional coupler. Place 50 ohm load on the scattering port.

4. Reconnect IO trigger line from the Picosecond pulse generator.

5. Obtain waveform on scope in continuous mode, then take average of 32 waveforms using 1024 samples. Write waveform to computer disk.

6. Repeat measurement. Perform waveform processing.

WAVEFORM PROCESSING:

1. Type SPDMENU to begin program.

2. To read waveforms from scope into program :

    [2] Input/Output

    [6] Acquire waveform

    [7] Tektronix 11802 scope

    Name waveforms consecutively, using current date.

    [F2] Run

Then write waveforms from program to disk:

    [F4], **[F4]** to escape to previous menu

[2] Input/Output

[2] Write waveform to disk

[F2] Run

3. To plot waveforms:

[7] Display

[1] Graph

Enter waveform names and scale as necessary.

[F6] Go to next page

Change from screen to plotter.

Change from blue background to black.

[F2] Run

4. To process waveforms:

The magnitude of the waveforms must be scaled up in order to cancel the effect of the attenuators on the oscilloscope input line.

Multiply background waveform by **10.**

Multiply (background + target) waveform by **10.**

Multiply incident waveform by **100.**

Save the scaled waveforms to disk.

5. Run the RCS program using re-scaled waveforms.

APPENDIX B:

RADAR CROSS SECTION THEORY AND CALCULATIONS

Following the development of References [3] and [7], we define the radar cross section, or RCS, of a target by the following equation:

$$\sigma = \lim_{R\to\infty} 4\pi R^2 \left|\frac{E_s}{E_i}\right|^2, \tag{1}$$

where E_S represents the magnitude of the field scattered by the target, E_i is the incident field strength, and R is the horizontal distance from the feedpoint of the BWS to the front of the target. We wish to rewrite the RCS equation in terms of quantities that can be measured on the TDMR, in particular the incident and scattered voltages.

The incident electric field is simply equal to the incident voltage divided by the height of the BWS at the target. Since the BWS has a slope of 19 degrees, the incident field may be rewritten as:

$$E_i = \frac{V_i}{h} = \frac{V_i}{R\tan 19^\circ} = \frac{V_i}{0.344\,R}. \tag{2}$$

When the pulse generator produces an incident voltage pulse, the BWS may be viewed as a transmitting antenna (figure B-1). It has an intrinsic 100-ohm impedance (R_{ant}), and a 100-ohm resistor is placed across the feed of the BWS to provide a match to the 50-ohm impedance of the source.

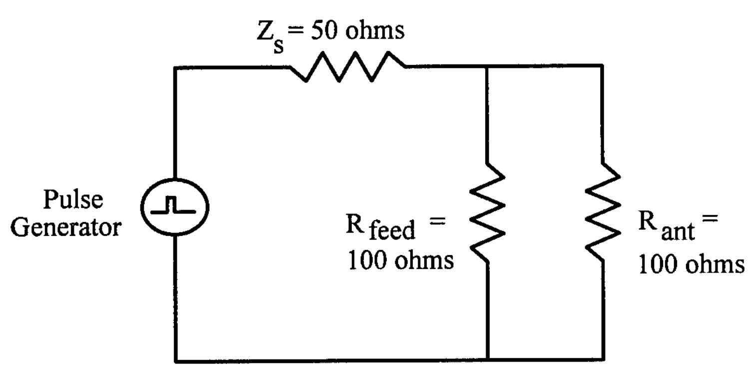


Figure B-1. Circuit diagram with BWS as transmitting antenna.

When the pulse reaches the BWS, the target reflects part of the incident field, so that the BWS now acts as a receiving antenna. The pulse generator pictured above is no longer transmitting, but instead the reflected voltage at the BWS becomes the voltage source, as diagrammed in figure B-2.

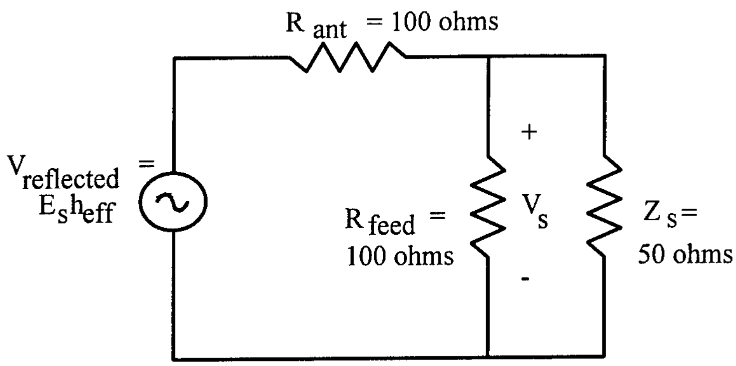


Figure B-2. Circuit diagram of BWS as receive antenna.

The scattered electric field is related to the measured scattered voltage by the effective height, h_{eff}, of the BWS as a receive antenna. The feed resistor and the input resistance of the receiver combine in parallel to form a load resistance of $R_L = 33$ ohms. The measured scattered voltage V_s is therefore related to the scattered field strength E_s as given below:

$$E_s = \frac{R_{ant} + R_L}{R_L} \frac{V_s}{h_{eff}}, \ or \tag{3}$$

$$E_s = \frac{100+33}{33} \frac{V_s}{h_{eff}}. \tag{4}$$

Substituting expressions (2) and (3) for E_s and E_i into the RCS equation provides a basis for obtaining a target's radar cross section based on the measured values of V_s, V_i, and R:

$$\sigma = 1.92 \frac{4\pi R^4}{h_{eff}^2} \frac{V_s^2}{V_i^2}. \tag{5}$$

This expression still requires that a value for the effective height of the bounded wave simulator be determined. Effective height* may be given in terms of effective aperture A_e, antenna impedance $R_{ant} + jX_{ant}$, and terminating impedance $R_L + jX_L$, as

$$h_{eff} = \sqrt{\frac{A_e\left[(R_{ant}+R_L)^2 + (X_{ant}+X_L)^2\right]}{120\pi R_L}}. \tag{6}$$

Since $X_{ant} = X_L = 0$, this becomes

$$h_{eff} = \sqrt{\frac{(100+33)^2}{(377)(33)} A_e} = 1.19\sqrt{A_e}. \tag{7}$$

---

* Krause, J. D., *Antennas*, McGraw-Hill Book Company, Inc., New York. 1950.

The relationship between effective aperture and the gain of an antenna is

$$A_e = \frac{\lambda^2}{4\pi} G_r \,, \tag{8}$$

where G_r is the receive mode gain of the BWS, and λ is the wavelength. Substituting equation (8) into (7), we obtain:

$$h_{eff} = 1.19 \sqrt{\frac{G_r \lambda^2}{4\pi}} \,. \tag{9}$$

The gain of the BWS as used in the scattering measurements is difficult to determine analytically because the target is within the BWS canopy. However, the gain can be accurately determined empirically by the substitution method. In this method, the field radiated by a standard antenna, in this case a quarter-wave monopole, is compared to the field produced by the BWS. Since the gain of the standard is known, the BWS gain may be inferred. The procedure is to start with the lowest frequency of interest and work towards the higher frequencies in convenient steps. At each new frequency, the monopole is clipped so that its length remains at the quarter wavelength. A number of measurements were made at discrete frequencies over the band of interest. For the BWS, it was found that the experimental gain was not independent of frequency. However, the maximum variation over the measurement range from 200 to 460 MHz was less than 0.1 dB. Thus, it was convenient to use the average gain, which was 16.17. Finally, using the measured BWS average gain of 16.17, the relationship for RCS given in equation (5) can be rewritten in terms of the measured voltages:

$$\sigma = \frac{4\pi R^4}{(1.19)^2 (16.17) \frac{\lambda^2}{4\pi}} (1.92) \frac{V_s^2}{V_i^2}, \text{ or} \tag{10}$$

$$\sigma = (1.05) \frac{4\pi R^4}{\lambda^2} \frac{V_s^2}{V_i^2} . \tag{11}$$

Note that this equation represents the RCS of the measured (scale model) target. Scaling must be performed along the frequency axis as well as on the magnitude of the RCS function in order obtain an RCS that corresponds to the full-scale target.

In comparison to the full-size object, a model's dimensions are scaled down by a known scale factor (SF). Typically, the brass models used at the TDMR have a scale factor of 48, but for this study a scale factor of 2.93 was used. The dimensions are related by the following:

$$L_{\text{full-scale}} = SF * L_{\text{model}}. \tag{12}$$

Because frequency is inversely proportional to wavelength, the full-scale frequencies will be lower than the corresponding model frequencies by the scale factor:

$$f_{\text{full-scale}} = \frac{1}{SF} f_{\text{model}}. \tag{13}$$

Finally, since radar cross section dimensions are in units of area (square meters), the magnitude of the RCS function must be scaled by the square of the scale factor

$$\sigma_{\text{full-scale}} = SF^2 * \sigma_{\text{model}}. \tag{14}$$

RCS is often displayed in dBsm by taking 10 log σ.

APPENDIX C:

RCS PROGRAM SUMMARY AND LISTING

A program has been written that automatically calculates and scales the RCS using the waveforms obtained from the TDMR measurements. This program utilizes subroutines of the Tektronix SPD (Signal Processing and Display) package. A brief summary of the program steps is listed below, and the actual code follows.

1. Read in the background waveform.

2. Read in the (target + background) waveform.

3. Perform waveform subtraction, saving to file "target.wav"
   [(target + background) - background = target]

4. Plot target waveform to screen.

5. Allow user to zero out part of target waveform. This gating function allows excess noise to be removed.

6. Read in incident waveform.

7. Perform FFT on target waveform, creating real and imaginary parts.

8. Convert to polar form, creating magnitude and phase waveforms.

9. Perform FFT on incident waveform, creating real and imaginary parts.

10. Convert to polar form.

11. Prompt user to enter distance to target in feet.

12. Divide magnitude of $V_s(f)$ by $V_i(f)$.

13. Square this ratio.

14. Scale by 4 pi * [$(12\ R/39.37)^4$] * 1.05 * SF^2. This factor of 12/39.37 converts feet into meters.

15. Create a waveform for wavelength (λ) values as a function of frequency.

16. Divide current expression by λ^2, resulting in:

$$\frac{4\pi\, R(meters)^4 (1.05)(SF)^2}{\lambda^2} \frac{V_s^2}{V_i^2}$$

17. Scale along the frequency axis by dividing by SF.

18. Take 10 log (current expression).

19. Write final RCS waveform to "logrcs.wav", print to screen.

APPENDIX D:

NRaD RADAR CROSS SECTION PROGRAM

```
REM *** Program Name:  RCS3B.BAS
REM *** Program modified from RCS4C1.BAS by Jodi McGee/Daniel Tam on Sep 16, 1994
REM *** RCS4C1.BAS  by Daniel Tam
REM *** Program RCS for Cylinder Calibration using Jim Schukantz's equation
REM *** JUL 18, 1990
REM ***
REM ***  -- Modified by Arch Page Sept 14, 1993
REM ***  -- Modified again by Arch Page March 17, 1994
REM ***  -- Modified by Jodi McGee / Daniel Tam Sept 16, 1994 for scale model target
REM ***  -- Modified scale factor is SF as a user input

REM ***  -- Microsoft Quick Basic 4.5 and Tektronix Signal Processing and Display (SPD)
REM ***  -- Programs V3.0

' $INCLUDE: 'tekspd.bi'

'********************** INITIALIZATION ***************************

'Enable floating point exception handling.
        CALL benablfpe
        CALL berrfile("rcs1.err", status%)

'Reserve memory for waveforms
        dummy& = SETMEM(-150000)

'************* WAVEFORM CREATION AND PROCESSING ******************

'Create a waveform data structure with space for a one tuple, one
'dimension of single-precision type data.

        INPUT "enter number of sample points (1024 default) "; Length$
        IF Length$ = "" THEN Length& = 1024 ELSE Length& = VAL(Length$)
        background% = 1
        CALL bcreate1wf(Length&, WFloat%, background%, status%)
        IF status% <> 0 THEN
                CALL brpterr(status%, "ERROR: waveform creation failed "): END
        END IF
10      INPUT "ENTER BACKGROUND FILE NAME (default = BK.WAV) "; F1$
        IF F1$ = "" THEN F1$ = "BK.WAV"
```

```
Length = LEN(F1$)           ' 8 lines added by Arch Page
dotpos = INSTR(F1$, ".")     ' to auto add the ".WAV" ext to filename
IF dotpos = 0 THEN
        F1$ = F1$ + ".WAV"
      ELSE
        F1$ = LEFT$(F1$, dotpos)
        F1$ = F1$ + "WAV"

END IF

CALL bafiletowf(F1$, background%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "error: converting from ascii file failed "): GOTO 10
END IF
tarbck% = 2
CALL bcreate1wf(Length&, WFloat%, tarbck%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "ERROR: waveform creation failed "): END
END IF
20   INPUT "ENTER TARGET+BACKGROUND FILE NAME (default = TGT.WAV) "; F2$
IF F2$ = "" THEN F2$ = "TGT.WAV"
Length = LEN(F2$)           ' 8 lines added by Arch Page
dotpos = INSTR(F2$, ".")     ' to auto add the ".WAV" ext to filename
IF dotpos = 0 THEN
        F2$ = F2$ + ".WAV"
ELSE
        F2$ = LEFT$(F2$, dotpos)
        F2$ = F2$ + "WAV"
END IF
CALL bafiletowf(F2$, tarbck%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "error: converting from ascii file failed "): GOTO 20
END IF
target% = 3
CALL bcreate1wf(Length&, WFloat%, target%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "ERROR: waveform creation failed "): END
END IF
PRINT "TARGET+BACKGROUND - BACKGROUND SUBTRACTION"
CALL bwsub(tarbck%, background%, target%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "ERROR: SUBRTACTION FAILEED"): END
END IF
'set units and scaling for waveform rcs%
CALL btupunits(tarbck%, 0, in1units$, in1len%, status%)
```

```
        CALL bwftunits(in1units$, 0, target%, status%)
        CALL bdimunits(tarbck%, 0, inHunits$, inHlen%, status%)
        CALL bwfdunits(inHunits$, 0, target%, status%)
        CALL brdimscale(tarbck%, 0, hsf#, status%)
        CALL bwdimscale(target%, 0, hsf#, status%)
        ntarget% = 18
        CALL bcopywf(target%, ntarget%, status%)
        leads = 1
100     REM
        CALL bgrinit(status%)
        CALL bgrdev("DISPLAY", status%)
        setno% = 1
        CALL bgrdefw(setno%, ntarget%, status%)
        llength# = Length&
        CALL bgrdefa(0, 0, llength#, status%)
        CALL bgrcurv(1, 0, 1, WHITE%, SOLID%, status%)

        CALL bgrvwpt(0, 16385, 32767, 32767, status%)
        CALL bgrdsply(status%)
        IF (leads = 0 AND leade = 0 AND tails = 0 AND taile = 0) THEN 200
        FOR i& = 0 TO lenght&
                CALL breadnum(ntarget%, 0, i&, value#, status%)
                CALL bwritenum(value#, 0, i&, target%, status%)
        NEXT i&
        FOR i& = leads TO leade
                CALL bwritenum(0#, 0, i&, target%, status%)
        NEXT i&
        FOR i& = tails TO taile
                CALL bwritenum(0#, 0, i&, target%, status%)
        NEXT i&
        setno% = 1
        CALL bgrdefw(setno%, target%, status%)
        CALL bgrdefa(0, 0, llength#, status%)
        CALL bgrcurv(1, 0, 1, WHITE%, SOLID%, status%)
        CALL bgrvwpt(0, 0, 32767, 16384, status%)
        CALL bgrdsply(status%)
        CALL bgrpause(status%)
        CALL bgrclose(status%)
        CLS
        PRINT "leading_s = "; leads
        PRINT "leading_e = "; leade
        PRINT "tail_s = "; tails
        PRINT "tail_e = "; taile
        PRINT
        INPUT "Gating function continue (y/n) "; gate$
```

```
    IF gate$ = "n" OR gate$ = "N" THEN 200
    INPUT "lead_s,lead_e,tail_s,tail_e "; leads, leade, tails, taile
    GOTO 100
200 bgrclose (status%)
    '   PRINT "writing original target to file"
    CALL bwftoafile(ntarget%, "otarget.wav", status%)
300 PRINT "writing target to file named TARGET.WAV"
    CALL bwftoafile(target%, "target.wav", status%)
    IF status% <> 0 THEN
            CALL brpterr(status%, "ERROR: WRITING TARGET TO FILE FAILED "): END
    END IF
    CALL bfreewf(background%, status%)
    CALL bfreewf(tarbck%, status%)
    incident% = 4
    CALL bcreate1wf(Length&, WFloat%, incident%, status%)
    IF status% <> 0 THEN
            CALL brpterr(status%, "ERROR: incident% creation failed "): END
    END IF
30  INPUT "ENTER INCIDENT FILE NAME (Default = INC.WAV)"; F3$
    IF F3$ = "" THEN F3$ = "INC.WAV"
    Length = LEN(F3$)           ' 8 lines added by Arch Page
    dotpos = INSTR(F3$, ".")     ' to auto add the ".WAV" ext to filename


    IF dotpos = 0 THEN
            F3$ = F3$ + ".WAV"
    ELSE
            F3$ = LEFT$(F3$, dotpos)
            F3$ = F3$ + "WAV"
    END IF

    CALL bafiletowf(F3$, incident%, status%)
    IF status% <> 0 THEN
            CALL brpterr(status%, "error: converting from ascii file failed "): GOTO 30
    END IF
    '**** all time waveforms are read in, now do the fft
    REM ************ CALCULATE FFT FROM DATA ********************
    rtarget% = 5: itarget% = 6
    CALL bwavetype(target%, wtype%, status%)
    CALL bwavelen(target%, Length&, status%)
    halfLength& = (Length& / 2) + 1
    CALL bcreate1wf(halfLength&, wtype%, rtarget%, status%)
    CALL bcreate1wf(halfLength&, wtype%, itarget%, status%)
    PRINT "start FFT of target "
    CALL brfft(target%, rtarget%, itarget%, status%)
```

```
'   PRINT "end FFT of target"
CALL btupunits(target%, 0, inVunits$, inVlen%, status%)
CALL bdimunits(target%, 0, inHunits$, inHlen%, status%)
CALL bwftunits(inVunits$ + " * " + inHunits$, 0, rtarget%, status%)
CALL bwftunits(inVunits$ + " * " + inHunits$, 0, itarget%, status%)
CALL bwfdunits("1 / " + inHunits$, 0, rtarget%, status%)
CALL bwfdunits("1 / " + inHunits$, 0, itarget%, status%)
CALL bwavelen(target%, Length&, status%)
CALL brdimscale(target%, 0, hsf#, status%)
CALL bwdimscale(rtarget%, 0, 1 / (Length& * hsf#), status%)
CALL bwdimscale(itarget%, 0, 1 / (Length& * hsf#), status%)
CALL bwcmult(rtarget%, hsf#, rtarget%, status%)
CALL bwcmult(itarget%, hsf#, itarget%, status%)
CALL bwftoafile(rtarget%, "rtarget.wav", status%)
CALL bwftoafile(itarget%, "itarget.wav", status%)
mtarget% = 12: ptarget% = 13
PRINT
CALL bcreate1wf(halfLength&, WFloat%, mtarget%, status%)
CALL bcreate1wf(halfLength&, WFloat%, ptarget%, status%)
CALL bpolar(rtarget%, itarget%, DEGREE%, FALSE%, 0#, mtarget%, ptarget%, status%)
PRINT "convert rectangular to polar status ="; status%
CALL btupunits(rtarget%, 0, inVunits$, in1len%, status%)
CALL bwftunits(inVunits$, 0, mtarget%, status%)
CALL bwftunits("Deg", 0, ptarget%, status%)
CALL bdimunits(rtarget%, 0, inHunits$, inHlen%, status%)
CALL bwfdunits(inHunits$, 0, mtarget%, status%)
CALL bwfdunits(inHunits$, 0, ptarget%, status%)
CALL brdimscale(rtarget%, 0, hsf#, status%)
CALL bwdimscale(mtarget%, 0, hsf#, status%)
CALL bwdimscale(ptarget%, 0, hsf#, status%)
CALL bwftoafile(mtarget%, "mtarget.wav", status%)
CALL bwftoafile(ptarget%, "ptarget.wav", status%)
rincident% = 7: iincident% = 8
CALL bcreate1wf(halfLength&, WFloat%, rincident%, status%)
CALL bcreate1wf(halfLength&, WFloat%, iincident%, status%)
CALL brfft(incident%, rincident%, iincident%, status%)
CALL btupunits(incident%, 0, inVunits$, inVlen%, status%)
CALL bdimunits(incident%, 0, inHunits$, inHlen%, status%)
CALL bwftunits(inVunits$ + " * " + inHunits$, 0, rincident%, status%)
CALL bwftunits(inVunits$ + " * " + inHunits$, 0, iincident%, status%)
CALL bwfdunits("1 / " + inHunits$, 0, rincident%, status%)
CALL bwfdunits("1 / " + inHunits$, 0, iincident%, status%)
CALL bwavelen(incident%, Length&, status%)
CALL brdimscale(incident%, 0, hsf#, status%)
CALL bwdimscale(rincident%, 0, 1 / (Length& * hsf#), status%)
```

```
CALL bwdimscale(iincident%, 0, 1 / (Length& * hsf#), status%)
CALL bwcmult(rincident%, hsf#, rincident%, status%)
CALL bwcmult(iincident%, hsf#, iincident%, status%)
CALL bwftoafile(rincident%, "rinciden.wav", status%)
CALL bwftoafile(iincident%, "iinciden.wav", status%)
mincident% = 14: pincident% = 15
CALL bcreate1wf(halfLength&, WFloat%, mincident%, status%)
CALL bcreate1wf(halfLength&, WFloat%, pincident%, status%)
CALL bpolar(rincident%, iincident%, DEGREE%, FALSE%, 0#, mincident%, pincident%,
status%)
CALL btupunits(rincident%, 0, inVunits$, in1len%, status%)
CALL bwftunits(inVunits$, 0, mincident%, status%)
CALL bwftunits("Deg", 0, pincident%, status%)
CALL bdimunits(rincident%, 0, inHunits$, inHlen%, status%)
CALL bwfdunits(inHunits$, 0, mincident%, status%)
CALL bwfdunits(inHunits$, 0, pincident%, status%)
CALL brdimscale(rincident%, 0, hsf#, status%)
CALL bwdimscale(mincident%, 0, hsf#, status%)
CALL bwdimscale(pincident%, 0, hsf#, status%)
CALL bwftoafile(mincident%, "minciden.wav", status%)
CALL bwftoafile(pincident%, "pinciden.wav", status%)
INPUT "Enter scale factor"; SF
INPUT "Enter distance from feedpoint to target in feet "; distance
rcs% = 9
CALL bcreate1wf(halfLength&, WFloat%, rcs%, status%)
CALL bwdiv(mtarget%, mincident%, rcs%, status%)
'set units and scaling for waveform rcs%
CALL btupunits(mtarget%, 0, in1units$, in1len%, status%)
CALL btupunits(mincident%, 0, in2units$, in2len%, status%)
CALL bdimunits(mtarget%, 0, inHunits$, inHlen%, status%)
CALL bwftunits(in1units$ + " / " + in2units$, 0, rcs%, status%)
CALL bwfdunits(inHunits$, 0, rcs%, status%)
CALL brdimscale(mtarget%, 0, hsf#, status%)
CALL bwdimscale(rcs%, 0, hsf#, status%)
rcs2% = 10
CALL bcreate1wf(halfLength&, WFloat%, rcs2%, status%)
CALL bwpower(rcs%, 2#, rcs2%, status%)
'set units and scaling for waveform rcs2%
CALL btupunits(rcs%, 0, V$, llen%, status%)
CALL bwftunits(V$ + " " + V$ + " " + V$, 0, rcs2%, status%)
CALL bdimunits(rcs%, 0, h$, inHlen%, status%)
CALL bwfdunits(h$, 0, rcs2%, status%)
CALL brdimscale(rcs%, 0, hsf#, status%)
CALL bwdimscale(rcs2%, 0, hsf#, status%)
rcs3% = 11
```

```
CALL bcreate1wf(halfLength&, WFloat%, rcs3%, status%)
' Calculate constant for BWS range to complete RCS calc.
k4pi3r4# = ((4 * PI# * ((distance * 12) / 39.37) ^ 4)) * 1.05
kconstant# = k4pi3r4# * SF * SF
CALL bwcmult(rcs2%, kconstant#, rcs3%, status%)
lamda% = 19
CALL bcreate1wf(halfLength&, WFloat%, lamda%, status%)
FOR i& = 0 TO halfLength&
        IF i& = 0 THEN ii& = 1 ELSE ii& = i&
        PRINT ".";
        lamdaf# = 3E+08 / (ii& * hsf#)
        lamdaf# = lamdaf# ^ 2
        CALL bwritenum(lamdaf#, 0, i&, lamda%, status%)
NEXT i&
rcs4% = 20
CALL bcreate1wf(halfLength&, WFloat%, rcs4%, status%)
CALL bwdiv(rcs3%, lamda%, rcs4%, status%)
'set units and scaling for waveform rcs%
CALL btupunits(rcs3%, 0, in1units$, in1len%, status%)
CALL btupunits(rcs3%, 0, in2units$, in2len%, status%)
CALL bdimunits(rcs3%, 0, inHunits$, inHlen%, status%)
CALL bwftunits(in1units$ + " / " + in2units$, 0, rcs4%, status%)
CALL bwfdunits(inHunits$, 0, rcs4%, status%)
CALL brdimscale(rcs3%, 0, hsf#, status%)
CALL bwdimscale(rcs4%, 0, hsf#, status%)
'set units and scaling for waveform rcs4%
CALL btupunits(rcs2%, 0, inVunits$, inVlen%, status%)
inVunits$ = "sigma"
CALL bwftunits("Sigma", 0, rcs4%, status%)
CALL bdimunits(rcs2%, 0, inHunits$, inHlen%, status%)
inHunits$ = "Frequency"
CALL bwfdunits(inHunits$, 0, rcs4%, status%)
CALL brdimscale(rcs2%, 0, hsf#, status%)
hsf# = hsf# / SF
CALL bwdimscale(rcs4%, 0, hsf#, status%)
PRINT
PRINT "writing rcs data to disk file named RCS.WAV"
PRINT
CALL bwftoafile(rcs4%, "rcs.wav", status%)
' Free wave%
PRINT "FREEING SOME WAVEFORM MEMORIES"


PRINT
CALL bfreewf(mincident%, status%)
```

```
CALL bfreewf(pincident%, status%)
CALL bfreewf(mtarget%, status%)
CALL bfreewf(ptarget%, status%)
CALL bfreewf(rcs3%, status%)
CALL bfreewf(rcs2%, status%)
CALL bfreewf(rcs%, status%)
CALL bfreewf(target%, status%)
CALL bfreewf(incident%, status%)
CALL bfreewf(rtarget%, status%)
CALL bfreewf(itarget%, status%)
CALL bfreewf(iincident%, status%)
CALL bfreewf(iincident%, status%)
CALL bfreewf(lamda%, status%)
' New ID for rcs5
rcs5% = 21
CALL bcreate1wf(halfLength&, WFloat%, rcs5%, status%): PRINT "Log Function";
    status%
CALL bwlog(rcs4%, 10!, rcs5%, status%)
CALL bdimunits(rcs4%, 0, inHunits$, inHlen%, status%)
CALL bwfdunits(inHunits$, 0, rcs5%, status%)
CALL brdimscale(rcs4%, 0, hsf#, status%)
CALL bwdimscale(rcs5%, 0, hsf#, status%)
' set New ID for rcs6%
rcs6% = 22
mulnum# = 10#
CALL bcreate1wf(halfLength&, WFloat%, rcs6%, status%): PRINT "10 X Log Function";
    status%
CALL bwcmult(rcs5%, mulnum#, rcs6%, status%)
' Set units and Scale factors
CALL btupunits(rcs5%, 0, V$, llen%, status%)
CALL bwftunits("dBsm", 0, rcs6%, status%)
CALL bdimunits(rcs5%, 0, h$, inHlen%, status%)
CALL bwfdunits(h$, 0, rcs6%, status%)
CALL brdimscale(rcs5%, 0, hsf#, status%)
CALL bwdimscale(rcs6%, 0, hsf#, status%)
Title$ = "RCS of " + F2$
CLS
PRINT
PRINT
INPUT "Use the default title of 'RCS of (tgt_name)' ???  (Y/N)"; an$
IF (an$ = "n" OR an$ = "N") THEN
        INPUT "Type desired string for title "; Title$
END IF
PRINT
CALL bwftitle(Title$, rcs6%, status%)
```

```
PRINT "writing (10 log rcs) data to disk file named LOGRCS.WAV"
CALL bwftoafile(rcs6%, "logrcs.wav", status%)
' 10 log rcs graphics added by DAN TAM on 3/18/94
setno% = 1
CALL bgrdefw(setno%, rcs6%, status%)
CALL bgrdefa(0, 0, llength#, status%)
CALL bgrcurv(1, 0, 1, WHITE%, SOLID%, status%)
CALL bgrvwpt(0, 0, 32767, 16384, status%)
CALL bgrdsply(status%)
CALL bgrpause(status%)
CALL bgrclose(status%)
CALL bfreewf(rcs4%, status%)
CALL bfreewf(rcs5%, status%)
CALL bfreewf(rcs6%, status%)
IF status% <> 0 THEN
        CALL brpterr(status%, "Free waveform failed"): END
END IF
CLS
PRINT
PRINT
PRINT "Returning the memory to the system"
' Return waveform memory to system
dummy& = SETMEM(150001)
END
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 1995	Final

4. TITLE AND SUBTITLE

TRANSIENT SCATTERING RESPONSE OF AN AIRBORNE TARGET

5. FUNDING NUMBERS

PE: 0602111N
WU: ICSXB200

6. AUTHOR(S)

J.H. Schukantz, Jr., D.W.S. Tam, J.B. McGee, L.B. Koyama, J.W. Rockway, S.T. Li

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Command, Control and Ocean Surveillance Center (NCCOSC)
RDT&E Division
San Diego, CA 92152–5001

8. PERFORMING ORGANIZATION REPORT NUMBER

TD 2796

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The research team performed measurements in a bounded wave simulator (BWS) to obtain the radar cross section (RCS) of a scale model of a particular airborne target. Measurements of the airborne target were obtained from two different flying orientations. The transient scattering response data were scaled to allow comparison with the RCS obtained by numerical methods for a full-size target. The RCS of the scale model of an airborne target from two different flying orientations was fairly independent of "flying" orientation, or rotation about its central axis. The research team established the frequency range for which these measurements are valid by determining the point at which the noise overcomes the signal reflected by the target, i.e., where the RCS of the noise spectrum reaches the level of the target RCS. A test was performed to determine how well the RCS data obtained by the scale model measurements agreed with those predicted by numerical analysis using the Numerical Electromagnetic Code (NEC). At low frequencies, the computer curve rose steeply to cross the measured curve at about 12 MHZ. Between 13 and 60 MHz, the computer curve followed the shape of the scale model measurements, but exceeded it by approximately 5 dB. The measured data were from a 45-degree fin rotation.

14. SUBJECT TERMS

bounded wave simulator (BWS)
radar cross section (RCS)
scale model measurement
Numerical Electromagnetic Code (NEC)

15. NUMBER OF PAGES

36

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAME AS REPORT

21a. NAME OF RESPONSIBLE INDIVIDUAL	21b. TELEPHONE *(include Area Code)*	21c. OFFICE SYMBOL
J. B. McGee	(619) 553–3778	Code 804/822

INITIAL DISTRIBUTION

Code 0012	Patent Counsel	(1)
Code 0271	Archive/Stock	(6)
Code 0274	Library	(2)
Code 70	T. F. Ball	(1)
Code 75	J. E. Griffin	(1)
Code 755	R. J. Dinger	(5)
Code 755	E. L. Nelson	(1)
Code 755	T. E. Tice	(5)
Code 755	G. A. Kinzel	(1)
Code 80	K. D. Regan	(1)
Code 804	J. W. Rockway	(1)
Code 82	R. J. Kochanski	(1)
Code 822	L. B. Koyama	(1)
Code 822	S. T. Li	(5)
Code 822	J. C. Logan	(1)
Code 822	J. B. McGee	(10)
Code 822	L. C. Russell	(1)
Code 822	J. H. Schukantz	(1)
Code 822	D. W. S. Tam	(1)
Code 832	P. M. Hansen	(1)

Defense Technical Information Center
Alexandria, VA 22304–6145 (2)

NCCOSC Washington Liaison Office
Washington, DC 20363–5100

Center for Naval Analyses
Alexandria, VA 22302–0268

Navy Acquisition, Research and Development
Information Center (NARDIC)
Arlington, VA 22244–5114

GIDEP Operations Center
Corona, CA 91718–8000

Office of Naval Research
Arlington, VA 22217–5000 (2)

Commander, NAWC–WPNS
Point Mugu, CA 93042–5001 (3)

Naval Postgraduate School
Monterey, CA 93943–5100

U.S. Army Research Laboratory
Adelphi, MD 20783–1197

Hughes Aircraft
Fullerton, CA 92634