HDL-TM-90-23
November 1990

The Fast Hartley Transform

by Mark Hyosun Mar

U.S. Army Laboratory Command
Harry Diamond Laboratories
Adelphi, MD 20783-1197
The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.
This report describes the Fast Hartley Transform (FHT), which is twice as fast as the Fast Fourier Transform (FFT), uses only half the computer memory, and is somewhat better for applications such as spectral analysis, signal processing, and convolution. The FHT output is generated with a Fortran computer code and compared to the results of a typical FFT algorithm.
Contents

Background ......................................................................................... 5
1. Introduction ...................................................................................... 5
2. General Description of FHT .......................................................... 5
3. Comparison ....................................................................................... 7
4. Performance ....................................................................................... 9
5. Conclusions ..................................................................................... 13
Acknowledgments ................................................................................ 14
References ........................................................................................... 14
Distribution ........................................................................................ 25

Appendices

A.—Fortran Program FHT.FOR .......................................................... 15
B.—Fortran Program BODE.FOR ....................................................... 21

Figures

1. Generalized EMP waveform .......................................................... 10
2. Fast Hartley transform of EMP waveform ...................................... 11
3. Fast Fourier transform of EMP waveform ..................................... 11
4. Calculated double exponential EMP spectrum .............................. 12
5. Comparison of figure 4 with figures 2 and 3 ................................. 12
Background

The purpose of this paper is to report the results of testing the Fast Hartley Transform (FHT) and comparing it with the Fast Fourier Transform (FFT). All the definitions and equations in this paper are quoted and cited from the series of references. The author of this report developed a Fortran program which computes the Hartley transform. He tested the program with a generalized electromagnetic pulse waveform and verified the result with the known value.

1. Introduction

Fourier analysis is an essential tool to obtain frequency domain information from transient time domain signals. The FFT is a popular tool to process many of today's audio and electromagnetic signals. System frequency response, digital filtering of signals, and signal power spectrum are the most practical applications of the FFT. However, the Fourier integral transform or the FFT requires the computer resources appropriate to the complex arithmetic operations. On the other hand, the FHT can accomplish the same results faster and requires fewer computer resources.[1] The FHT is twice as fast as the FFT, uses only half the computer resources, and so could be more useful than the FFT in typical applications such as spectral analysis, signal processing, and convolution.[2] This paper presents a Fortran computer program for the FHT algorithm along with a brief description and compares the results and performance of the FHT and the FFT algorithms.

2. General Description of FHT

Equation (1) defines the analytic form of the Hartley transform, and (2) shows its inverse transform, which switches the frequency function back into the time domain [1]:

\[ H(f) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(t) \text{cas}(2\pi ft) dt, \quad (1) \]

\[ X(t) = \int_{-\infty}^{\infty} H(f) \text{cas}(2\pi ft) df, \quad (2) \]

where \( \text{cas}(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft) \).
The cas function was introduced by R. V. L. Hartley, who first proposed the Hartley transform in 1942 [3].

The above equations are very similar to the Fourier transform, equation (3), and its inverse, equation (4) [1].

\[
F(f) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(t) e^{-j2\pi ft} dt ,
\]

\[
X(t) = \int_{-\infty}^{\infty} F(f) e^{j2\pi ft} df ,
\]

where \(e^{2\pi ft} = \cos (2\pi ft) + j \sin (2\pi ft)\) and \(e^{-j2\pi ft} = \cos (2\pi ft) - j \sin (2\pi ft)\).

(These are known as Euler’s formulas.)

Note the electrical engineering convention of labeling the imaginary unit \(i\) as \(j\) (\(i\) equals the square root of \(-1\)).

These four equations deal only with continuous time variables. In the real world, however, signals are sampled at discrete intervals of time. So there are discrete transforms that approximate the Fourier integral and the Hartley integral. The discrete forms of the Hartley transform (DHT) pairs are [1]

\[
H(f) = \frac{1}{N} \sum_{t=0}^{N-1} X(t) \cos (2\pi ft/N) ,
\]

and

\[
X(t) = \sum_{f=0}^{N-1} H(f) \cos (2\pi ft/N) .
\]

The discrete Fourier transform (DFT) pairs are very similar to the DHT [1].

\[
F(f) = \frac{1}{N} \sum_{t=0}^{N-1} X(t) e^{-j2\pi ft/N} ,
\]

and

\[
X(t) = \sum_{f=0}^{N-1} F(f) e^{j2\pi ft/N} .
\]

As can be seen in equation (7), the DFT requires \((N - 1)\) complex multiplications and \((N - 1)\) complex additions to compute each output point (the first term in the sum involves \(\exp (j * 0) = 1\) and therefore does not require a multiplication). Thus, to compute \(N\) output points, \(N(N - 1)\) complex multiplications and the same number of complex additions are required. Now each complex multiplication requires four real multiplications and two real additions. Hence, for computa-
tion of all the output points for \(N\)-point data, the DFT requires \(4N(N - 1)\) real multiplications and \(4N(N - 1)\) real additions. To overcome this computational requirement, the FFT algorithm (Cooley and Tukey algorithm) was developed for the machine computation of a complex Fourier transform [4]. The FFT uses a permutation process to bisect the data sequence until data pairs are reached. The fundamental concept for the permutation process is that it is faster to divide the data set into pairs, compute the transform of the pairs individually, and recombine them to make the entire transform rather than to compute the transform as a whole data set. The Fourier transform of the time domain data set can be obtained by superimposing all permuted data pairs. An \(N\)-point FFT, where \(N\) is a power of 2, requires \(2N \log_2 N\) real multiplications and \(3N \log_2 N\) real additions, which means a factor of about 200 times less multiplication for \(N\) equals 1024 time data points [5]. Similarly, Bracewell developed a fast algorithm for the Hartley transform. However, to use the algorithm, the decomposition formula that expresses a complete DHT in terms of its half-length subsequences is required. Bracewell has shown the following decomposition formula by application of the shift and the similarity theorems for the DHT [6]. Using an equivalent concept for the Fourier transform, the similar decomposition formula can be defined [1].

\[
H(f) = H_1(f) + H_2(f) \cos \left(\frac{2\pi f}{N_s}\right) + H_2(N_s - f) \sin \left(\frac{2\pi f}{N_s}\right),
\]

\[
F(f) = F_1(f) + F_2(f) e^{i \frac{2\pi f}{N_s}},
\]

where \(N_s\) is the number of elements in the half-length sequence, and thus \(N_s = \frac{N}{2}\) for a data set of \(N\) elements.

As can be seen, there is one important difference between equations (9) and (10). While the FFT decomposition formula is symmetric, the FHT decomposition formula is asymmetric because of the sine coefficients on both \(H(f)\) and \(H(N_s - f)\). This asymmetric matrix processing requires special handling such as retrograde indexing for computer implementation. The retrograde indexing behavior can be described by "using an independent variable as an index for the elements multiplied by the sine coefficients. This index decreases while the other indexes increase [1]."

3. Comparison

The major difference between the two transform algorithms is the real function \(\cos\) in the Hartley transform and the complex exponential term in the Fourier transform. Since real arithmetic is much simpler
than complex computation, the FHT is faster than the FFT and requires fewer computer resources. Furthermore, the complex Fourier spectrum can be obtained from the Hartley transform. It is faster to generate the Fourier transform and power spectrum with the FHT than with the FFT, because the FHT algorithm uses real rather than complex quantities and so requires fewer floating-point operations.

If a function can be expressed uniquely into even and odd parts, then from the even and odd parts the original function can be uniquely reconstructed. Based upon the symmetrical property for the even function and the asymmetrical property for the odd function, the following relationships can be established [6].

Let \( H(f) = E(f) + O(f) \), where \( E(f) \) and \( O(f) \) are the even and odd parts of \( H(f) \), respectively. Then

\[
E(f) = \frac{H(f) + H(-f)}{2} = \int_{-\infty}^{\infty} V(t) \cos 2\pi ft \, dt \quad (11)
\]

and

\[
O(f) = \frac{H(f) - H(-f)}{2} = \int_{-\infty}^{\infty} V(t) \sin 2\pi ft \, dt . \quad (12)
\]

According to equations (11) and (12), the Fourier transform can be obtained from the Hartley transform simply by reflections and additions of the even and odd parts. Since the real part of the FFT is equal to the even part of the FHT, and the imaginary part of the FFT equals the negative odd part of the FHT, the real and imaginary parts of the FFT can be obtained from the FHT according to the following equations [1]:

\[
F_r = H(f) + H(N - f) \quad (13)
\]

\[
F_{im} = H(f) - H(N - f) , \quad (14)
\]

where \( F_r \) is the real portion of the complex Fourier transform, \( F_{im} \) is the imaginary portion, and \( N \) is the number of elements in the data set.

The power spectrum can be obtained directly from the FHT using the following equation [1]:

\[
P_s(f) = \left[ |H(f)|^2 + H(N - f)^2 \right]/2 , \quad (15)
\]

where \( P_s \) is the power spectrum.

While the FFT computes the square of the real and imaginary parts and sums the two values at a given frequency, the FHT squares and sums the two values of the Hartley transform at the positive and negative frequencies. Since the FHT computes the energy content in
the positive and negative frequency domain, a factor of two is required in the above equation.

The convolution theorem is almost identical between the Hartley and the Fourier transforms. Equation (16) summarizes the convolution theorem for the Hartley transform, corresponding to equation (17) for the Fourier transform [1].

\[
f_1(t) \ast f_2(t) = H_1(f) H_2(f) + H_1(-f) H_2(-f),
\]

where \( H_{2e}(f) \) is the even part of \( H_2(f) \) and \( H_{2o}(f) \) is the odd part of \( H_2(f) \).

\[
f_1(t) \ast f_2(t) = F_1(f) F_2(f)
\]

The \( \ast \) symbol denotes the convolution operation.

Note that if one of the functions being convolved is either even or odd, then the convolution theorem for the Hartley transform reduces to the particularly simple form indicated below [1]:

\[
f_1(t) \ast f_2(t) = H_{1e}(f) H_{2o}(f).
\]

### 4. Performance

The Fortran program, FHT.FOR, is presented in appendix A as developed from the basic program presented by Bracewell [2]. Using the generalized double exponential waveform as a typical electromagnetic pulse (EMP) electric field, a time domain data file is generated. The generalized double exponential EMP electric field time behavior is given by

\[
E(t) = 5.25 \times 10^6 \left[ \exp(-4 \times 10^6 t) - \exp(-4.76 \times 10^6 t) \right],
\]

in volts per meter, where \( t \) is in seconds. This pulse has a peak value of 50 kV/m, a 10- to 90-percent risetime of about 5 ns, and a time to half-value of about 200 ns [7]. Figure 1 plots the time waveform of this constructed high-altitude EMP electric field. The time domain waveform has 1024 equally spaced points with a time increment of 1 ns. The user can choose any number of data points consistent with the available computer memory size, but must be a power of 2 for algorithm simplicity and additional execution speed. The sample data have to be equally spaced for the FHT algorithm. If more than 1024 data points are required, the FHT.FOR program has to be modified to expand the size of the arrays. The time increment can be controlled also, and 1 ns is used as an example. As the FHT.FOR program executes, the user has an option to choose a format for the transformed
output data. One format has frequency and magnitude only while the other one includes frequency and the real and imaginary parts of the FFT spectrum. Figure 2 shows the FHT.FOR output, and figure 3 shows the FFT output as generated with the HOBOII signal-processing software package [8]. As can be seen from figures 2 and 3, the FFT and FHT results appear to be identical. If drawn on one graph, they would overlay. The real and imaginary parts of the frequency spectrum obtained from the FHT algorithm were verified with the FFT output from HOBOII. An inverse FFT applied to the FHT results reproduces the input waveform which verifies the accuracy of the FHT algorithm.

As can be noticed in figures 2 and 3, both transform algorithms produce a nonphysical behavior at higher frequencies. To explore this deviation and compare these transforms with the original double exponential EMP spectrum, another Fortran program, BODE.FOR, was developed. This program is included in appendix B. The BODE.FOR program computes the double exponential frequency spectrum using the analytic form of Fourier transform for the differences of two exponentials [7]. The calculated spectrum is shown in figure 4, according to the output of BODE.FOR. In figure 5, the FHT and FFT are superimposed on the analytic spectrum. Both transforms trace the calculated spectrum, except for discrepancies at the high

Figure 1. Generalized EMP waveform.
frequency as previously mentioned. Since the double exponential pulse has a risetime of about 5 ns, there is an early time slope discontinuity between the origin and the first sample data point for a 1-ns sample time increment. This situation can be improved if we take

Figure 2. Fast Hartley transform of EMP waveform.

Figure 3. Fast Fourier transform of EMP waveform.
Figure 4. Calculated double exponential EMP spectrum.

Figure 5. Comparison of figure 4 with figures 1 and 3.

Note:
Solid line, calculated double exponential EMP spectrum.
Dotted line, FFT and FHT.
more data points in the early time, which requires a smaller time increment. Another difference between the analytic frequency spectrum and that obtained with the FHT and FFT transforms is that the low frequency limit is undetermined. In order to obtain the low frequency value of the transformed data, the time domain data set has to have a much larger time window, which would require more time to process the transform. The time increment could also be increased, but would lead to a loss of the high-frequency content. According to the Nyquist sampling theorem, 200,000 time domain data points are required with a time increment of 0.5 ns to generate a double exponential frequency spectrum with the FHT or FFT, which would be similar to figure 4. The Nyquist sampling theorem requires that the sampling rate must be at least twice the highest frequency of interest in the waveform being sampled [9].

When the speed of two transforms is compared, theoretically, the FHT is faster than the FFT; however, it is hard to compare the run time of these two programs for the transform process itself because the programs used in this effort, FHT.FOR and HOBOII, are implemented differently. Also, for all practical purposes both codes process the transform within a few seconds.

5. Conclusions

Since the FHT uses only real valued functions, there is no need for complex calculations, which implies faster run-times and less computer memory to process a signal in comparison to a typical FFT algorithm. Finally, the FHT uses fewer operations to transform a given signal, so there are fewer round-off errors.

The example illustrated does not prove that the FHT is superior to the FFT, although it demonstrates that the FHT is fully compatible with the FFT. However, if large amounts of data are being manipulated, there could be a significant difference in speed and the amount of memory required. The FHT offers better performance using fewer computational resources.

The FHT.FOR program can be further developed to perform a convolution that requires only real arithmetic. An FHT convolution algorithm would be faster and easier to implement than a similar algorithm developed for the FFT.
Acknowledgments

I would like to thank Andrew Hermann, who advocated this subject for my first project. I would particularly like to thank William Coburn for his guidance and encouragement throughout this project.

References


Appendix A.—Fortran Program FHT.FOR
This routine generates the fast Hartley transform (FHT) and the complex form of fast Fourier transform (FFT) according to a given time domain function.

```plaintext
REAL L2
INTEGER P7,T,U,S2,Q,D,E,SS,SO,P,O,P,M7,T,M7,RMAG(1024) DIMENSION F(10,1024),R(1024),X(1024),FR(1024),RMAG(1024) DIMENSION S(1024),C(1024),M(l1),RR(1024),XX(1024) CHARACTER*20 FILNAM,FILENAM

Opening the output files.

WRITE(*,*) TYPE OUTPUT FILENAME FOR THE TIME DOMAIN' READ(*,*) FILNAM
OPEN(3,FILE=FILNAM,STATUS='NEW')
WRITE(*,*) TYPE OUTPUT FILENAME FOR THE FREQUENCY SPECTRUM' READ(*,*) FILENAM
OPEN(IO,FILE=FILENAM,STATUS='NEW')

Initialization.

WRITE(*,*) 'HOW MANY DATA POINTS.#'
WRITE(*,*) 'POWER OF 2 BUT LESS OR EQUAL TO 1024' READ(*,*) NU
P = 0
1 NUM = NU/2
P = P + 1
NU = NUM
IF (NUM. GE. 2) GOTO 1
N4 = 2**(P-2)
N2 = N4 + N4
N = N2 + N2
N7 = N - 1
P7 = P - 1
WRITE(*,*) 'WHAT IS YOUR TIME INCREMENT ?' READ(*,*) DTM
WRITE(*,*) 'WHAT ARE YOUR PARAMETERS ?' WRITE(*,*) 'E(t) = EO*(EXP(-AP*t))-EXP(-BT*t)' WRITE(*,*) 'EO=?, AP=?, BT=?'
READ(*,*) EO,AP,BT
TM = 0.0
```
APPENDIX A

C Generating the time domain waveform.

DO 10 I = O,N7
   FNF = EO*(EXP(-AP*TM)-EXP(-BT*TM)) WRITE(3,*) TM,FNF
   TM = TM + DTM
   DF = I.O/((N-I)*DTM)
   FR(I) = I*DF
   F(O,I) = FNF
   F(I,I) = FNF
10 CONTINUE

C Generating the power of 2 numbers.

I = 1
   M(O) = 1
   M(I) = 2
20 M(I+1) = M(I) + M(I)
   I = I + 1
   IF (I. LT .P) GOTO 20

C Get the sin and cos coefficients.

PI = 4*ATAN(I.O)
   W = 2*PI/N
   A = 0
   DO 30 I = 1,N
      A = A + W
      S(I) = SIN(A)
      C(I) = COS(A)
30 CONTINUE

C Start permutation.

J = -1
   I = -1
50 I = I + 1
   T = P
40 T = T -1
   J = J - M(T)
   IF (J. GE .-1) GOTO 40
   J = J + M(T+1)
   IF (I. LE .J) GOTO 50
APPENDIX A

\[ T = F(O,I+1) \]
\[ F(O,I+1) = F(O,J+1) \]
\[ F(O,J+1) = T \]

IF (I < N - 3) GOTO 50

C First stage.

DO 60 I = O,N-2,2
    \[ F(I,I) = F(O,I) + F(O,I+1) \]
    \[ F(I,I+1) = F(O,I) - F(O,I+1) \]
  60 CONTINUE

C Second stage.

U = P7
SS = 4
DO 90 L = 2,P7
    S2 = SS + SS
    U = U - 1
    SO = M(U-1)
    DO 100 Q = O,N7,S2
        I = Q
        D = I + SS
        F(L+1,I) = F(L,I) + F(L,D) \[ F(L+1,D) = F(L,I) - F(L,D) \]
        K = D - 1
        DO 110 J = SO,N4,SO
            I = I + 1
            D = I + SS
            E = K + SS
            Y = F(L,D)*C(J) + F(L,E)*S(J) \[ Z = F(L,D)*S(J) - F(L,E)*C(J) \]
            F(L+1,I) = F(L,I) + Y
            F(L+1,D) = F(L,I) - Y
            F(L+1,K) = F(L,I) + Z
            F(L+1,E) = F(L,K) - Z
        K = K + 1
    110 CONTINUE
    E = K + SS
    100 CONTINUE
    SS = S2
  90 CONTINUE

C Normalizing the fast Hartley transform's magnitude.

\[ RMAG(O) = F(L,O)/(DF*N) \]
APPENDIX A

\[
\begin{align*}
RR(O) &= F(L,O)/(DF*\text{N}) \\
XX(O) &= 0
\end{align*}
\]

C Select either magnitude of the FHT
C or the real and imaginary part of the FFT.

WRITE(*,*) 'SELECT THE DESIRED OUTPUT' WRITE(*,*) '1 = MAGNITUDE, 2 = REAL & IMAGINARY' READ(*,*) OPTION
IF (OPTION. EQ .1) THEN

C The fast Hartley transform's magnitude is derived from the power spectrum

DO 120 I = N-1,(N/2)+1,-1
    \( F(L,I) = F(L,I)/(\text{N}^*\text{DF}) \)
    \( F(L,N-I) = F(L,N-I)/(\text{N}^*\text{DF}) \)
    \( \text{RMAG}(I) = \sqrt{((F(L,I)^2 + F(L,N-I)^2)/2} \)
    WRITE(10,*)
120 CONTINUE

C Get the real and imaginary parts of the fast Fourier transform from the fast Hartley transform.
ELSE

DO 130 I = N-1,(N/2)+1,-1
    \( B = F(L,N-I) \)
    \( RR(I) = F(L,I) + B \)
    \( XX(I) = F(L,I) - B \)
    IF (I. LE .N-I) THEN
        \( J = I \)
    ELSE
        \( J = N - 1 \)
        \( RR(J) = RR(J) \)
        \( XX(J) = -XX(J) \)
    ENDIF

    \( R(I) = RR(I)/(2*N*DF) \)
    \( X(I) = XX(I)/(2*N*DF) \)
    WRITE(IO,*)
130 CONTINUE

ENDIF
END
Appendix B.—Fortran Program BODE.FOR
This is a program to compute the analytic frequency spectrum of a double exponential pulse. Use COMPLEX constants, variables, operations, and functions.

Input variables (all real):
- First: Starting frequency
- Last: Stopping frequency
- INC: Additive frequency increment

Intermediate variables, all complex:
- DI: First factor in denominator
- D2: Second factor in denominator

These factors are set up for using the CMPLX function, which converts from the form of two REAL values, representing the real and imaginary parts of the complex numbers, to the form of one Fortran COMPLEX number.

```fortran
REAL*8 K, FIRST, LAST, INC, OMEGA, ABSVAL, FRE COMPLEX*16 E, DI, D2
CHARACTER*20 FILNAM

C Opening the output file.
WRITE(*,*) 'TYPE OUTPUT FILENAME'
READ(*,*) FILNAM
OPEN (3, FILE=FILNAM, STATUS='NEW')

C Read parameters, validate.
WRITE(*,*) 'FIRST = ?, LAST = ?, INC = ?' READ(*,*) FIRST, LAST, INC
IF (FIRST .GE. LAST) THEN
  WRITE(*,*) 'INVALID DATA; PROGRAM ABORTED' STOP
ENDIF

C Set frequency to starting value.
OMEGA = FIRST*8*ATAN(1.O)

C While OMEGA <= LAST
10 CONTINUE
  IF (OMEGA .LE. LAST) THEN
```

23
APPENDIX B

\[
\begin{align*}
D1 &= \text{CMPLX}(4.0D6,0\text{MEGA}) \\
D2 &= \text{CMPLX}(4.76D8,0\text{MEGA}) \\
E &= 2.47D13/(D1*D2)
\end{align*}
\]

C Get complex absolute value, = magnitude of output.

\[
\begin{align*}
\text{ABSVAL} &= \text{CDABS}(E) \\
\text{FRE} &= \text{OMEGA}/(8*\text{ATAN}(1.0)) \\
& \text{WRITE}(3,* \text{FRE,ABSVAL)}
\end{align*}
\]

C Incrementing frequency.

\[
\begin{align*}
\text{FRE} &= \text{INC} + \text{FRE} \\
\text{OMEGA} &= \text{FRE}*8*\text{ATAN}(1.0) \\
& \text{GOTO 10} \\
& \text{END IF} \\
& \text{END}
\end{align*}
\]
<table>
<thead>
<tr>
<th>Organization</th>
<th>Address</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catholic University of America</td>
<td>Attn Cardinal Station</td>
<td>PO Box 232, Washington, DC 20017</td>
</tr>
<tr>
<td>Rutgers University</td>
<td>Attn Department of Ceramics</td>
<td>College of Engineering, New Brunswick, NJ 08903</td>
</tr>
<tr>
<td>Sandia Laboratories</td>
<td>Attn Livermore Laboratory</td>
<td>PO Box 969, Livermore, CA 94550</td>
</tr>
<tr>
<td>San-Jia National Laboratories</td>
<td>Attn 1'0 Box 5800</td>
<td>Albuquerque, NM 87185</td>
</tr>
<tr>
<td>Commander</td>
<td>Technical Support Activity</td>
<td>Ft. Monmouth, NJ 07703</td>
</tr>
<tr>
<td>University of California</td>
<td>Attn School of Engineering</td>
<td>Los Angeles, CA 90024</td>
</tr>
<tr>
<td>US Army Laboratory Command</td>
<td>Attn Technical Director, AMSLC-TD</td>
<td></td>
</tr>
<tr>
<td>USAISC</td>
<td>Attn Record Copy, AMSLC-IM-VA</td>
<td>Attn Technical Reports Branch, AMSLC-IM-VP (2 copies)</td>
</tr>
<tr>
<td>Harry Diamond Laboratories</td>
<td>Attn D/Division Directors</td>
<td>Attn Library, SLCHD-TL (3 copies)</td>
</tr>
<tr>
<td></td>
<td>Attn Library, SLCHD-TL (Woodbridge)</td>
<td>Attn Chief, SLCHD-NW (W. Vault)</td>
</tr>
<tr>
<td></td>
<td>Attn Chief, SLCHD-NW-C5 (L. Jasper)</td>
<td>Attn Chief, SLCHD-NW-E (J. Ingram)</td>
</tr>
<tr>
<td></td>
<td>Attn Chief, SLCHD-NW-EH (J. Beilfuss)</td>
<td>Attn Chief, SLCHD-NW-EP (J. Milella)</td>
</tr>
<tr>
<td></td>
<td>Attn Chief, SLCHD-NW-ES (T. Bock)</td>
<td>Attn Chief, SLCHD-NW-P (J. Corrigan)</td>
</tr>
<tr>
<td></td>
<td>Attn Chief, SLCHD-NW-RP (J. McGarity)</td>
<td>Attn Chief, SLCHD-NW-R5 (F. Jack Agee)</td>
</tr>
<tr>
<td></td>
<td>Attn Chief, SLCHD-NW-TN (S. Share)</td>
<td>Attn Chief, SLCHD-NW-TS (J. Halpin)</td>
</tr>
<tr>
<td></td>
<td>Attn Mark Mar (15 copies)</td>
<td></td>
</tr>
<tr>
<td>Institute for Telecommunications Sciences</td>
<td>Attn Library</td>
<td>Ft. Monmouth, NJ 07703</td>
</tr>
<tr>
<td>NASA</td>
<td>Attn Technical Library</td>
<td>Langley Research Center, Hampton, VA 23665</td>
</tr>
<tr>
<td></td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attn Technical Library</td>
<td>John F. Kennedy Space Center, FL 32889</td>
</tr>
<tr>
<td>NASA</td>
<td>Attn Technical Library</td>
<td></td>
</tr>
<tr>
<td>National Communications System</td>
<td>Attn Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>Attn Technical Library</td>
<td></td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>Attn Technical Library</td>
<td></td>
</tr>
<tr>
<td>Cleveland, OH 44135</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>National Security Agency</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>National Security Agency</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>National Security Agency</td>
<td></td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>National Security Agency</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Aerospace Medicine Research Institute</td>
<td></td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>Attn Technical Library</td>
<td></td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td></td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>NASA</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
<tr>
<td>Lewis Research Center</td>
<td>NASA</td>
<td></td>
</tr>
<tr>
<td>Attn Technical Library</td>
<td>Attn Technical Library</td>
<td>Ft. Meade, MD 20755</td>
</tr>
</tbody>
</table>