SHARE PROGRAM LIBRARY AGENCY

PROGRAM NUMBER

134002

University of Miami

1365 MEMORIAL DRIVE - CORAL GABLES, FLORIDA
(305) - 284-6257

IBM Corporation
Program Information Department (PID)
40 Saw Mill River Road
Hawthorne, New York 10532, U.S.A.
Attention: Program Control Desk

CONTRIBUTED PROGRAM LIBRARY SUBMITTAL FORM
(for IBM S/360, 1130 and 1800)

This form should be completed and submitted with the program package to PID at the address shown above. Standards and instructions for submitting programs are in your *User Group Reference Manual* or the *Contributed Program Submittal Standards Manual* available from PID.

1. Program Order Number *(to be filled in by PID)* . . . 360D-13.4.002
2. System Type *(machine)* . . . 360D
3. Search Key . . . COOLEY-TUKEY / FAST / FOURIER TRANSFORM / FOUR1
4. Programming Language . . .
5. Author's Name and Address . . .
6. Direct Inquiries to Name and Address *(if different than Author)* — Norman Brenner
   Norman Brenner, Rm. 36-005
   IBM Research
   P.O. Box 218
   Yorktown Heights, NY 10598
   26
7. Title of Program . . . Cooley-Tukey Fast Fourier Transform--FOUR1
8. Submitter's User Group Affiliation Code and Installation Code . . . MIT
9. Submitter's Own Program Identification and Suffix *(optional)* . . . FFT1
10. Primary Subject Code . . . 13.4
11. Secondary Subject Codes . . . 43.3 44.0
12. Operating or Monitor System Required . . .
13. New or Revision Code *(if revision, show prior Program Order Number in item 1)* . . . R
14. Year Completed . . . 68
15. Date of Submittal . . . 092168
16. Documentation *(number of original pages submitted)* . . . 1
17. **Abstract** *(should contain sufficient information for a reader to determine the value of the program). Listed on the reverse side of this form are subjects which may serve as a guide for a descriptive abstract.*

CONTRIBUTED PROGRAM LIBRARY SUBMITTAL FORM

Subject Guide

a. Purpose
b. Programming Language used
c. Version and modification level or release number of IBM Programming System used, or program order number for non-IBM authored program used
d. Field of application
e. Type of routine (main program, subroutine, etc.)
f. Specific description of machine requirements
g. Engineering Changes (EC) level of equipment (if pertinent)

ABSTRACT

SUBROUTINE FOUR1 (DATA,N,ISIGN)

A very short (65 cards) version of the Cooley-Tukey Fast Fourier Transform in USASI Basic Fortran. For N data points, FOUR1 runs in time proportional to N log N, while pre-FFT methods run in N**2. For large N, this is seconds versus hours. The FFT is the fastest and most accurate way of computing periodograms of time series, convolutions and correlations, Fourier coefficients, digital filters, partial differential equations, etc. See IEEE Transactions on Audio and Electroacoustics (June 1967), a special issue on the FFT and its applications. DATA is a one-dimensional complex array of floating-point data which is transformed and replaced with the transform values. N is the length of array DATA. It must be a power of two. ISIGN is +1 or -1, the transform direction.

(Please attach additional pages if necessary) . . . . **Total pages attached** 0

Permission to Publish

"I hereby give anyone permission to reprint, reproduce, and distribute this program to anyone else."

(18) **Signature of Submitter and Date** Norman Brenner 21 Sept. 1968

(19) **Signature of Installation Addressee** ____________

T4SF

Table of Contents

DISCLAIMER

Deck Key Sheet

Two Fortran decks are present in the FOUR1 Package.

Deck name	ID in cc 73-75	Last seq. no. in cc 77-79 (1st is 1)
(main Test program)	TF1	77
FOUR1	FF1	65

1. Title	Fast Fourier Transform--FOUR1
2. Language/Length	USASI Basic Fortran/65 cards
3. Programmer	Norman Brenner, MIT Lincoln Lab
4. Date	12 September 1968

5. Description

This is a Fortran version of the Cooley-Tukey Fast Fourier Transform for use upon one-dimensional arrays of complex data whose length is a power of two. The definition of the discrete Fourier transform is:

$$\text{TRANSFORM}(K) = \sum_{J=1}^{N} \text{DATA}(J) * \text{EXP}(\text{ISIGN}*2\pi i*(J-1)*(K-1)/N)$$

for all K from 1 to N. N is a power of two, one or greater. ISIGN is +1 or -1. The transform values are returned to array DATA, replacing the input. If a set of data be +1 transformed, and the transform values -1 transformed (or vice versa), the original data reappear, multiplied by N. DATA is a complex array, with real and imaginary parts stored adjacently.

If the input DATA(J) are considered to be located at times (J-1)*T, then the transform values TRANSFORM(K) are located at frequencies (K-1)*F, $F = 2\pi/(NT)$. By periodicity, all frequencies above the "foldover frequency" π/T may be identified with the negative frequency $2\pi/T$ lower.

If a non-power of two number of points are to be transformed, pad out DATA(J) with zeroes to the next highest power of two. The results will be interpolated according to the previous paragraph with the new N--"spectrum broadening".

6. Entry/Calling Sequence

To transform an array of length 256:

With COMPLEX arithmetic	Without COMPLEX arithmetic

<pre>   DIMENSION DATA(256)
   COMPLEX DATA
   DO 10 J=1,256
10 DATA(J)=complex value
   CALL FOUR1(DATA,256,-1)</pre>
	<pre> DIMENSION DATA(2,256)
	   DO 10 J=1,256
	   DATA(1,J)=real part
	10 DATA(2,J)=imaginary part
	   CALL FOUR1(DATA,256,-1)</pre>

In general, CALL FOUR1(DATA,N,ISIGN).

7. Error messages/Return

No error messages; normal return.

8. Algorithm

See IEEE Transactions on Audio and Electroacoustics (June 1967), a special issue on the Fast Fourier Transform.

9. Timing

Naive programs doing the summation directly take N^2 time. FOUR1 takes $N(A+B \log_2 N)$ time. On the CDC 3300, with 16 microsecond floating multiply, A = 450 and B = 120 microseconds.

10. Accuracy

Gentleman and Sande show that the FFT is the most accurate Fourier transform method. Their upper bound on the RMS relative error is: $6 \times 2^{1/2} \times 2^{-b} \times \log_2 N$, where b is the number of bits in the floating point fraction. See 1966 FJCC Proceedings, vol. 29, pp. 569-571.