DECUS
PROGRAM LIBRARY

DECUS NO.	8-485
TITLE	GEOMETRIC DATA FRUNCATION FOR FOURIER TRANSFORM PROGRAMS
AUTHOR	E. A. Barnhardt
COMPANY	Southwestern at Memphis Memphis, Tennessee
DATE	
SOURCE LANGUAGE	PAL III

Although this program has been tested by the contributor, no warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related program material, and no responsibility is assumed by these parties in connection therewith.

1. Program Title: Geometric Data Truncation for Fourier Transform Programs

2. Abstract: This program is intended for use as a truncating-weighting subroutine in conjunction with a Fourier Transform program. The application of a weighting function to the data record before the application of a Fourier transform program reduces the spread in frequencies which results from the transformation of a finite record. This procedure is often called apodization in that it removes the side lobes in the transformed record that results from a rectangular data window.

3. Requirements:
   3.1 Storage: The subroutine occupies 115(8) locations and is relocatable at the time of assembly.
   3.2 Hardware: PDP-8
   3.3 Software: The user must have a mainline program for gathering and storing in serial order the data before the application of this truncation algorithm. Following the weighting of the data the transform must be applied. The real series must be stored in serial order and without gaps. The length must be of the form 2^N, just as required by the FFT. Additionally, the data must be in signed 11 bit, fixed-point format.

4. Usage: The program is intended to be assembled along with the mainline program. It is to be called after the data are gathered and before the actual FFT. The weighted data are replaced in their original locations with the original contents being lost. The subroutine is called with the number of data points in the AC and the address of the first point in CALL +1. Return is to CALL +2.

5. Discussion: When performing the discrete Fourier transform on a finite record in the time domain an inherent distortion results in the frequency domain. The FFT gives the power spectral density of an observed signal. Denoting Fourier transforms by ($\overline{\quad}$) and convolutions by (*),

$$A(\nu) = \overline{A(t)} \quad \& \quad F(\nu) = \overline{F(t)}$$

where

$$A(\nu) = \int_{-\infty}^{\infty} A(t) e^{-i2\pi\nu t} dt$$

If F(t) is the temporal representation of the observed signal, the power spectral density is given by

$$F(\nu) = \left| \int_{-\infty}^{\infty} F(t)e^{-i2\pi\nu t}dt \right| = \left| \overline{F(t)} \right|$$

The data window here extends from $-\infty$ to $+\infty$ and weights each datum equally, i.e., $A(t) = 1$ for $-\infty \leqslant t \leqslant +\infty$.

The multiplication of two functions in the time domain is equivalent to convolving the transforms of the two functions in the frequency domain.

$$\overline{A(t)\cdot F(t)} = \overline{A(t)}*\overline{F(t)} = A(\nu)*F(\nu)$$

$$= \int_{-\infty}^{\infty} A(\nu - \nu')F(\nu)d\nu'$$

For the case of an infinite data window

$$A(\nu) = \overline{A(t)} = \int_{-\infty}^{\infty} 1\cdot e^{-i2\pi\nu t}dt = \delta(\nu - \nu')$$

Hence the power spectral density

$$F(\nu) = \left| \int_{-\infty}^{\infty} F(t)e^{-2i\pi\nu t}dt \right| = \left| \int_{-\infty}^{\infty} \delta(\nu - \nu')F(\nu)d\nu' \right|,$$

is the true power spectrum.

Infinite records are impossible and the observations of the temporal signal must be terminated. This limiting of F(t) is truncation.

The most common form of truncation is the rectangular data window. Observations are made for a certain period of time and all information before and after this period is neglected. This gives rise to an A(t) that is a rectangle of unit height. The Fourier transform, $A(\nu) = \frac{\sin \pi\nu}{\pi\nu}$, is a function with a finite spread and smaller side lobes, (see Fig 1A). The true spectrum is then convolved with a function which spreads each frequency into several frequencies.

Several other truncation functions have been used. These functions reduce the spread of the central peak and/or remove the side lobes. Two of these are the triangular function with $A(\nu) = \left(\frac{\sin \pi\nu}{\pi\nu}\right)^2$ (see Fig. 1B). and the Gaussian function with $A(\nu) = e^{-\pi\nu^2}$, (see Fig. 1C). The application of these functions involves a degree of data handling that may be overly time consuming in order to obtain a clean power density spectrum.

This program involves a geometric truncation which requires no multiplications, or divisions, and yet gives a smooth truncated signal to be transformed. In the frequency domain the function to be convolved with the true spectral function has no side lobes and a minimal width.

6. Description: The data record is first divided into 16 blocks. The binary representation of each datum within a block is shifted to the right a number of times dependent upon the position of the block in the data record. The components of the blocks would be shifted 7, 6, 5,..., 1, 0, 0, 1, 2,..., 6, 7 places, resulting in multiplications of the components by 1/128, 1/64, 1/32,...1/2, 1, 1, 1/2, 1/4,..., 1/64, 1/128 respectively. These factors are indicated by the horizontal bars in Fig. 2.

The truncation function when smoothed to its average value is the hyperbolic secant, sech (πt). The geometrical function has a number of discontinuities but at points of discontinuity the Fourier transform integral converges to the average

$$\lim_{\varepsilon \to 0} \frac{1}{2}[A(t + \varepsilon) + A(t - \varepsilon)]$$

of the right and left hand limits.

The Fourier transform of sech (πt) is sech $(\pi\nu)$, (see Fig. 1D). So the function to be convolved with the true spectrum has no side lobes and minimal width.

7. Execution Time: The time required to weight a data record is (100 μs) x (data record length). For example, a 512(10) point record length can be weighted in 0.0512 seconds.

8. User Modification: The user who intends to transform data records of a fixed length may wish to shorten the subroutine by calculating and storing some of the constants which would otherwise be recalculated and stored at each execution time.

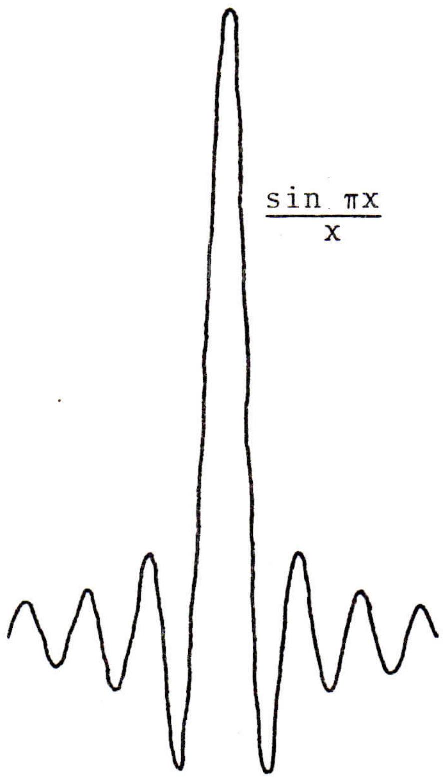
9. Listing: A listing is below.

10. Logic Flow Chart: A logic structure flow chart follows the listing.

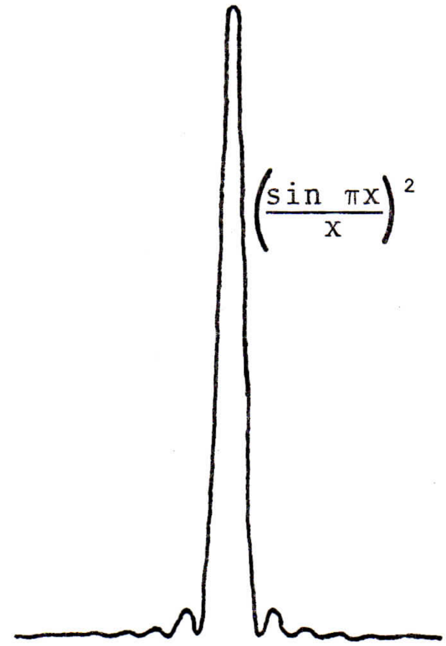
11. References:

(1) Bergland, G. D. "A Guided Tour of the Fast Fourier Transform", IEEE Spectrum, 6, 41 (1969).

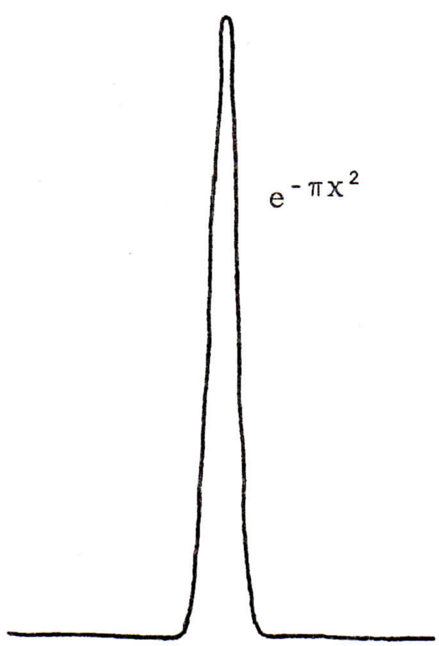
(2) Bingham, C., Godfrey and Tukey, "Modern Techniques of Power Spectrum Estimation." IEEE Trans. Audio Electroacoustics, AU-15, 56 (1967).



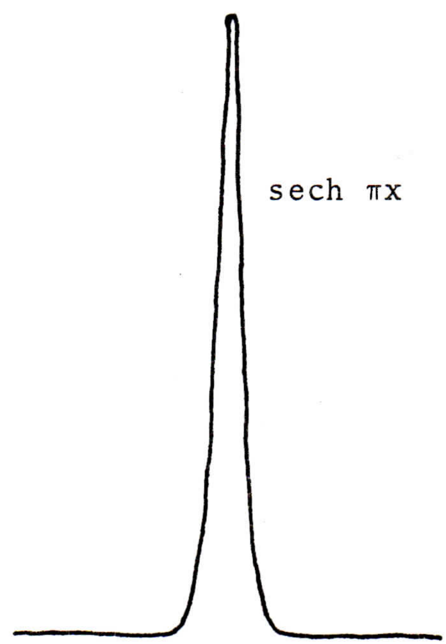
1A



1B

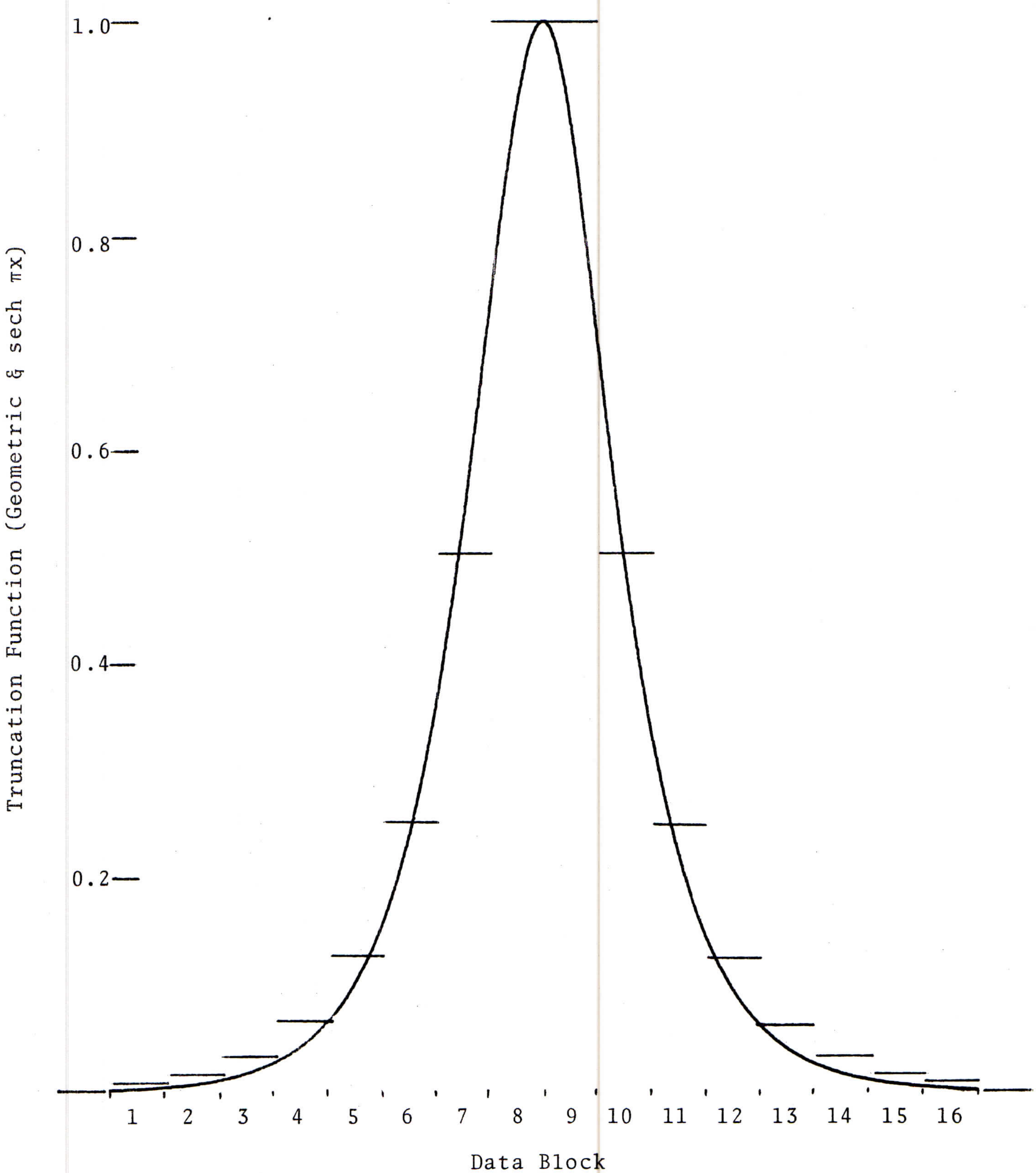


1C



1D

Figure 1



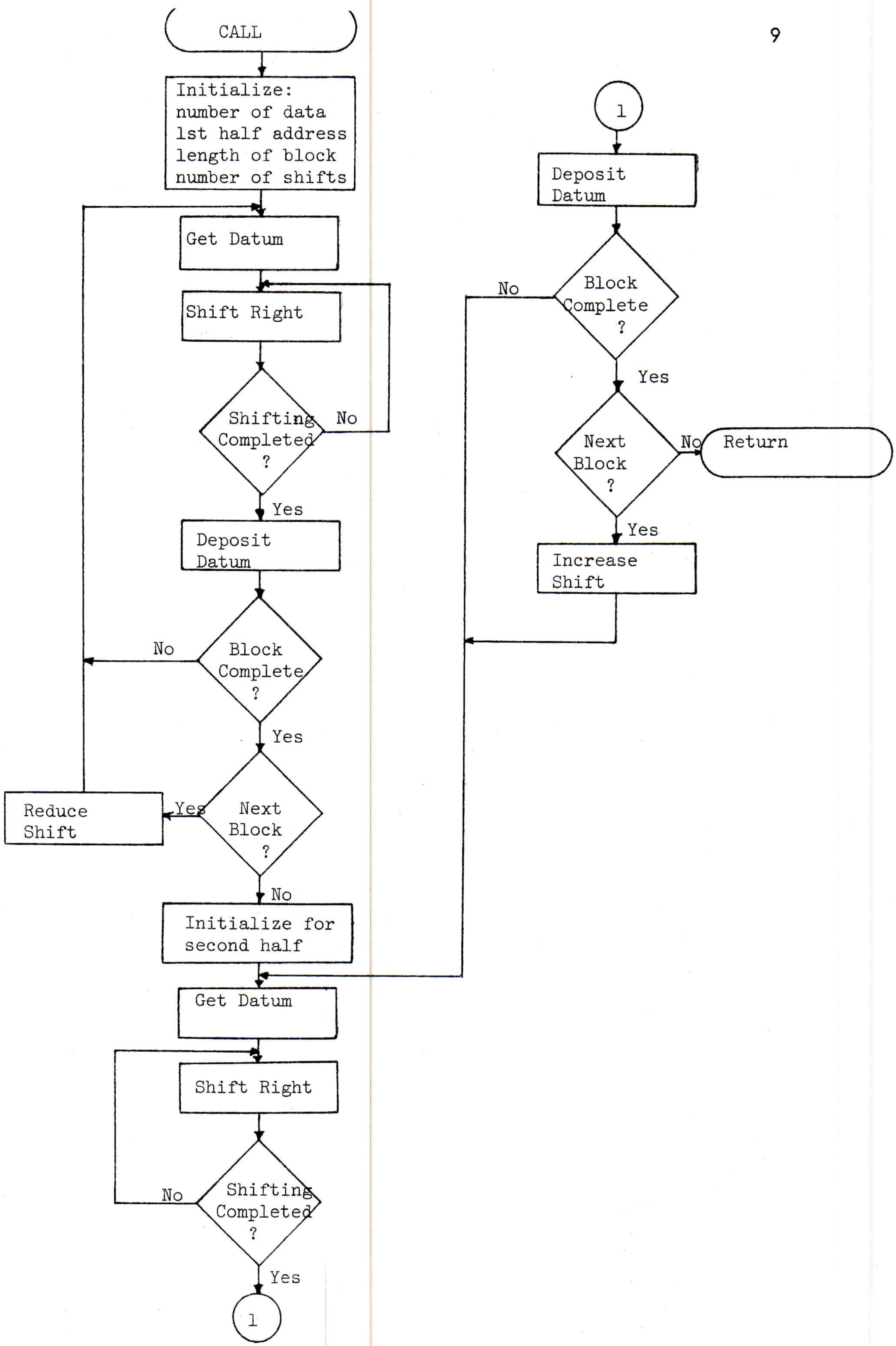
Truncation Function (Geometric & sech πx)
1.0
0.8
0.6
0.4
0.2
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Data Block

Figure 2

```
                         /GENERALIZED APODIZATION FUNCTION
                         /E.A. BARNHARDT, 30 SEPTEMBER 1971
                         /FOR DATA OF LENGTH 2↑N, THAT IS 1024, 512, 256, ETC
                         /RUNNING TIME , 100 MICROSEC TIMES DATA LENGTH
                         /DATA RESTORED TO ORIGINAL LOCATION.
                         /CALL WITH NUMBER OF DATA POINTS IN AC AND ADDRESS
                         /OF FIRST POINT IN CALL+1, RETURN IS TO CALL+2
                         /PROGRAM OCCUPIES 115(8) LOCATIONS
                         *4600
4600  0000  APOD,        0
4601  3313               DCA PTS           /NUMBER OF DATA POINTS (8)
4602  1600               TAD I APOD
4603  3311               DCA ADD1          /ADDRESS OF FIRST DATUM
4604  2200               ISZ APOD          /SET RETURN TO CALL +2
4605  1313               TAD PTS
4606  7012               RTR
4607  7012               RTR
4610  3302               DCA TEMP2         /LENGTH OF ONE DATA BLOCK
4611  1302               TAD TEMP2
4612  7041               CIA
4613  3306               DCA CNTR3         /MINUS BLOCK LENGTH
4614  1313               TAD PTS
4615  1306               TAD CNTR3
4616  1306               TAD CNTR3
4617  7041               CIA
4620  3307               DCA CNTR4         /ACTUAL # OF POINTS TO BE SHIFTED
4621  1313               TAD PTS
4622  7010               RAR
4623  1311               TAD ADD1
4624  1302               TAD TEMP2
4625  3312               DCA ADD2          /ADDRESS OF 2ND BLOCK OF 2ND HALF
4626  1314               TAD M7
4627  3303               DCA CNTR0         /MINUS 7, MINUS 6, MINUS 5, ...
4630  1306  LOOP1,       TAD CNTR3
4631  3310               DCA CNTR5
4632  1711  SLOOP1,      TAD I ADD1        /GET DATUM
4633  4265               JMS SPSR          /SHIFT
4634  3711               DCA I ADD1        /DEPOSIT SHIFTED DATUM
4635  2311               ISZ ADD1          /UPDATE ADDRESS
4636  2307               ISZ CNTR4         /ALL COMPLETE?
4637  2306               ISZ CNTR3         /NO, BLOCK COMPLETE?
4640  5232               JMP SLOOP1        /NO, GET NEXT DATUM
4641  2303               ISZ CNTR0         /YES, REDUCE SHIFT OR GO TO 2ND HALF
4642  5230               JMP LOOP1         /DO NEXT BLOCK
4643  7300               CLA CLL           /SECOND HALF INITIALIZATION
4644  7001               IAC
4645  7041               CIA
4646  3303               DCA CNTR0         /MINUS 1, MINUS 2, MINUS 3, ...
4647  1306  LOOP2,       TAD CNTR3
4650  3310               DCA CNTR5
4651  1712  SLOOP2,      TAD I ADD2        /GET DATUM
4652  4265               JMS SPSR          /SHIFT
4653  3712               DCA I ADD2        /DEPOSIT SHIFTED DATUM
4654  2312               ISZ ADD2          /UPDATE ADDRESS
```

```
4655  2307           ISZ CNTR4       /ALL COMPLETE?
4656  5260           JMP SKIP        /NO, SKIP
4657  5600           JMP I APOD      /YES, RETURN
4660  2310  SKIP,    ISZ CNTR5       /BLOCK COMPLETE?
4661  5251           JMP SLOOP2      /NO, GET NEXT DATUM
4662  1303           TAD CNTR0       /YES, INITIALIZE FOR NEXT BLOCK
4663  7041           CIA
4664  5247           JMP LOOP2
4665  0000  SRSR,    0               /SHIFT SUB. (MODIFIED DEC-08-FMJA-D)
4666  3301           DCA TEMP1       /STORE NUMBER
4667  1303           TAD CNTR0       /GET SHIFT COUNTER
4670  3315           DCA CNT         /STORE FOR LOCAL USE
4671  1301           TAD TEMP1
4672  7100  LOOP3,   CLL
4673  7510           SPA
4674  7020           CML
4675  7010           RAR
4676  2315           ISZ CNT
4677  5272           JMP LOOP3
4700  5665           JMP I SRSR
4701  0000  TEMP1,   0
4702  0000  TEMP2,   0
4703  0000  CNTR0,   0
4704  0000  CNTR1,   0
4705  0000  CNTR2,   0
4706  0000  CNTR3,   0
4707  0000  CNTR4,   0
4710  0000  CNTR5,   0
4711  0000  ADD1,    0
4712  0000  ADD2,    0
4713  0000  PTS,     0
4714  7771  M7,      7771
4715  0000  CNT,     0

ADD1     4711
ADD2     4712
APOD     4600
CNT      4715
CNTR0    4703
CNTR1    4704
CNTR2    4705
CNTR3    4706
CNTR4    4707
CNTR5    4710
LOOP1    4630
LOOP2    4647
LOOP3    4672
M7       4714
PTS      4713
SKIP     4660
SLOOP1   4632
SLOOP2   4651
SRSR     4665
TEMP1    4701
TEMP2    4702
```


CALL
Initialize:
number of data
1st half address
length of block
number of shifts
Get Datum
Shift Right
Shifting Completed ?
No
Yes
Deposit Datum
Block Complete ?
No
Yes
Next Block ?
Yes
Reduce Shift
No
Initialize for second half
Get Datum
Shift Right
Shifting Completed ?
No
Yes
1
1
Deposit Datum
Block Complete ?
No
Yes
Next Block ?
No
Return
Yes
Increase Shift