THE IMPLEMENTATION OF THE DISCRETE FOURIER TRANSFORM BY TABLE LOOK-UP TECHNIQUE

David Stephen Cohen

A Dissertation Submitted to the Faculty of Engineering, University of the Witwatersrand, Johannesburg, for the Degree of Master of Science in Engineering

Johannesburg 1976

I declare that this dissertation is my own work and has not previously been submitted for any degree at any University.

D. S. Cohen

ABSTRACT

The Development of a Discrete Fourier Transform Algorithm using the Table Look Up Method

The development of a system to compute the Discrete Fourier Transform (DFT) of a time sampled, wide dynamic range noiselike signal is reported in this dissertation. The mathematical basis of the algorithm is covered and an algorithm is then developed and programmed on a Nova 1200 minicomputer. The results are then compared with those of a Fast Fourier Transform Algorithm. (FFT)

Fourier analysis of continuous and sampled time signals is discussed and it is shown that the Fourier Series Coefficients of a sampled time sequence are related by a constant to the Discrete Fourier Transform of that time sequence. The Fourier Series Coefficients are considered to be a special case of the bivariate time average which is solved using a computational algorithm based on statistical averaging methods and a table look up method. The signal component of the bivariate average is sampled with a logarithmic analog to digital converter (LADC) thus allowing a wide dynamic range signal to be represented by a short word length. The problem of generating the sinusoidal component is overcome by spacing the abscissas in the statistical average numerical integration formula sinusoidally and generating linear in time addresses which correspond to the arguments of a sinusoidal wave.

The Fourier Series Coefficient algorithm is extended to compute Discrete Fourier Transforms and the linear address (argument)

generator is modified to generate addresses that correspond to all the frequencies extracted by the DFT. This algorithm is termed the Discrete Fourier Transform using the Table Look Up method (DFTTLU).

The basic features of this technique are summarized. The LADC output word and the generated address form an address pair which specifies a constant, in a stored table of constants, which is added to an accumulator. The accumulator is initially set to zero, and after the number of samples in a signal burst have been processed, will contain a number proportional to the harmonic, the frequency of which has been set by the generated addresses (arguments). The remaining harmonics are processed by setting the frequency of the generated addresses accordingly. The basic calculation step of this technique requires no multiplication. The algorithm is programmed in Assembler Code and is run on a minicomputer system that has a LADC peripheral.

An indication of the accuracy of the DFTTLU is obtained by comparing its results with those of an FFT algorithm of known accuracy analysing the same signal. The FFT is also programmed on the minicomputer and a 12 bit analog to digital converter peripheral (synchronised with the LADC) samples and encodes the signal. The differences between the results are less than 2% of the maximum harmonic amplitude for the worst case.

Although the aim of the work is the development of a real time Fourier Transform system that analyses noiselike signals, the system developed is slow because of the inherent complexity of the

Assembler Code needed to generate the addresses. It is envisaged that a hardware based system would simplify and overcome this problem.

ACKNOWLEDGEMENTS

This work has been supported by bursaries from the Council for Scientific and Industrial Research and the University of the Witwatersrand.

The author wishes to thank Professor Hanrahan who initiated and supervised the project and who offered many suggestions. Thanks are also due to Mr Rex van Olst for the use of the Nova Fast Fourier Transform Calculation, Mr Alistair Walker for the use of the double word square root routine and Mrs Towsey who typed the manuscript.

CONTENTS PAGE

CONTENTS (contd.) PAGE

CONTENTS (contd.) PAGE

1 FOURIER ANALYSIS OF SIGNALS USING A DIGITAL SIGNAL PROCESSING SYSTEM

1.1 Introduction

In recent years digital instrumentation, with its inherent reliab_lity, accuracy and ease of reading has rivalled, and in many cases surpassed analog instrumentation in most fields of electrical measurements. A glance at most electronic instrument makers' catalogues will show that digital voltmeters and frequency counters are common 'off the shelf' items while digital signal processing modules such as those performing the Fast Fourier Transform (FFT), designed for use as 'stand alone' apparatus, or add on peripherals for a mini-computer (. microprocesser), are entering the market in increasing numbers. The improvements in digital electronics technology and the resulting decrease in costs have given rise to digital systems designed to compute more complex electrical measurements such as time averages, correlation coefficients and Fourier Transforms of signals.

Simplicity and accuracy are two important aspects in the design of a digital signal measurement system. Most systems consist of;

1. An analog to digital converter (ADC) which samp[illegible] the amplitude of an analog signal with respect to time and represents this amplitude with a digital code word.

2. A digital processer which calculates the required measurement (eg time average) of the analog signal. The digital

processer can be a special purpose digital instrument or a computer programmed to perform the required measurement.

3. The output display displays the result of the measurement in a digital (eg LED or NIXIE display) or graphic (eg CRO or plotter display) form.

The accuracy of most existing digital signal processing systems relies in most cases, not on theoretical considerations, but on processing a large number of time samples to reduce sampling errors, and on finely quantised amplitude levels in the ADC to reduce amplitude quantisation errors. The resulting digital instrument is usually more accurate than its analog counterpart and less complex than an analog instrument of comparable accuracy. The major sources of error in most digital signal processing systems can be broadly classified as;

1. Time sampling errors
2. Amplitude quantising errors.
3. Rounding or truncating errors in the numerical algorithm used to calculate the measurement.

These errors are discussed in chapter 4 and chapter 2 of references 5 and 14 respectively.

Hanrahan (ref. 6) reviews some of the literature on these errors and presents a study of the underlying theory of time averages of amplitude quantised time sampled data. The particular approach used in this work leads to the development of a simple algorithm, based on a table look up (TLU) method and on statistical averages, to calculate time averages of signals. The basic features of the TLU method are as follows:

Samples of the signal being processed are quantised fairly coarsely. The quantiser output word is used as an address to specify a constant, in a table of constants, which is added to an accumulator. The accumulator, initially set to zero, contains a number proportional to the required average after a large number of samples have been processed. This technique can compute multivariate as well as single variate time averages. A simplified block diagram of a system to calculate a single variate average is given in Fig 1.1 .

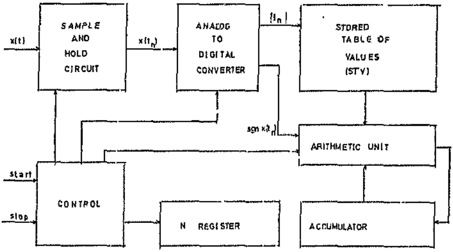


Fig.1.1 Single Variate Time Average System

This approach in implementing a digital processer has the following advantages:

1) A simple ADC is used. The ADC has fewer amplitude levels when compared with most commercially produced ADC[11] and less bits are needed to code an amplitude level.

2) The basic calculation step does not need a multipication operation.

These factors allow a relatively simple digital signal processing system to be developed to compute time averages such as:

1. Mean value
2. Mean square value
3. Correlation Coefficients
4. Autocorrelation Coefficients
5. Fourier Coefficients.

The table look up method is not exact because of the quantisation of the signals in both time and amplitude. The accuracy obtainable ranges from errors of a few per cent with relatively coarse amplitude quantisation (ie approximately 8 levels and 500 - 1 000 samples) to 0,1% error with 16 - 64 amplitude levels and up to 4 000 samples.

The TLU method does not restrict the ADC characteristic and consequently the ADC characteristic can be chosen to suit the properties of the signal being processed. Signals that have a wide dynamic range and a Gaussian or similar amplitude distribution (ie noise like signals) are of particular interest in the fields of acoustic and vibration analysis. These signals can be quantised logarithmically, so providing high resolution at low signal amplitudes and coarse resolution at high signal amplitudes. Hanrahan[10] describes a system that has a 19 level, 3dB spacing logarithmic analog to digital converter (LADC) to compute the mean and mean square values of noise signals to an accuracy of the order of 0,5dB. A logical extension of the work, suggested by Professor Hanrahan as the subject of this dissertation, is Fourier Analysis of logarithmically quantised noiselike signals, the aim of the work being the development of a real time on line digital Fourier Analyser that exploits the advantages of the LADC and the TLU method. This system could be used in acoustic signal analysis, low frequency vibration analysis or any application that requires the Fourier Transform of noiselike, wide dynamic range signals.

1.2 Review of the Literature

The literature pertinent to the TLU method, analog to digital converters and Fourier Analysis systems is reviewed in this section. An existing TLU system that calculates Fourier Coefficients is described and the LADC is discussed. A brief description of the state of the art in Fourier Analysis systems is given.

1.2.1 The Table Look Up Method and the Logarithmic Analog to Digital Converter

Hanrahan[6] describes a TLU system that calculates the Fourier Coefficients of deterministic signals which are quantised into eight positive and eight negative linearly spaced levels in this particular case The typical error is given as a percentage of the maximum valu fundamental which is defined to be 1. The errors are of t same order or better than those of a conventional heterodyne wave analyser (viz. 3 per cent full scale error and 6 per cent spurious response). A more complex system based[7] on the same principle reduces the errors to one per cent error and one per cent spurious response.

The concept of quantising wide dynamic range noiselike signals using a logarithmic analog to digital converter (LADC) is discussed in reference 8 and a TLU system incorporating an LADC to calculate mean and mean square values is described. The error analysis shows the errors to be of the order of 0,1 dB provided that a noiselike signal is processed. The error rises to approximately 2dB when a deterministic signal is processed. Applications of this system are discussed in reference 9 . The design considerations of a LADC computer peripheral and the computer programme needed to compute mean and mean square values of a noiselike signal is covered in reference 10. The LADC developed has 19 positive and 19 negative levels spaced 3dB apart.

1.2.2 The State of the Art in Fourier Analysis Systems

Fourier analysis systems can be divided into two types, analog and digital systems. The systems described here are just a few of the many systems available.

1.2.2.1 Analog Systems

The heterodyne wave analyser was used by Bell Telephone Laboratories[2] in the late forties as the basis of the 'Sona Graph' spectrum analyser. This electro-mechanical instrument, which is used to analyse speech signals, produces a frequency spectrogram of about 2,5 seconds of speech signals in about three minutes. The dynamic range of the spectrogram is about 12dB.

Today, the best description of the state of the art is found in electronic equipment catalogues.[3,12] For example, the specifications of the HP 3580A Spectrum Analyser are summarized below.[12]

Frequency Range:	5Hz to 50kHz
Dynamic Range	80dB
Maximum Error	1dB
Sweeptime	0,1 sec. to 2 000 sec.
Minimum Bandwidth	1Hz

A bandpass filter (1Hz minimum bandwidth) between the required lower and upper frequency limits and ter output is displayed on a CRT.

Another analog system is the Bruel and Kjaer[3] Type 3347 Real Time one third Octave Analyser. This system consists of 30 one third octave filters with centre frequencies from 25Hz to 20kHz. The filter outputs are scanned every 20ms and the results are displayed on a CRT in the form of a bar graph - the horizontal axis is the frequency axis. The dynamic range of the instrument is 64dB and the CRT displays a range of 50dB with a resolution of 0,2dB.

Generally the analog Fourier analysis instruments are complicated and require a calibration or setting up procedure before they can be used.

1.2.2.2 Digital Systems

Digital systems that perform Fourier Analysis can be divided into two types.

1. Systems that use linear filter theory or recursive techniques.[14]

2. Systems that use the Cooley Tukey Fast Fourier Transform (FFT).[5]

The FFT algorithm is used in virtually all digital systems. Most minicomputers with 4K of memory core and the ability to control an ADC peripheral can be programmed to analyse electrical signals using the FFT algorithm.

An example of a big commercially produced system that uses the FFT is the HP5451 series.[12] This consists of a minicomputer, keyboard, a dual 10 bit analog to digital converter, a display control unit and a CRT, and a software package to control the system. A Fourier Transform of 1024 time samples is produced in about 100ms. and has a dynamic range of 80dB and a frequency range of 0 - 100kHz.

At the other end of the scale there are the 'FFT modules', an example of which is the Plessey SPM-02. This is designed for use as a *'stand alone' or add on FFT processer with a minicomputer* or microprocesser. The module can process 1 024 complex points in 250ms with a frequency range of 0 - 50 kHz.

All the above examples must not be construed as a complete description as the state of the art, but they do give an indication of the different methods used in Fourier analysis.

1.3 The Present Work

The present work is concerned with the development of a digital system that exploits the advantages of the TLU method and the LADC to perform real time on line Fourier Transforms of noise signalsthat have a wide dynamic range. The system is implemented as a software programme that is run on a Nova 1200 minicomputer in conjunction with the 19 level, 3dB spacing logarithmic analog to digital converter.[10] *The speed and accuracy of the system are* investigated by varying certain parameters and comparing the results with those given by an algorithm of known accuracy (eg Cooley Tukey Fast Fourier Transform, FFT). The ultimate aim of this work is the development of a Fourier Analysis Instrument that can rival and, in some cases, replace existing digital (and analog) Fourier Analysis systems that process noiselike signals.

The flexibility of the computer facilitates changes to the algorithms in the light of results obtained in anal sing signals. The programme steps needed to implement the algorithm are analogous to the steps required if the algorithm is implemented in a special purpose hardware digital instrument.

1.4 Scope and Format of the Dissertation

This dissertation first covers the mathematical basis of the DFT TLU and then discusses the development of a computational algorithm. The algorithm is programmed on a Nova 1200 minicomputer and the results are compared with the results of an FFT algorithm.

Fourier analysis of continuous and time sampled signals is covered in Chapter 2 and it is shown that the Fourier Coefficients of a time sampled, periodic signal are related to the Discrete Fourier Transform of that time sampled signal. The Fourier Coefficients are also shown to be a special case of the bivariate time average which is solved using a computational algorithm based on statistical averaging methods and a table look up method. The problem of generating the sinusoidal component of the bivariate average is discussed and a simple linear in time sinusoidal channel address generation is developed.

In Chapter 3 the linear in time address generator is modified so that the Fourier Coefficient system can be extended to compute the Discrete Fourier Transform.

The algorithm that is programmed to solve the DFT TLU is discussed in detail in Chapter 4 and numerical values of harmonics and the machine code scale factors are derived and tabulated.

The DFTTLU is compared with the FFT in Chapter 5 in order to obtain an error characteristic. The results of the FFT are regarded as the 'true' value and the differences between the FFT and the DFT TLU are termed the 'errors'.

The conclusions and recommendations for further work are covered in Chapter 6.

Derivations or discussions that break the continuity of this dissertation are covered by the appendices at the end of this report. The generation of the linear in time addresses which replace sine or cosine waves needed for Fourier Transform Analysis is discussed in Appendices A3 and A4 and the derivation of the stored table of values(STV) is discussed in A1 and A2. A system that compares the results of the DFT TLU and an FFT algorithm is given in A6.

2. THEORY UNDERLYING THE APPLICATION OF THE TABLE LOOK UP METHOD TO THE DISCRETE FOURIER TRANSFORM

2.1 Introduction

This chapter reviews the definitions of the Fourier Series Coefficients and the Fourier Transform for continuous and sampled input data. In a sampled data system, where the signal is periodic, band limited and sampled at its Nyquist Rate, the complex Fourier Coefficients are shown to be samples of the continuous Fourier Transform taken at integral multiples of a frequency proportional to the sampling rate. It is then shown that the complex Fourier Coefficients are proportional to the Discrete Fourier Transform (DFT) points of the sampled signal. The complex Fourier Coefficients are a special case of the bivariate time average[5] and are solved with statistical averages and the table look up technique. In this dissertation the theory that shows how the table look up method can be used to solve a bivariate statistical time average is not given as this is covered in appendices 3 and 7 of reference 6.

2.2 The Fourier Series Coefficients and the Fourier Transform

For a continuous, periodic, real time function x(t) of period T_1, the general equation of the Fourier Series can be written as follows:

$$x(t) = \frac{a_o}{2} + \sum_{r=1}^{\infty} a_r \cos(2\pi r f_1 t) + b_r \sin(2\pi r f_1 t) \qquad 2.1$$

r is the order of the Fourier Series Coefficients and $f_1 = 1/T_1$.

The Fourier Series Coefficients are written as follows:

$$a_r = \frac{2}{T_1} \int_{-T_{1/2}}^{T_{1/2}} x(t) \cos (2\pi r f_1 t) \, dt. \qquad 2.2$$

$$b_r = \frac{2}{T_1} \int_{-T_{1/2}}^{T_{1/2}} x(t) \sin (2\pi r f_1 t) \, dt \qquad 23$$

Equations 2.2 and 2.3 can be written in the exponential form

$$c_r = \frac{1}{T} \int_{-T_{1/2}}^{T_{1/2}} x(t) e^{-j2\pi r f_1 t} \, dt \qquad 2.4$$

ie $$c_r = \frac{a_r - jb_r}{2} \qquad 2.5$$

The Fourier Transform can be written as

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} \, dt \qquad 2.6$$

Consider a burst waveform $x(t)$ of duration NT. By definition as $x(t)$ is time limited, its Fourier Transform cannot be band limited. Hence it cannot be sampled fast enough to completely prevent aliasing. However, if the sampling interval T is small enough, aliasing can be reduced to negligible extent. Under these circumstances the interpolation[5] (sampling) theorem may be applied, and it would be possible to reconstitute $x(t)$ almost exactly from its samples $x_s(nT)$, i.e.

$$x(t) = \sum_{n=0}^{N-1} \frac{\sin \pi \left(\frac{t - nT}{NT}\right)}{\pi \left(\frac{t - nT}{NT}\right)} x_s(nT) \qquad 2.7$$

where N is a finite number. If the signal burst is repeated to give $x_p(t)$, the Fourier series coefficients of $x_p(t)$ can be computed. The periodic repetition of the sampled burst can be written as

$$x_p(t) = \sum_{l=-\infty}^{\infty} \sum_{n=0}^{N-1} \frac{\sin \frac{\Omega}{2}(t - nT - lNT)}{\frac{\Omega}{2}(t - nT - lNT)} x_s(nT) \qquad 2.8$$

$$\Omega = 2\pi / N T \qquad 2.9$$

In this case the complex Fourier Series Coefficients are denoted D_k.

$$D_k = \frac{1}{NT} \int_0^{NT} x_p(t) e^{-jk\Omega t}\, dt \qquad 2.10$$

$$D_k = \frac{1}{NT} \int_0^{NT} \sum_{l=-\infty}^{\infty} \sum_{n=0}^{N-1} \frac{\sin \frac{\Omega}{2}(t - nT - lNT)}{\frac{\Omega}{2}(t - nT - lNT)} x_s(nT) e^{-jk\Omega t}\, dt \qquad 2.11$$

$$D_k = \frac{1}{NT} \sum_{n=0}^{N-1} \int_0^{NT} x_s(nT) e^{-jk\Omega t} \sum_{l=-\infty}^{\infty} \frac{\sin \frac{\Omega}{2}(t - nT - lNT)}{\frac{\Omega}{2}(t - nT - lNT)} dt \qquad 2.12$$

$$D_k = \frac{1}{NT} \sum_{n=0}^{N-1} x_s(nT) e^{-j\Omega Tnk} \qquad 2.13$$

The DFT of $x_p(nT)$ exists and is

$$X(k\Omega) = \sum_{n=0}^{N-1} x_p(nT) e^{-j\Omega Tnk} \qquad 2.14$$

$$= \sum_{n=0}^{N-1} x_s(nT) e^{-j\Omega Tnk} = X_k \qquad 2.15$$

From equations 2.13 and 2.15

$$NTD_k = X_k \qquad 2.16$$

The kth Fourier series coefficient D_k of a periodic or repeated burst waveform sampled sufficiently fast to reduce aliasing to a negligible level, multiplied by the waveform period or burst duration NT, is equal to the kth harmonic of the DFT, X_k, of that waveform.

It is meaningless to talk of the Fourier Transform of a sequence of samples[5] but a sequence of discrete numbers $x_s(nT)$ with a time function

$$f(t) = \sum_{n=-\infty}^{\infty} x_s(nT)\,\delta(t - nT) \qquad 2.17$$

has a Fourier Transform.

$$F(\omega) = \sum_{n=-\infty}^{\infty} x_s(nT)e^{-j\omega nt} \qquad 2.18$$

The time function of equally spaced impulses $\delta(t - nT)$, has a transform that is periodic in frequency with period $\frac{2\pi}{T}$ In most cases the DFT of a sampled time function will only approximate the Fourier Transform of the time function (cf equations 2.14 and 2.18) but the properties of the DFT and the Fourier Transform correspond exactly. However there are only N distinct numbers in the DFT and only N different frequencies can be distinguished. These frequencies are denoted

$$\text{frequency} = k\Omega, \quad k = 0,1,2 \ldots\ldots N-1 \qquad 2.19$$

where $\Omega = \frac{2\pi}{NT}$ and T = sampling period

The complex numbers NTD_k can be regarded as samples of the Fourier Transform taken at integral multiples of the frequency Ω and a 'continuous Fourier Transform Spectrum' could be approximated using the interpolation formula

$$X(f_1) = NT \sum_{q=-\infty}^{\infty} D(qf_1) \frac{\sin\pi(f - qf_1)NT}{\pi(f - qf_1)NT} \qquad 2.20$$

where $f_1 = \dfrac{\Omega}{2\pi}$

Algorithms that compute Fourier Series Coefficients a_r, b_r, c_r are of great significance as they can be used to compute a frequency spectrum of a burst like phenomena.

2.3 Expressing the Fourier Series Coefficients as a Bivariate Time Average

Consider equation 2.15

$$X_k = \sum_{n=0}^{N-1} x_s(nT)\, e^{-j\Omega Tnk} \qquad 2.15$$

$$\therefore\ Re(X_k) = \sum_{n=0}^{N-1} x_s(nT) \cos\Omega Tnk \qquad 2.21$$

$$\text{now } X_k = NTD_k \qquad 2.16$$

$$Re(D_k) = \frac{1}{NT} \sum_{n=0}^{N-1} x_s(nT) \cos\Omega TnK \qquad 2.22$$

$$= \overline{x_s(nT) \cos\Omega TnK} \qquad 2.23$$

$$Re(D_k) = \overline{x_s(t_n)\cos k\Omega t_n} \qquad 2.24$$

Similarly for $Im(D_k)$

$$Im(D_k) = \overline{x_s(t_n)\sin k\Omega t_n} \qquad 2.25$$

2.4 The Evaluation of the Time Average as a Statistical Average

In reference 6 a versatile method of calculating time averages, using amplitude quantised time sampled data, and based on statistical averages and equivalent to the required bivariate time average, has been developed. This method is applied to the estimation of D_k. The general average, expressed as a statistical average[13] is

$$\overline{xy} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} XY\, p_{xy}(X,Y)\, dXdY \qquad 2.26$$

where $p(X,Y)$ is the joint probability density function of X and Y.

$$\overline{xy} = \int_{-X_L}^{X_L}\int_{-Y_L}^{Y_L} \mathrm{sgn}(X).\ \mathrm{sgn}(Y)\, P_{xy}(X.Y)\, dXdY \qquad 2.27$$

The intermediate mathematical steps between equations 2.26 and 2.27 are detailed in Appendix 3 in reference 6.

$\pm X_L$ and $\pm Y_L$ are the amplitude limits which $x(t)$ and $y(t)$ are assumed not to exceed. The distribution function, $P_{xy}(X,Y)$ is defined as the joint probability that $x(t)$ is more remote from zero than X AND $y(t)$ is more remote from zero than Y.

sgn(X) is interpreted as

$\mathrm{sgn}(X) = 1 \quad X \geqslant 0+$

$\mathrm{sgn}(X) = -1 \quad X \leqslant 0-$

Equation 2.27 is the basis for developing the table look up algorithm (TLU) to solve D_k. Numerical integration is performed with respect to one variable (say X) while the other (Y) is held constant at some value Y_r.

$$\alpha(Y_r) = \int_{-X_L}^{X_L} \mathrm{sgn}(X)\, P_{xy}(X,Y_r)dX \qquad 2.28$$

This is estimated with a numerical integration formula

$$\int_{-X_L}^{X_L} f(X)dX \doteq \frac{h_x}{D_x}\sum_{i=0}^{M_x-1} w_i\; f(X_i) \qquad 2.29$$

The only constraint on the quantising of the X channel signal is a one to one correspondence between the M_x quantiser thresholds, X_i, in the interval $0 \leqslant X \leqslant X_L$ and the abscissas in the numerical integration formula. h_x and D_x are integration constants and w_i are the integration weights of the formula. The thresholds on the negative axis are:

$$X_{-i} = -X_i \;;(W_{-i}=-W_i) \qquad i = 0,1 \;\ldots\; M_x-1$$

The Y channel is similar and the integration weights are denoted u_j

Under these conditions, the integrand, $P_{xy}(X,Y)$, can be estimated as the ratio of the number of sample pairs, C_{ij}, which fell more remote from zero than X_i and Y_j respectively, to the total number of sample pairs, N

ie. $$P_{xy}\,(X,Y) \doteq \frac{C_{ij}}{N} \qquad 2.30$$

Equation 2.27 can now be written

$$\overline{xy} \doteqdot \frac{h_x h_y}{D_x D_y N} \sum_{i=0}^{M_x-1} \sum_{j=0}^{M_y-1} \operatorname{sgn}(X_i)\ \operatorname{sgn}(Y_j)\ \ w_i u_j C_{ij} \qquad 2.31$$

N samples of $x(t)$ and N samples of $y(t)$ are taken at sampling instants t_n, $n = 0, 1, 2 \ldots N-1$, and the amplitude of the samples are quantised according to the above constraints. At the end of each sampling period C_{ij} is calculated and the summation in equation 2.31 is performed. The algorithm suggested by equation 2.31 is wasteful as $4M_xM_y$ sample totals C_{ij} have to be stored and a summation must be performed N times. In appendix 7 of reference 6 equation 2.31 is further simplified and is written

$$\overline{xy} \doteqdot \frac{h_x h_y}{D_x D_y N} \sum_{n=0}^{N-1} \operatorname{sgn}\ (x(t_n), y(t_n))\ S\,(I_n K_n) \qquad 2.32$$

$S(I_n\ K_n)$ is a two dimensional array and each element in the array is calculated according to:

$$S(s,r) \quad = \sum_{i=0}^{s} \ \sum_{j=0}^{r} w_i . u_j \qquad s \ = \ 0,1 \ \ldots.\ M_x-1$$

$$r \ = \ 0,1 \ \ldots.\ M_y-1$$

I_n and K_n are the digital outputs of the ADCs operation on $x(t_n)$ and $y(t_n)$ respectively. I_n is equal to the subscript of the largest X abscissa less than or equal to $x(t_n)$ and K_n is equal to the subscript of the largest Y abscissa less than or equal to $y(t_n)$. The algorithm embodied in equation 2.32 is one of addressing particular locations in a store (table) containing an array of constants $S(s,r)$. An accumulator is initially set to zero.

When the two samples are taken at time t_n I_n and K_n are obtained. These values, acting as addresses, specify that the constant $S(I_n K_n)$ held in address location $(I_n\ K_n)$ in the stored table of values, is to be added to the accumulator. Signed addition is necessary, the sign being obtained from $\mathrm{sgn}(x(t_n).y(t_n))$After N address pairs have been similarly processed the accumulator will contain the value

$$\frac{\overline{xy}\ D_x D_y N}{h_x h_y}$$

A block diagram of a system to implement this algorithm is given in Fig.2.1.

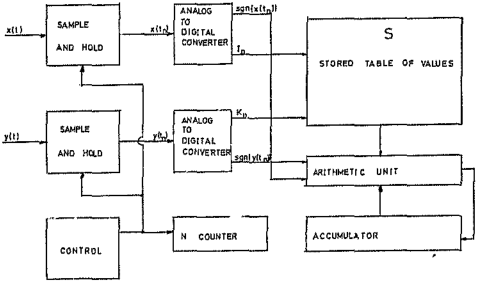


Fig. 2.1 System to Estimate $\overline{xy}$

2.5 Applying the Statistical Average using the TLU method to the Discrete Fourier Transform

The Fourier Coefficients can be estimated with equation 2.32

$$\overline{xy} = \frac{h_x h_y}{D_x D_y N} \sum_{n=0}^{N-1} sgn(x_s(t_n).\ y(t_n))\ S(I_n K_n) \qquad 2.32$$

$$Re(D_k) = \frac{h_x h_y}{D_x D_y N} \sum_{n=0}^{N-1} sgn(x_s(t_n).\ y(t_n))\ S\ (I_n K_n) \qquad 2.33$$

where $y(t_n) = \cos(\frac{2\pi nk}{N}) \quad k = 0 \ldots N-1$

$$Im(D_k) = \frac{h_x h_y}{D_x D_y N} \sum_{n=0}^{N-1} sgn(x_s(t_n).\ y(t_n))\ S(I_n K_n) \qquad 2.34$$

where $y(t_n) = \sin(\frac{2\pi nk}{N}) \quad k = 0 \ldots N-1$

To simplify the equations, T, the sampling period is defined to be one unit of time and equation 2.8 becomes

$$\Omega = \frac{2\pi}{N} \qquad 2.35$$

Equation 2.16 is now written as

$$ND_k = X(k\Omega) = F_k = X_k \qquad 2.36$$

F_k *is the Discrete Fourier Transform implemented* with the table look up method (DFTTLU).

Equations 2.33 and 2.34 are written

$$F_k = \frac{h_x h_y}{D_x D_y} \sum_{n=0}^{N-1} sgn(x_n . y_n)\ S(I_n K_n) \qquad 2.37$$

$$y_n = \cos(\frac{2\pi nk}{N}) \quad \text{for } Re(F_k) \qquad 2.38$$

$$y_n = \sin(\frac{2\pi nk}{N}) \quad \text{for } Im(F_k) \qquad 2.39$$

The N harmonics can be calculated by setting k to 0,1 ... N - 1 and solving equations 2.37 and 2.38.

ie. If x_n is a band limited signal sampled at its Nyquist Rate, F_k, $k = 0,1 \ldots N-1$ can be regarded as samplesof the Fourier Transform taken at *integral multiples of the frequency* $\Omega = \frac{2\pi}{N}$.

The signal in the Y channel is deterministic and K_n follows a deterministic pattern. K_n could be generated with a digital sinusoidal address generator. The different frequencies of K_n are *denoted* K_{nk}. *The subscripts nk correspond* to the nk in $\sin\left(\frac{2\pi nk}{N}\right)$.

If the Y channel abscissas (cf. eqn 2.39) follow a sinusoidal pattern, and the integration weights, u_j, and constants h_y and D_y are *adjusted accordingly*, K_{nk} becomes *equivalent to the arguments* of the sinusoidal function and has a LINEAR in time characteristic. Thus there is no need for any kind (analog or digital) sinusoidal generator. If the symmetrical properties of the sinusoid are exploited, K_{nk} need only have values *that correspond to arguments* between 0 and $\frac{\pi}{2}$ radians The Y channel contribution to the signum function is the sign associated with the quadrant in which the argument falls. Consider the set of addresses (K_{nk} arguments) required by a sine wave. The output of K_{nk} *address generator, in sign/magnitude form*, must correspond to the cyclic state starting at B in Fig.2.2

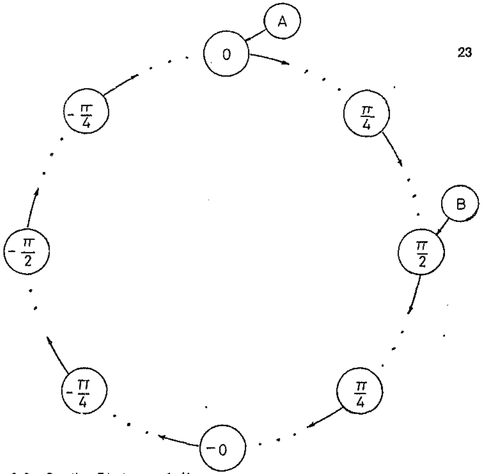


Fig. 2.2 Cyclic States of K_{nk}

The magnitude of the signed address generator (SAG) output forms the Y channel component of the address pair and the sign bit forms the Y channel component of the signum function. (This is similar to the logarithmic analog to digital convertor output.) In this dissertation a negative signed address (argument) will never mean that an angular phasor has been displaced clockwise on a phasor diagram but that the sine or cosine of a negative argument has a negative magnitude. A cosine wave is regarded as a sine wave delayed by $\frac{\pi}{2}$ radians. This corresponds to starting the SAG at state A in Fig.2.2. The SAG can be considered a modified UP / DOWN counter which is simple to implement using TTL hardware.[15] The system in Fig.2.1 is modified to incorporate the SAG and the system in Fig.2.3 results.

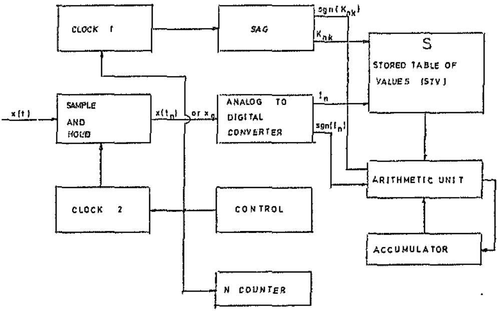


Fig. 23 System to Estimate F_k

This system can only calculate one harmonic at a time. Clock 1 sets the frequency (k) of the required harmonic and Clock 2 sets the sampling rate. The Accumulator is initially set to zero. At each sampling instant, t_n, the output of the quantiser, I_n and the output of the SAG, K_{nk}, form the address pair, $(I_n\ K_{nk})$ which extracts the constant, $S(I_n\ K_{nk})$ *from location* $(I_n\ K_{nk})$. This constant is either added or subtracted to or from the Accumulator, depending on the signum function. After N samples, the Accumulator contains a value proportional to the DFT frequency set by Clock 1.

In A7 a simple 8 point DFTTLU example shows the linear in time K_{nk} address generator. The signal is linearly quantised and a sine wave is analysed. The results are not exact because of the simplicity of the 'system'.

2.6 Discussion

This chapter has set out the relationship between the Fourier Series Coefficients and the Fourier Transform and has shown that the Fourier Series Coefficients of a sampled, periodic or repeated burst waveform can be regarded as samples of the Fourier Transform of the continuous signal provided that the sampling rate is high enough to prevent aliasing. The properties of the Fourier Transform are valid for the DFT if the signal is band limited and sampled at or above its Nyquist rate. The Fourier Series coefficients are a special case of the bivariate time average which, in turn, is solved with statistical average methods and the table look up method. The TLU method, which requires no multiplication in its basic calculation step can be used to compute the DFT which normally needs complex multiplication and addition.

Finally the flexibility of the TLU techniques is demonstrated by the modification of the X and Y channel abscissas to suit the 'signal' being processed. In the X channel, the abscissas are spaced logarithmically to process noiselike signals with a wide dynamic range. In the Y channel, the abscissas are spaced sinusoidally and there is no need for a sine wave generator.

3 COMPUTING THE DISCRETE FOURIER TRANSFORM FROM THE FOURIER SERIES COEFFICIENTS

3.1 Introduction

If a real band limited signal is sampled at its Nyquist Rate and N of the samples are processed, the DFT allows N distinct frequencies to be extracted.[5] These frequencies are denoted $k\Omega$ where

$\Omega = \frac{2\pi}{NT}$ T = sampling period

The frequencies $\frac{N}{2}$ .... $(N - 1)$ are mirror images of frequencies 0 .... $(\frac{N}{2} - 1)$

The system in 2.5 can extract N distinct frequencies(one at a time) by setting Clock 1 to generate addresses that are equivalent to the sinusoidal waves of frequencies $k\Omega$, $k = 0, 1 \ldots\ldots (N - 1)$. The output of the signed address generator (SAG) is denoted by K_{nk}, where K is the signed address generated at time n to extract the k th harmonic. The subscript, nk, is analogous to the nk in the angular argument used in the normal DFT notation.

$$\text{eg. } \sin\left(\frac{2\pi nk}{N}\right) \quad \text{or} \quad \cos\left(\frac{2\pi nk}{N}\right)$$

N^2 address pairs must be generated to extract N real harmonics and N^2 address pairs must be generated to extract N imaginary harmonics, in order to compute the DFTTLU. The term 'harmonics' is used loosely in this dissertation and will refer to the frequency domain results of the DFTTLU (ie. the N output points of the DFTTLU). Thus the major problem in implementing the DFTTLU is the simple and efficient operation of the $2N^2$ signed addresses in the Y (sinusoidal) channel.

To clarify the discussion in this chapter, the calculation steps needed to implement an 8 point DFTTLU are presented and methods of generating the signed addresses are discussed with the aid of a DFTTLU system block diagram.

3.2 A System to Implement the Discrete Fourier Transform usin, the Table Look Up Method

A simple DFTTLU system is discussed to facilitate the explanation of the signed address generator (SAG). I_n is the digital encoded value of the sampled signal amplitude at time $n(n = 0 \ldots\ldots N - 1)$. K_{nk} is the output of the SAG. The system only extracts imagi ary harmonic components (ie. the addresses, K_{nk}, correspond to the arguments of a sine wave). The N^2 calculation steps are detailed in Fig.3.1, and the block diagram of the system to implement these steps is given in Fig. 3.2.

Examination of Fig.3.1 reveals that the nth row of K_{nk} addresses is the same as the nth column of K_{nk} addresses. The DFTTLU calculation steps can thus be solved row by row or column by column. The row by row method or 'Rows Method' is similar to the algorithm used to calculate Fourier Coefficients; the procedure simply being repeated N times. The Rows Method is detailed in 3.2.1. The Columns Method implements the calculation steps by generating columns of address pairs and adding the contents of the specified address to the corresponding Accumulator ie Each address pair in a column of address pairs is associated with a different harmonic and the Accumulator associated with the particular harmonic is incremented. The columns method is described in 3.2.2.

Some notes on the DFTTLU system in Fig.3.2.

1. Only imaginary components of the harmonics are extracted.

2. Initially the Accumulators are set to zero.

3. The controller allows N samples of the signal to be taken and the coded amplitude values are stored in the N location I_n array.

4. The SAG is set to state B (cf. Fig.2.2) before a new row or column of K_{nk} addresses are generated.

5. Strobing the SAG causes a K_{nk} value to be generated according to the value of index k.

3.2.1 The Rows Method

Consider Fig.3.2, which shows a system having eight elements for purposes of explanation (see A7). The control is programmed to calculate each harmonic separately as in the system described in section 3.2

1) The controller sets the required frequency index on the address generator and switches the corresponding Accumulator on to the adder.

2) The controller strobes the SAG and addresses the first location in the I_n array. The output of the SAG and the array form the first address pair, $I_o K_{ok}$.

$$F_0 \propto S(I_0K_{00}) + S(I_1K_{10}) + S(I_2K_{20}) + S(I_3K_{30}) + S(I_4K_{40}) + S(I_5K_{50}) + S(I_6K_{60}) + S(I_7K_{70})$$

$$F_1 \propto S(I_0K_{01}) + S(I_1K_{11}) + S(I_2K_{21}) + S(I_3K_{31}) + S(I_4K_{41}) + S(I_5K_{51}) + S(I_6K_{61}) + S(I_7K_{71})$$

$$F_2 \propto S(I_0K_{02}) + S(I_1K_{12}) + S(I_2K_{22}) + S(I_3K_{32}) + S(I_4K_{42}) + S(I_5K_{52}) + S(I_6K_{62}) + S(I_7K_{72})$$

$$F_3 \propto S(I_0K_{03}) + S(I_1K_{13}) + S(I_2K_{23}) + S(I_3K_{33}) + S(I_4K_{43}) + S(I_5K_{53}) + S(I_6K_{63}) + S(I_7K_{73})$$

$$F_4 \propto S(I_0K_{04}) + S(I_1K_{14}) + S(I_2K_{24}) + S(I_3K_{34}) + S(I_4K_{44}) + S(I_5K_{54}) + S(I_6K_{64}) + S(I_7K_{74})$$

$$F_5 \propto S(I_0K_{05}) + S(I_1K_{15}) + S(I_2K_{25}) + S(I_3K_{35}) + S(I_4K_{45}) + S(I_5K_{55}) + S(I_6K_{65}) + S(I_7K_{75})$$

$$F_6 \propto S(I_0K_{06}) + S(I_1K_{16}) + S(I_2K_{26}) + S(I_3K_{36}) + S(I_4K_{46}) + S(I_5K_{56}) + S(I_6K_{66}) + S(I_7K_{76})$$

$$F_7 \propto S(I_0K_{07}) + S(I_1K_{17}) + S(I_2K_{27}) + S(I_3K_{37}) + S(I_4K_{47}) + S(I_5K_{57}) + S(I_6K_{67}) + S(I_7K_{77})$$

Fig. 3.1 Calculation Steps for 8 point DFTTLU

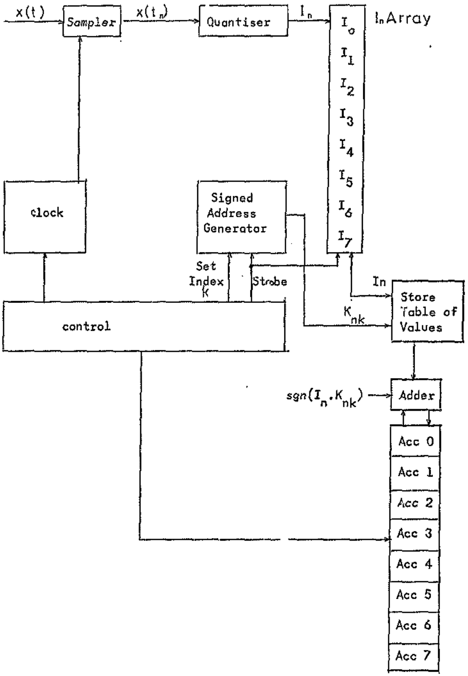


Fig.3.2 Block Diagram of System to Perform Spectral Analysis

3) The contents of the address, I_oK_{ok} in the STV, denoted $S(I_oK_{ok})$ are added to the Accumulator.

4) The controller again strobes the SAG and addresses the next location in the I_n array. The two outputs form the next address pair.

5) The contents of this address in the STV are then added to the Accumulator.

6) The above procedure is repeated for the remaining address pairs and the final value in the Accumulator is

$$S(I_oK_{ok}) + \ldots\ldots + S(I_7K_{7k}) \propto F_k$$

7) The controller sets the next frequency index k on the SAG and switches the corresponing Accumulator on to the adder.

8) The above procedure is repeated for all the remaining harmonics.

3.2.2 <u>The Columns Method</u>

Consider Fig.3.1. Column i of the signed addresses is the same as row i. If the frequency index, k, is set to zero and the SAG is strobed N times, the first column of signed addresses will be generated. In any column of address pairs, the I_n component does not change but each value in the STV addressed by one of these address pairs must be added to its corresponding Accumulator eg. the value obtained by the second address pair in a column must be added to the second Accumulator in Fig.3.2. To solve the DFTTLU column by column the controller in Fig.3.2 must carry out the following steps:

1) The controller sets the index k on the SAG.

2) The controller selects the value of I_n in the array where n = k and switches it through to one of the address inputs of the STV.

3) The controller switches the first Accumulator on to the adder and strobes the SAG.

4) The first address pair, $I_o K_{ok}$, obtains the contents of the STV and adds it to the first Accumul tor.

5) The controller switches the second Accumulator on to the adder and strobes the SAG.

6) The second address pair obtains the corresponding contents of the STV and adds it to the second Accumulator.

7) This procedure is repeated for the remaining Accumulators.

8) The remaining columns of address pairs are processed similarly.

3.3 Discussion on the Columns and Rows Methods

The Columns method is less complicated to programme as only the K_{nk} component of the address pair changes as each column of address pairs is generated. T' e disadvantage is that a different Accumulator must be incremented or decremented for each of the address pairs in the column. This problem is overcome by linking the selection of the Accumulators to the strobing of the SAG and the required Accumulator will be selected as each of the K_{nk} values are generated. The Columns method becomes a real time operation if a column of address pairs can be processed in less than the sampling period and, after the N columns of address pairs have been processed, the Accumulators will contain total proportional to the DFT points. The I_n array is not needed as the column of address pairs can be processed as soon as the I_n value is available from the LADC.

The Nova 1200 instruction set execution times do not permit a column of K_{nk} values to be generated in less than the sampling period and real time operation cannot be achieved. The use of TTL hardware to achieve real time operation is discussed in 6.2. The I_n array must be retained to store the N time samples of the signal burst. The non-real time algorithm and Assembler programme used to process the signal burst is described in A3 and A4 respectively. The most important details of A3 and A4 are summarized here for completeness.

From Fig. A3.3 it can be seen that if the first row and the first column are not counted, the K_{nk} addresses repeat after the row or column in which k or n equal $\frac{N}{4}$ or, in some cases, $\frac{N}{8}$. Thus only the first $\frac{N}{4}$ or $\frac{N}{8}$ columns of address pairs need be generated. The I_n values that share the same columns of K_{nk} values are shuffled out of the I_n array and are stored in a sub-array. These values of I_n follow a similar pattern to the 'bit reversed' pattern[5] of the FFT and are shuffled in a similar manner. Processing of the DFTTLU cannot begin until all N samples of the signal have been taken and stored in the I_n array of N locations. The advantage of the modified Columns method is that one column of K_{nk} addresses can be used to generate 4 or 8 columns of address pairs.

The Rows method is the same as the method used to calculate the Fourier Coefficients and this method is repeated N times to solve the DFTTLU. In each row the I_n and K_{nk} component of the address pair changes N times and this complicates the programming. Thus the modified columns method is chosen as the basis of the algorithm.

4 THE DETAILED DESCRIPTION OF THE DISCRETE FOURIER TRANSFORM ALGORITHM USING THE TABLE LOOK UP METHOD

4.1 Introduction

The derivation of the integration weights, STV, algorithm and Assembler programme is given in A1, A2, A3 and A4 respectively. The important details of these appendices are given in this chapter under the general headings of:

1) The definition of the amplitude quantising levels (the abscissas of the integration formulae), and the resulting integration weights and constants w_i, u_j, h_x, h_y.

2) The derivation of the STV, the machine code values that correspond to the STV and the resulting scaling factors D_x and D_y.

3) The numerical values of the DFTTLU harmonics and the overall machine code scale factor. The overflow and underflow of the Accumulators is considered and the STV and overall scale factor are adjusted to solve this problem. The overall scale factor for the different values of N and for the adjusted STV are tabulated at the end of this chapter.

4.2 The System

The system has two inputs, the X and the Y channel. The X channel consists of N samples of the noise signal which are quantised logarithmically and encoded in the sign magnitude form by the LADC. The LADC has 19 levels spaced at 3dB intervals and the output word has values between 0_8 and 22_8. A negative input signal is indicated by setting the most significant bit of the word (Bit zero on the Nova 1200) to one. The output of the LADC (excluding the sign bit) forms half the address pair of the STV. The maximum value of the noise signal is defined to be ± 1 (equivalent to about $\pm$ 2,5V) and the integration weights w_i and constant h_x are derived accordingly in A1.1.

The Y channel input is an algorithm, termed the SAG, which generates addresses, K_{nk}, that are equivalent to the arguments of a sinusoidal wave. K_{nk} has values between 0_8 and 17_8. A negative valued point on the sinusoidal wave is indicated by setting the most significant bit of K_{nk} to one. The K_{nk} word, excluding the sign bit, forms the other half of the STV address pair. The maximum value of the sinusoidal wave is ± 1 and the integration weights v_j and constant h_y are derived accordingly in A1.2 The SAG must generate $N^2/8$ K_{nk} addresses (see Chapter 3) in order to process a N point transform.

The STV is a 17_8 by 22_8 array and is derived in A.2. The sign bits of the LADC output and K_{nk} form the inputs of the signum function. Only the first $\frac{N}{2}$ harmonics are calculated and there are $\frac{N}{2}$ Accumulators for the real harmonic components and $\frac{N}{2}$ Accumulators for the imaginary harmonic components. The resultant harmonic is then calculated from its real and imaginary components.

The program is divided into the following subprograms

1) N samples of the signal are taken at a rate set by an external

oscillator and are stored in an N location block MB.

2) 4 or 8 similar time samples are shuffled out of MB and are placed in block SB.

3) The SAG generates the required K_{nk} values which are combined with the time samples to form the address pairs.

4) The STV is accessed and the relevant Accumulator is incremented or decremented.

5) Steps 2, 3 and 4 are repeated until all the time samples have been processed.

6) The resultant harmonics are calculated.

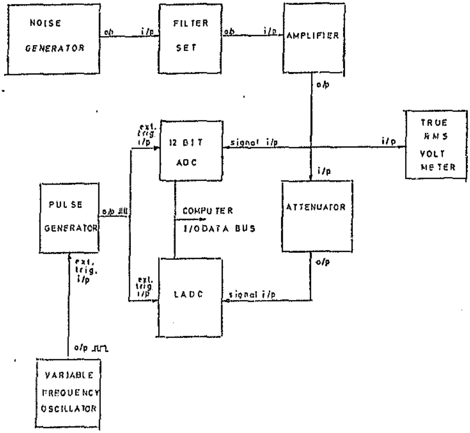


Fig.4.1 Noise Signal System (The ADC for the FFT is also shown)

4.3 The Numerical Values of the DFTTLU Harmonics

In 2.5 it was shown that the harmonics of the DFTTLU are approximately equal to the harmonics of the DFT algorithm for the same input signal.

The definition of the DFT[5] from eqn. 2.15

$$X(k\Omega) = \sum_{n=0}^{N-1} x_s(nT)e^{-j2\pi nk/N} \qquad 4.1$$

Consider $x_s(nT)$ is a constant maximum signal (ie zero frequency - dc level).

$$x_s(nT) = 1 \qquad n = 0,1 \ldots\ldots N-1 \qquad 4.2$$

Then

$$X(k\Omega) = N \qquad k = 0 \qquad 4.3$$

and $$X(k\Omega) = 0 \qquad k = 1,2,\ldots\ldots N-1 \qquad 4.4$$

Consider the case where $x_s(nT) = \sin\left(\frac{2\pi n}{N}\right)$ ie $k = 1$

Then $$X(k\Omega) \leq \frac{N}{2} \text{ for } k = 1 \qquad 4.5$$

$$X(k\Omega) = 0 \qquad k \neq 1 \qquad 4.6$$

$(\frac{N}{2})$ is the maximum value that $X(k\Omega)$ can have for a time varying signal of peak value ± 1. The derivation and values of $(\frac{N}{2})$ are given in 4.4. In the dissertation all harmonics will be measured with respect to $(\frac{N}{2})$ and a harmonic that has a magnitude half the maximum value is written as

$$X(k\Omega) = 0,5(\tfrac{N}{2})$$

The decibel scale is defined

$$0\text{dB} \equiv (\tfrac{N}{2})$$

and a harmonic of $0,5(\frac{N}{2})$ can be written in dB as

$$20\log\frac{0,5(\frac{N}{2})}{(\frac{N}{2})} \doteqdot -6,02\text{dB} \qquad 4.7$$

4.4 DFTTLU Scale Factors

When the algorithm derived in 3.2.2 is programmed, the constants in the STV must be represented by a 2's complement word.

From A2, the maximum constant that has to be stored in the STV is

3,407545177

The machine code that is chosen to represent this value is

$$40\ 000_8 = 16\ 384_{10} = 2^{14}$$

The remaining constants in the STV are scaled up by the value $(D_x D_y)$, (cf. eqn 2.37/2.38) where

$$D_x D_y = \frac{16\ 384}{3,407\ 545\ 177}$$

The fractional part of the machine code values of the STV are truncated.

The STV with a maximum value of 2^{14} is defined as

$S_{14}(I_n K_{nk})$

If the machine code used to represent 3,407 545 177 is

$$20\ 000_8 = 2^{13} = 8\ 192_{10}$$

the STV is denoted as:

$S_{13}(I_n K_{nk})$

ie. If each constant in S_{14} is halved (the machine code is shifted one place to the right), the resulting STV will be

$S_{13}(I_n K_{nk})$

Consider the case where the STV is $S_{14}(I_n K_{nk})$ and substituting this into equation 2.37/2.38.

$$F_k = \frac{h_x h_y}{D_x D_y} \sum_{n=0}^{N-1} \mathrm{sgn}(I_n\ K_{nk})\ S_{14}\ (I_n K_{nk}) \qquad 4.8$$

From A1 and A2

$h_x = 0,3431557502$

$h_y = 1.0$

and $(D_x D_y) = \dfrac{16384}{3,407545177}$

Substituting these values into $\dfrac{h_x h_y}{D_x D_y}$ and factorising

$$\text{ie } \frac{h_x h_y}{D_x D_y} = \frac{(0,3431557502)(1.0)(3,407545177)}{(16384)} \qquad 4.9$$

$$\frac{h_x h_y}{D_x D_y} = \frac{1}{(2048)(6,841590625)} \qquad 4.10$$

$$\text{ie } \frac{h_x h_y}{D_x D_y} \doteqdot \frac{1}{(2048)(7)} \qquad 4.11$$

and substituting into 4.8

$$F_k = \frac{1}{(2048)(7)} \sum_{n=0}^{N-1} \mathrm{sgn}(I_n\ K_{nk})\ S_{14}(I_n K_{nk}) \qquad 4.12$$

This form of the equation is easy to solve with a digital computer that uses integer arithmetic.

The computer is programmed to solve the following equation:

$$(2048)F_k = \frac{1}{7} \sum_{n=0}^{N-1} \mathrm{sgn}\ (I_n K_{nk})\ S_{14}\ (I_n K_{nk}) \qquad 4.13$$

The 2048 is considered a computing scale factor. If overflow or underflow of an Accumulator occurs, each constant in the STV is halved and the time samples are reprocessed. Equation 4.9 would then be written as

$$\frac{h_x h_y}{D_x D_y} = \frac{(0.3431557502)(1.0)(3.407545177)}{(8192)} \qquad 4.14$$

i.e. $$\frac{h_x h_y}{D_x D_y} \doteq \frac{1}{(1024)(7)} \qquad 4.15$$

The computer would then solve the following equation

$$(1024)F_k = \frac{1}{7} \sum_{n=0}^{N-1} \operatorname{sgn}(I_n K_{nk})\, S_{13}(I_n K_{nk}) \qquad 4.16$$

The above process would be repeated until no overflow or underflow occurred in the Accumulators.

The general equation is

$$\left(\begin{array}{c}\text{overall}\\ \text{scale factor}\end{array}\right)F_k = \frac{1}{7} \sum_{n=0}^{N-1} \operatorname{sgn}(I_n K_{nk})\, S(I_n K_{nk}) \qquad 4.17$$

The derivation of $\left(\frac{N}{2}\right)$ defined in 4.3, can now be discussed in the light of equation 4.17.

Consider the case of a N = 128 point DFTTLU with an STV S_{11} (ie. the overall scale factor is 256).

$$(256)F_k = \frac{1}{7} \sum_{n=0}^{127} \operatorname{sgn}(I_n K_{nk})\, S_{11}(I_n K_{nk}) \qquad 4.18$$

From equation 2.36, equation 4.1 can be written as:

$$(256)X_k = 256 \sum_{n=0}^{127} x_s(nT)e^{-j2\pi nK/128} \qquad 4.19$$

Consider the case where a signal

$$x_s(nT) \quad = \quad \sin \frac{2\pi n}{128} \qquad k = 1 \qquad \text{(as in section 4.3)}$$

is processed, both equations 4.18 and 4.19 should give a result less than or equal to 16384. (This is not the case in practice, as equation 4.18 is only valid for noise like signals). 16384 is equivalent to $(\frac{N}{2})$, defined in section 4.3 , for the above case.

In Table 4.1, the overall scale factors, the values of $(\frac{N}{2})$, and the corresponding STVs are tabulated for N = 128, 256, 512, 1024. The maximum 16 bit 2 s complement number that has been chosen to represent $\frac{N}{2}$ is 16384. (ie. 40 000_8). Thus the STV, S_{14}, (Table A2.3), which has a resolution of 15 bits, will always be used with a resolution of 12 bits or less. This accounts for the blank spaces in Table 4.1.

Table 4.1 $(\frac{N}{2})$ Values

Overall Scale Factor	STV	$(\frac{N}{2})$ Values			
		N = 128	N = 256	N = 512	N = 1024
2048	S_{14}				
1024	S_{13}				
512	S_{12}				
256	S_{11}	16 384			
128	S_{10}	8 192	16 384		
64	S_9	4 096	8 192	16 384	
32	S_8	2 048	4 096	8 192	16 384
16	S_7	1 024	2 048	4 096	8 192

4.5 Discussion

The Assembler programme in A4 computes equation 4.17 and automatically adjusts the STV until overflow or underflow of the Accumulators (which contain the partial sum of the harmonics) is eliminated. The programme then lists the number of times the STV constants have been halved and which of the Accumulators had overflowed or underflowed. Once the DFTTLU of a signal burst has been computed, the overall scale factor is obtained from Table 4.1, given N and the number of times the STV was adjusted. The layout of the Accumulators in the core is given in A 3.8.7 and subprogrammes are written to analyse or display the harmonic values in the Accumulators. The overall scale factor has been made a positive integer power of 2 and the machine coded harmonic values can be easily scaled to their proper values by shifting them the required number of times to the right. The DFTTLU harmonics are also in a suitable form to be compared with the FFT harmonics. This comparison is discussed in Chapter 5.

5 ERROR ANALYSIS

5.1 Introduction

The error characteristics of the DFTTLU can be investigated using the analysis in Chapter 4 of reference 6. This analysis involves the investigation of time and amplitude sampling errors as well as computational errors such as truncation errors and the errors introduced by the abscissa integration formula. The joint probability density function of the noise signal and the N sine and cosine signals must also be calculated to implement this rigorous error analysis. The results of this analysis would show how the DFTTLU of a sampled signal differed from the Fourier Transforms of the continuous signal. However numerical methods and digital systems are used in most modern practical signal processing systems and it would be more useful to know how the results of the DFTTLU differ from the results of another digital Fourier Analysis system of known accuracy given the same sampled time signal. These results would be of practical use to a signal processing system designer faced with the task of choosing a Fourier Analysis system.

The Fast Fourier Transform (FFT) algorithm[5,16] is used as the Fourier Analysis system of known accuracy.[18] The Assembler programme uses integer arithmetic and can process a maximum of 1 024 complex points (see appendix A5). The FFT harmonics are regarded as the true values and the differences between the FFT and DFTTLU harmonics are regarded as the errors which are analysed in two ways. One method is to plot each DFTTLU harmonic and its associated error. The other method is to classify the errors according to the magnitude of the associated DFTTLU harmonic and then to calculate an average value of each group of errors. Clearly the machine code values of the DFTTLU and FFT harmonics must be referred to the same value of $(\frac{N}{2})$ before errors can be calculated. The sampling subprogramme (A4.3) must control the LADC and the ADC (used in conjunction with the FFT) and these peripherals must sample the signal simultaneously. After the two sets of harmonics have been computed the results must either be plotted or averaged. These Assemblerprogrammes are discussed in appendices A4, A5 and A6 while the error plots and error averages are in 5.2 and 5.3 respectively.

5.2 Plotting the Harmonics

The programmes required are:

1) The Fast Fourier Transform, G_k, $k = 0,1 \ldots (N-1)$ to supply the true harmonics.

2) The DFTTLU. The harmonics are denoted F_k, $k = 0,1 \ldots (N-1)$

3) A programme to scale the two sets of harmonics, G_k and F_k, so that comparisons can be made.

4) A programme to calculate $G_k - F_k$ and to plot F_k and $G_k - F_k$.

These results are passed to a 10 bit digital to analog converter which is connected to an XY plotter. The $G_k - F_k$ and the F_k values are plotted on the Y axis while the X axis is the frequency index (k) axis. Band pass, low pass and band stop filtered noise is analysed for N = 128, 256, 512 and 1 024 and fig 5.1 shows the format of the plot. Fig 5. 2 shows a plot of band pass filtered noise and the errors associated with each harmonic.

5.2.1 Scaling the Harmonics, F_k and the Errors, $G_k - F_k$, for the 10 bit Digital to Analog Converter (DAC)

The 10 bit DAC can only process numbers that are in the range -512_{10} to 511_{10} ($-1\ 000_8$ to 777_8). The format of the A4 page (pages) is given in Fig.5.1. *In the case where N=1 024 and* 512 harmonics are plotted, 128 harmonics are plotted to a page for clarity and 4 pages are needed. F_k is plotted in the top half of the page between Y = 0 and Y = 511. The errors are plotted in the Y di bout Y = −256. Errors greater than zero lie between Y nd Y = 0. Errors less than zero lie between Y = −256 and Y = −512.

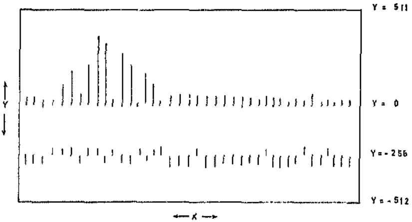


Fig.5.1 Plotter Format

The program calculates the following equation

$$(\text{scale factor})F_k = (\tfrac{1}{2})(\tfrac{1}{7}) \sum_{n=0}^{N-1} \mathrm{sgn}(I_n K_{nk}) S(I_n K_{nk}) \qquad 5.1$$

The extra factor of a half is included so that the harmonics will not exceed 511, the positive vertical range limit of the plotter. The 'scale factors' in Table 5.1 are half the corresponding 'overall scale factors' in Table 4.1. The fraction of $(\frac{N}{2})$ that corresponds to Y = 512 is given in Table 5.1 in place of the value of $(\frac{N}{2})$ in Table 4.1.

These fractions of $(\frac{N}{2})$ are calculated as follows:
Consider a case where $(\frac{N}{2})$ = 16 384 in Table 4.1. The corresponding value would be 8192 in Table 5.1 (ie. 8192 = $(\frac{N}{2})$ for equation 5.1). 512 is one sixteeth ($^1/_{16}$) of 8192. Thus Y = 511 (Y = 512 can not be plotted) is marked as $^1/_{16}(\frac{N}{2})$

The 'scale factors', STVs and fractions of $(\frac{N}{2})$ that correspond to Y = 512 are tabulated in Table 5.1 for N = 128, 256, 512, 1024.

5.2.2 Discussion

The harmonic plots are not intended to give numical values of errors, but rather to show trends in errors. In fig 5.3 an amplitude distribution of errors associated with the band stop and band pass regions is given. Two trends, valid for band pass, low pass and band stop filtered noise, emerge.

1) The DFTTLU estimate of low magnitude harmonics is greater than that of the FFT.

2) The difference in the estimates of the higher magnitude

harmonics follow a random pattern and have lower values than the lower magnitude harmonic differences.

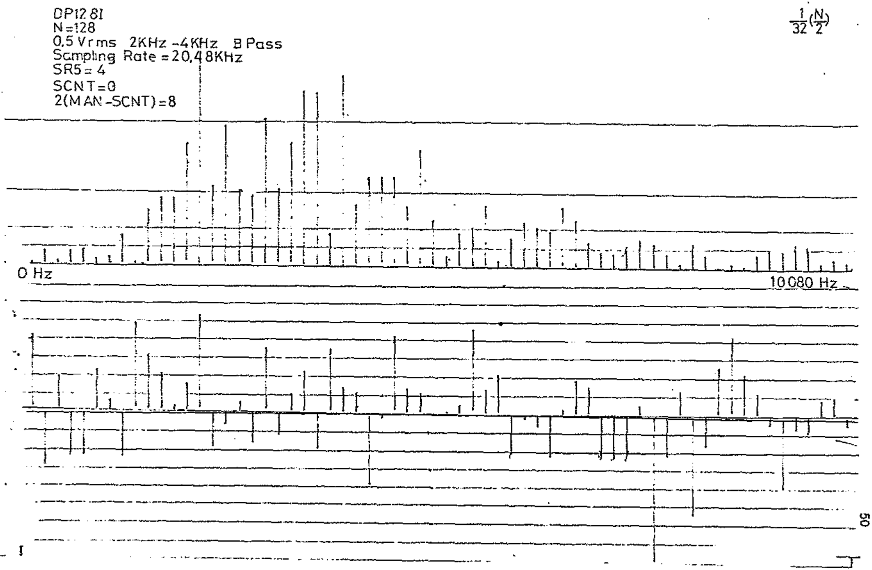
These results are consistent with those obtained in the error analysis of Fourier Series Coefficient in Chapter 4.8 of reference 6. The results of the harmonic plots and the simple amplitude distributions suggest further investigation into the variation of errors with the harmonic amplitude and the calculation of a mean value of the random errors.

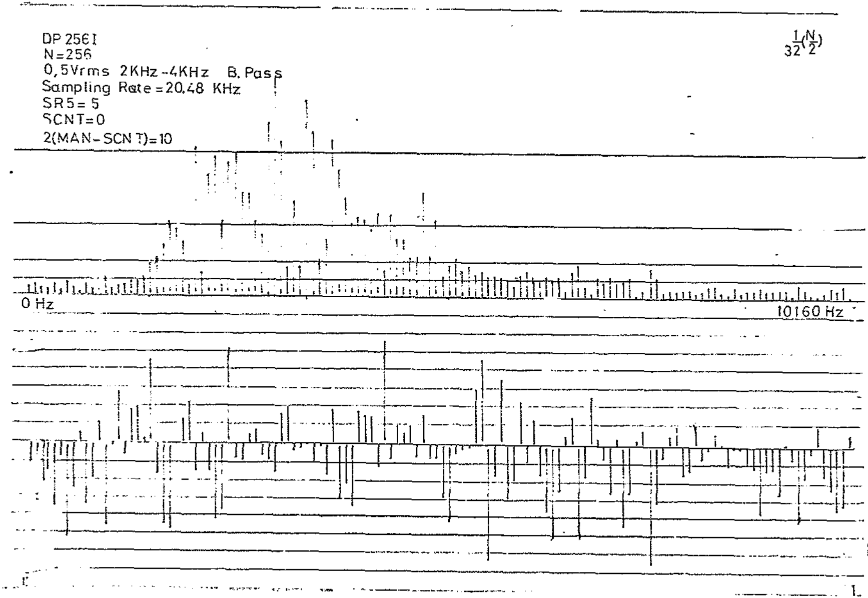
Table 5.1 Scales for Harmonic Plots.

Scale Factor	STV	Value of Harmonic at Y = 512			
		N = 128	N = 256	N = 512	N = 1024
1024	S_{14}				
512	S_{13}				
256	S_{12}				
128	S_{11}	$\frac{1}{16}(\frac{N}{2})$			
64	S_{10}	$\frac{1}{32}(\frac{N}{2})$	$\frac{1}{16}(\frac{N}{2})$		
32	S_9	$\frac{1}{64}(\frac{N}{2})$	$\frac{1}{32}(\frac{N}{2})$	$\frac{1}{16}(\frac{N}{2})$	
16	S_8	$\frac{1}{128}(\frac{N}{2})$	$\frac{1}{64}(\frac{N}{2})$	$\frac{1}{32}(\frac{N}{2})$	$\frac{1}{16}(\frac{N}{2})$
8	S_7	$\frac{1}{256}(\frac{N}{2})$	$\frac{1}{128}(\frac{N}{2})$	$\frac{1}{64}(\frac{N}{2})$	$\frac{1}{32}(\frac{N}{2})$

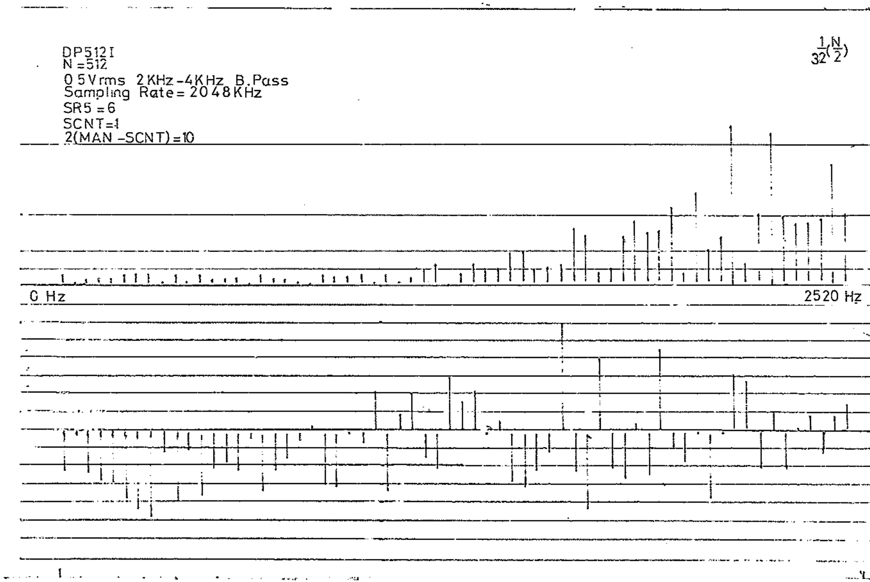
Fig. 5.2 HARMONIC PLOTS

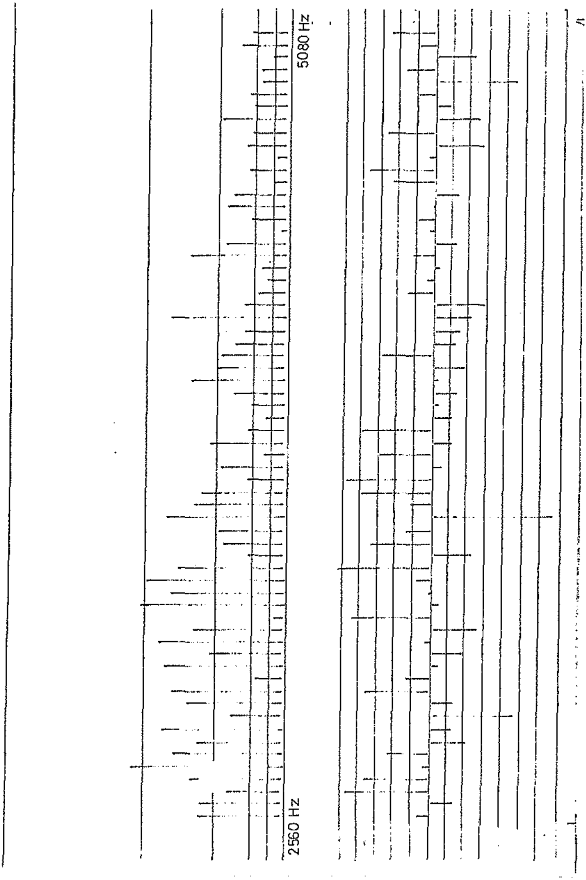
N	Λ(Hz)
128	160
256	80
512	40
1 024	20

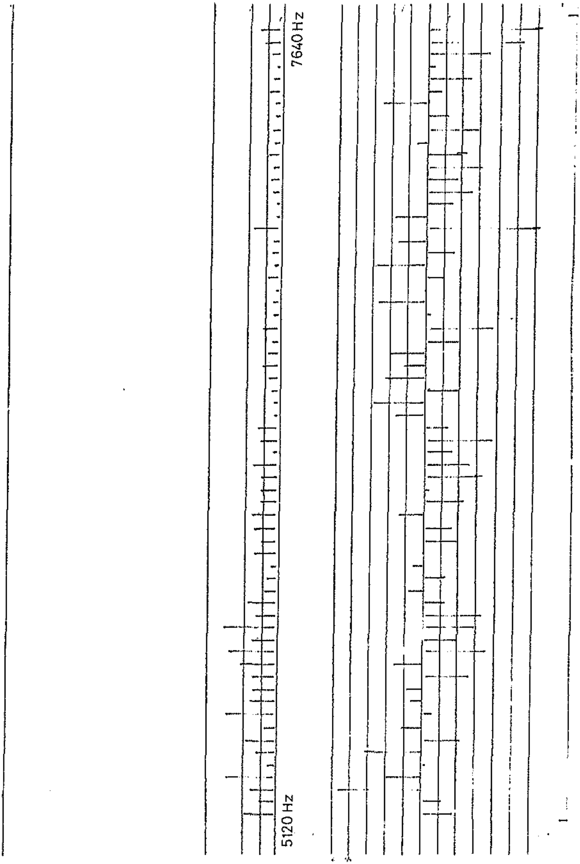

DP12 81
N=128
0,5 Vrms 2KHz -4KHz B Pass
Sampling Rate = 20,48KHz
SR5 = 4
SCNT=0
2(MAN-SCNT)=8
$\frac{1}{32}(\frac{N}{2})$
0 Hz
10080 Hz

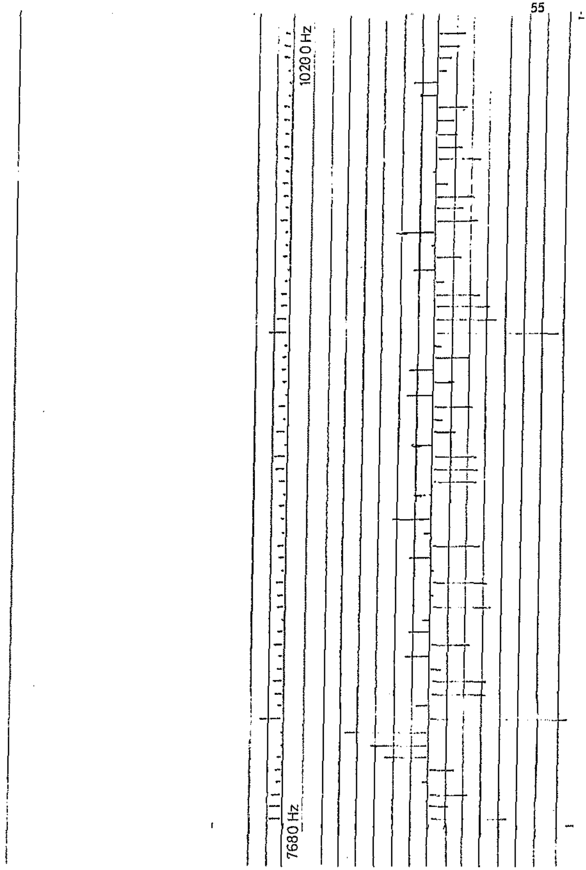


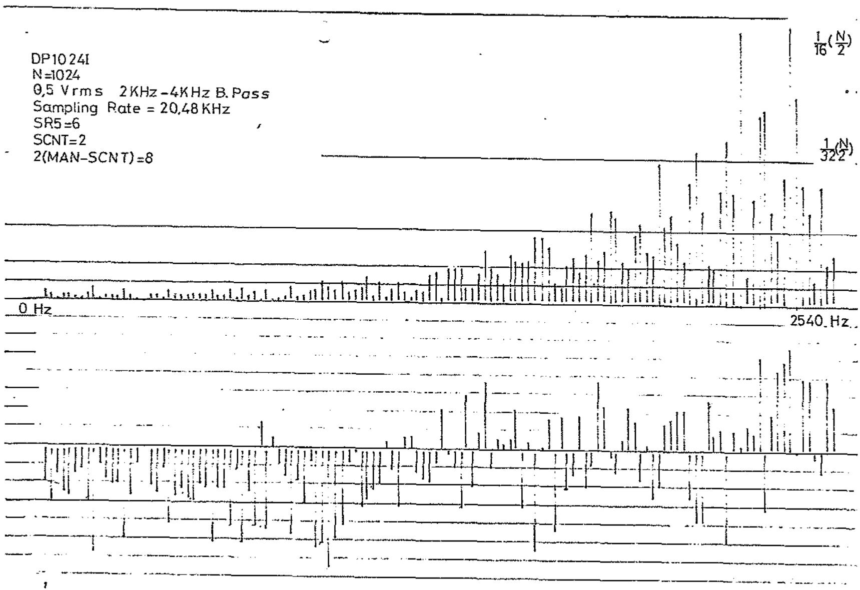
DP 256I
N=256
0,5Vrms 2KHz-4KHz B. Pass
Sampling Rate=20.48 KHz
SR5= 5
SCNT=0
2(MAN-SCN T)=10
$\frac{1}{32}(\frac{N}{2})$
0 Hz
10160 Hz
1

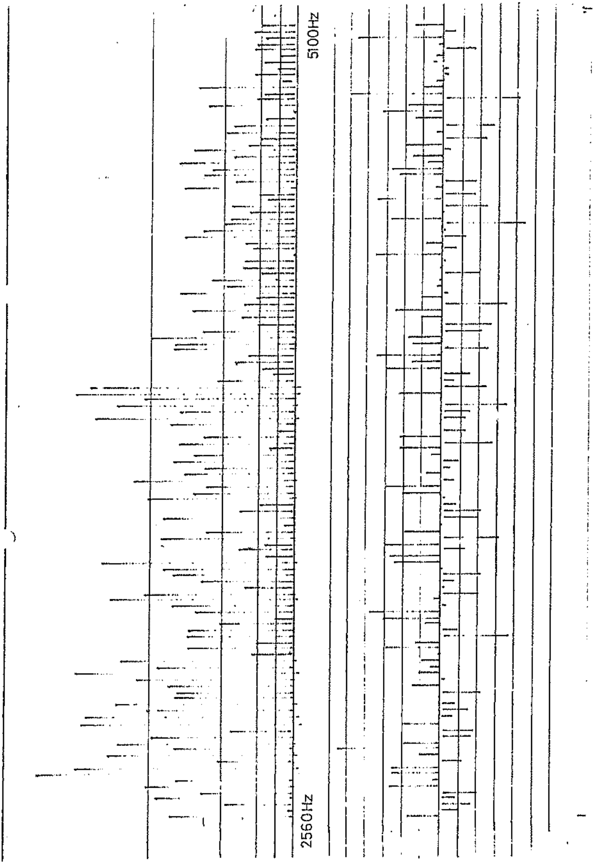

DP512I
N =512
0 5Vrms 2KHz-4KHz B.Pass
Sampling Rate= 20 48KHz
SR5 =6
SCNT=1
2(MAN -SCNT)=10
$\frac{1}{32}(\frac{N}{2})$
0 Hz
2520 Hz

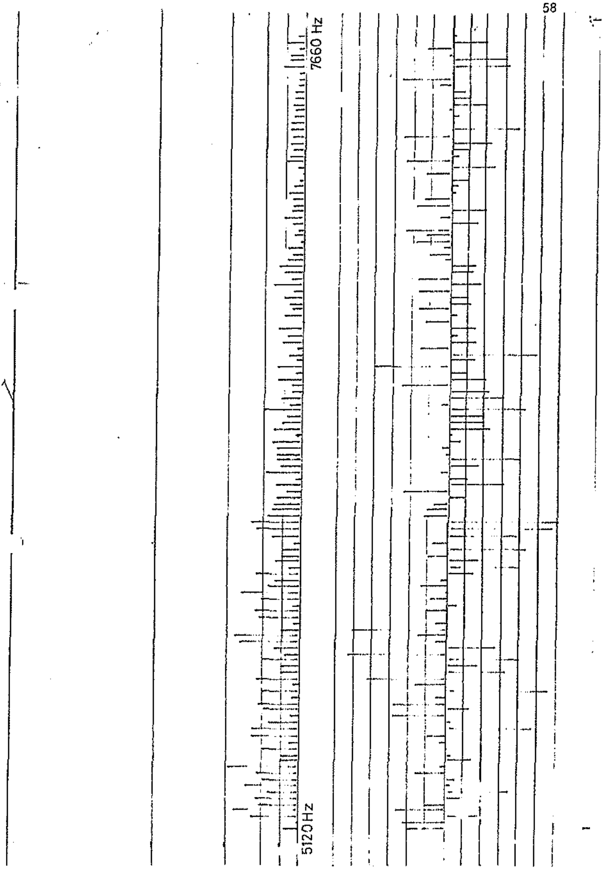

5080 Hz
2560 Hz

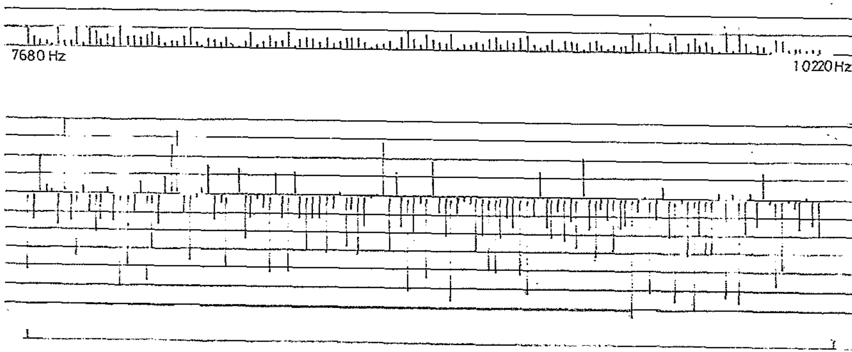

7640 Hz
5120 Hz

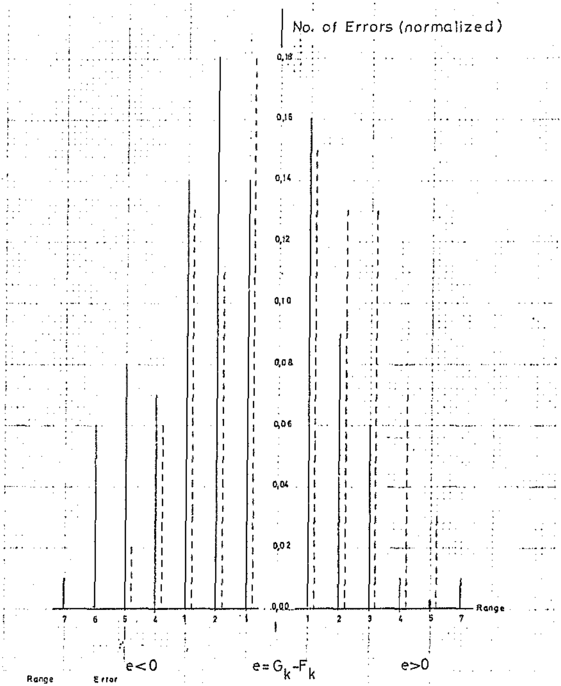

10200 Hz
7680 Hz


DP1024I
N=1024
0,5 Vrms 2KHz-4KHz B.Pass
Sampling Rate = 20,48 KHz
SR5=6
SCNT=2
2(MAN-SCNT)=8
$\frac{1}{16}(\frac{N}{2})$
$\frac{1}{32}(\frac{N}{2})$
0 Hz
2540 Hz


5100Hz
2560Hz


7660 Hz
5120 Hz


7680 Hz
10220 Hz



Range	Error
1	$0(N/2) \leq e < (1/2048)(N/2)$
2	$(1/2048)(N/2) \leq e < (2/2048)(N/2)$
3	$(2/2048)(N/2) \leq e < (3/2048)(N/2)$
4	$(3/2048)(N/2) \leq e < (4/2048)(N/2)$
5	$(4/2048)(N/2) \leq e < (5/2048)(N/2)$
6	$(5/2048)(N/2) \leq e < (6/2048)(N/2)$
7	$(6/2048)(N/2) \leq e < (7/2048)(N/2)$

Fig. 5.3 Simple Error Amplitude Distribution of DP 1024I in fig 5.2

5.3 A Numerical Error Analysis

The plots of the harmonics and their errors in Fig.5.2 show that the errors have random magnitudes and signs. The section deals with a method of calculating a mean value of the errors and investigating the variation of the errors with the magnitudes of the harmonics. In this dissertation, the term 'spurious response' is used to describe harmonics that have a zero or very small value. In Fig.5.2, the errors associated with the spurious response tend to have larger magnitudes and have the same sign. The programmes needed for this analysis are:

1) The FFT programme.

2) The DFTTLU programme.

3) The scaling programme.

4) A Programme that quantises F_k and classifies the errors accordingly. It then calculates a mean value of each group of errors.

The result is a graph of mean error plotted against harmonic magnitude.

5.3.1 Quantising the Harmonics

The harmonic are coarsely quantised so that the number of harmonics that fall into any quantising level is reasonably high and statistically meaningful averages can be calculated. The maximum value of any harmonic is $(\frac{N}{2})$ and this is divided into approximately 6dB quantising levels.

eg $(\frac{N}{2})$, $\frac{1}{2}(\frac{N}{2})$, $\frac{1}{4}(\frac{N}{2})$ .....

The maximum machine code value that can represent $(\frac{N}{2})$ is 16 384 $= 2^{14}$. The quantising programme calculates the ranges according to the following formula:

$$2^{p-1} \leqslant H_p < 2^p \qquad p = \quad 1, 2 \,\ldots\ldots\, 12, 13, 14. \qquad 5.2$$

Hp are those harmonics that have a value in the range defined above. The number of harmonics in the limits is denoted Np. There are 15 quantising levels, this being set by choosing 2^{14} to be the maximum value of $(\frac{N}{2})$. If overflow or underflow of the Accumulators occurs (A3.10.5) and the STV is halved, $(\frac{N}{2})$ is also halved and the quantising level (levels) above the new value of $(\frac{N}{2})$ will be empty. Fig.5.4 shows the spacing of the quantising level on a linear scale.

5.3.2 Definition of Error Terms

As in the previous case (5.2) the error in a harmonic is defined as:

$$\text{error} = G_k - F_k \qquad 5.3$$

In this case the magnitude of the error is used.

$$e = \left| G_k - F_k \right| \qquad 5.4$$

The errors, e, that are associated with harmonics in the range Hp are denoted e_p. The mean error, Ep is defined as

$$Ep = \frac{1}{Np} \sum_{Np} e_p \qquad 5.5$$

ie Ep is the average of the magnitudes of the errors associated

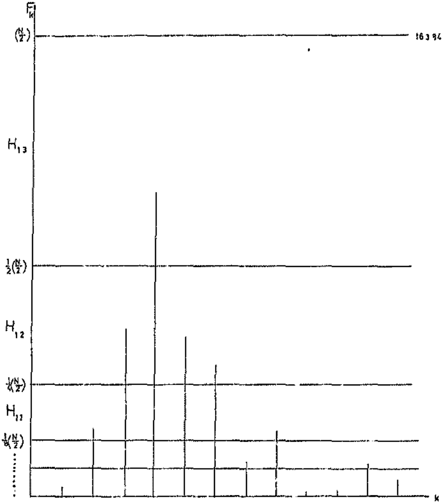


Fig.5.4 Amplitude quantisation of the Harmonics

with harmonic magnitudes in the range Hp. The standard deviation of e_p is denoted σ_p and is defined as

$$\sigma_p = \sqrt{\frac{\sum_N (e_p)^2 - (\sum_N e_p)^2/Np.}{N-1}} \qquad 5.6$$

The percentage mean . error is

$$E_p/(N/2) \times 100 \quad \% \qquad 5.7$$

The percentage standard deviation is

$$\sigma_p/(N/2) \times 100 \quad \% \qquad 5.8$$

The Assembler programme that computes e_p, $(e_p)^2$ and Np is given in A4.12.7.

5.3.3 Error Tests

The system that implements the error tests is given in A6. The input signal consists of filtered noise which amplified to have an rms value of about 2V at the ADC and is attenuated to about 0,5V rms at the LADC input.

Test One

The constants in the STV are adjusted so that the machine code of 40 000_8 is equivalent to $(\frac{N}{2})$. Filtered noise is processed for N=128,256,512 and 1 024. Errors from about 30 transforms are averaged to get repeatable results for Ep and σp.

Test Two

This test i similar to test one but the machine code equivalent to $(\frac{N}{2})$ is set to 20 000_8 ie One less bit is used to represent the constants in the STV.

Test Three

This test is similar to test one but the machine code equivalent to $(\frac{N}{2})$ is set to 10 000_8.

Test Four

The attenuator (see A6) is set to 14,7 dB, 17,7dB and 8,7dB. In each case tests similar to test one are performed and Ep is measured. It was not possible to measure σ_p as there was not enough memory core to hold the results of $(e_p)^2$ (see A4.12.7). The STV is adjusted to maximum possible resolution ie. As many bits as possible are used to represent the constants in the STV without Accumulators overflowing or underflowing.

5.3.4 Discussion of Test Procedure

Only band pass and low pass filtered noise is used in the above tests. In this dissertation wide band low pass filtered noise is noise that is cut off at 5kHz while narrow band low pass filtered noise is cut off at 2kHz. Wide band pass filtered noise has a pass band from 2kHz to 5kHz. Narrow band pass filtered noise has a pass band from 2,5kHz to 3,5kHz. In all cases the roll off is 24dB/octave.

In test one the maximum value of $(\frac{N}{2})$ (ie. the maximum harmonic magnitude that could occur) is $40\ 000_8$. STVs S_{11}, S_{10}, S_9, S_8 are used for N=128, 256, 512, 1024 respectively. The effects of using one less bit to represent the constants in the STV are investigated in tests two and three. Test four investigates superficially the effect of overloading or underloading the LADC .

5.3.5 Discussion of Error Test Result

The processing of noiselike signals results in harmonics that have large variations in magnitude and errors from at least thirty transforms must be accumulated before repeatable values of E_p and σ_p can be calculated. The results of the error tests are in the form of a graph of E_p and σ_p plotted as a function of H_p. In most of the results Ep and its corresponding value of σ_p differed by less than 0,01 per cent of $(\frac{N}{2})$ in magnitude and both are represented by the same line on the graph.

If wide band low pass or wide band pass filtered noise is processed, there is an approximately even distribution of harmonic magnitudes in the Hp ranges and an error characteristic similar to Fig.5.5 is obtained. This result confirms those in section 4.8 or 5.4 of reference 6. Reducing the resolution of the STV by 2 bits has no effect on the results in Fig.5.5.

If narrow band low pass or narrow band pass filtered noise is processed, there is an uneven distribution of harmonc magnitudes, the number of high magnitudes being much less than the number of low magnitudes. If the results of 30 - 60 transforms are accumulated, the number of harmonics in the higher Hp ranges is of the order of 10 - 20 while the number of harmonics in the lower Hp ranges is of the order of 1000. The values of Ep and σp associated with the higher Hp ranges will not be as reliable as those in the lower Hp ranges. The error characteristic is given in Fig.5.6 and the less reliable results are indicated by a dotted line. Fig.5.7 and Fig.5.8 show the variation of Ep and σp as the number of bits representing a constant in the STV is reduced.

Tests one, two and three show that:

1) If the noise signal being processed has a narrow bandwidth, the STV constants must have maximum resolution for maximum accuracy.

2) The higher magnitude harmonics of a narrow band noise signal can have errors of the order of 2% $(\frac{N}{2})$

3) The minimum error is of the order of 0,5% $(\frac{N}{2})$

4) The accuracy of the lower magnitude harmonics is independent of type of pass band of the noise signal and the resolution of the STV.

Test four shows that overloading or underloading the LADC can cause errors of the order of 18% $(\frac{N}{2})$ in the high magnitude harmonics. This error is independent of the type of pass band of the noise signal (Fig.5.9).

This error analysis is a simple and practical method of obtaining an error characteristic of the DFTTLU using a minicomputer with 16K core and an integer arithmetic facility. The analysis has only been superficially implemented as, for example, the following has not been investigated.

1) 3 - 6 bits to represent an STV constant.

2) more or less than 16 'abscissas' in the Y channel.

3) less than 18 'abscissas' in the X channel.

The investigation has shown that the DFTTLU is a simple and practical algorithm to take Fourier Transforms of noise like signals.

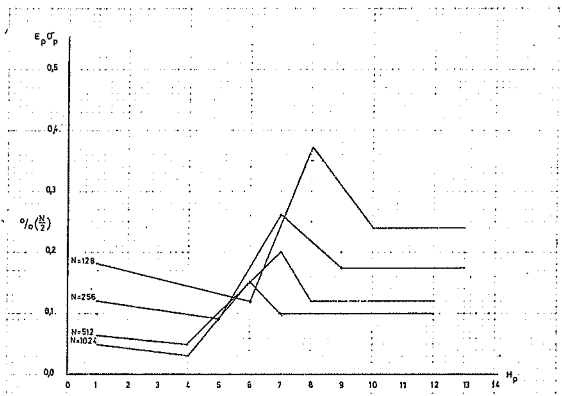


Fig. 5.5 Error Characteristics $\left(\frac{N}{2}\right) = 40000_8, 20000_8, 10000_8$ Wide Band Noise

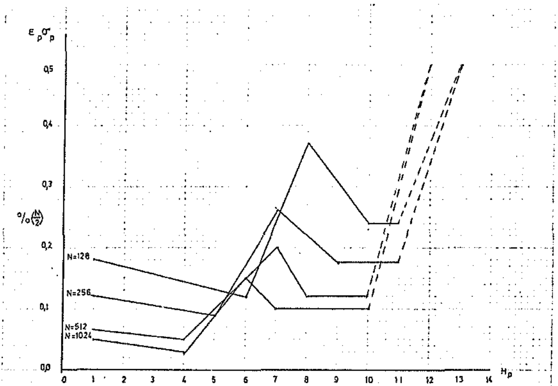


Fig 5.6 Error Characteristics $(\frac{N}{2})=40000_8$ Narrow Band Noise

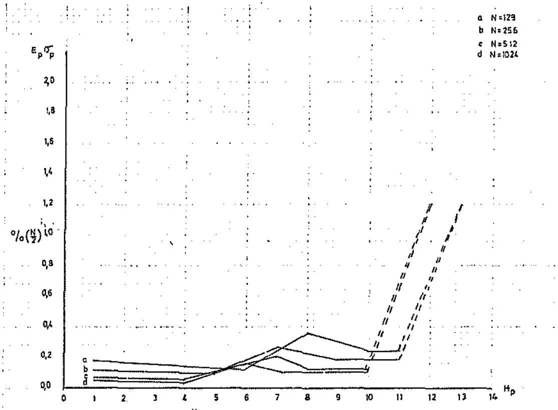


Fig.5,7 Error Characteristics $(\frac{N}{2})=20\,000_8$ Narrow Band Noise

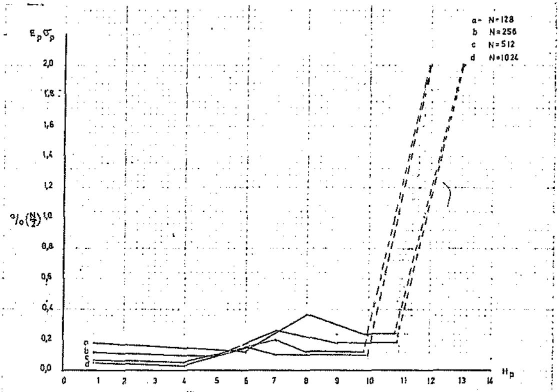


Fig. 5,8 Error Characteristics $\left(\frac{N}{2}\right) = 10\,000_8$ Narrow Band Noise

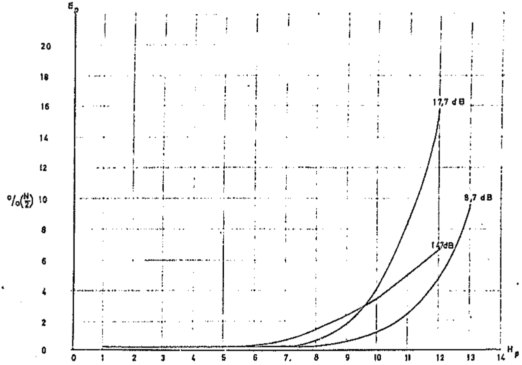


Fig.5.9 Error Characteristics - Varying Attenuator - N any of above values

6 CONCLUSIONS

6.1 General Discussion

The Discrete Fourier Transform of a sampled noise signal can be computed using the table look up algorithm. The differences between the resulting harmonics and the harmonics obtained using the Fast Fourier Transform are less than 0,5% of the maximum harmonic magnitude for a noiselike signal which has been filtered so that it has a pass band of the order of 3000 Hz.

The ultimate aim of the work is the development of a real time Fourier Transform system that processes wide dynamic range noiselike signals - the present system meets all the aims except that of real time operation because the number of calculation steps is proportional to N^2 and the Assembler code needed to generate the K_{nk} addresses is inherently cumbersome. Investigation has shown that the decimation[5] in time or frequency used by the Fast Fourier Transform to reduce the number of calculation steps to N log N cannot be applied to the table look up method.

6.2 Real Time Operation

Real time operation can be achieved if a set of harmonics can be processed in a time less than the signal burst time. This means that a column of address pairs (real and imaginary) must be processed in less than the sampling period. The most time consuming operation is the generation of the K_{nk} addresses using the AssemblerCode. This operation lends itself to hardware implementation as the K_{nk} generator is simply an UP/DOWN counter that uses sign/magnitude notation. A basic step in the table look up algorithm (described in Chapter 3.3) is

1) Generate a K_{nk} address

2) Combine the logarithmic analog to digital converter output, I_n, and K_{nk} to form an address pair.

3) Extract the required constant from the Stored Table of Values.

4) Adding (or subtracting) to constant to (or from) to correct Accumulator

$\frac{N}{2}$ of the basic steps are required to process a column and the time needed is less than

$$2.\frac{(\text{Sampling period})}{N}$$

Duplicate systems are needed as the real and imaginary harmonic components must be computed simultaneously. The formulae derived in A3.5.3 can be used to simplify the design of the K_{nk} generator ie. When processing a column of address pairs it is simpler to process all the address pairs that contribute to the add harmonics separately, then the even numbered harmonics that are multiples of four and finally the remaining even numbered harmonics. The K_{nk} generator is then limited to specified values in each case. In addition only $\frac{N}{4}$ sets of columns of $\frac{N}{4}$ rows of K_{nk} need be generated.

6.3 Recommendations for Future Work

The logical extension of this work is the design and construction of a fast, simple and relatively accurate Fourier Transform instrument which has the following advantages when compared with a system based on the Fast Fourier Transform.

1) The basic calculation step does not require multiplication. The only multiplication needed is in the calculation of the resultant harmonics.

2) The logarithmic analog to digital connecter is less complicated than the linear one required by the Fast Fourier Transform.

3) The wordlength depends on the number of levels in the logarithmic analog to digital converter which is of the order of 16 - 20. The Fast Fourier Transform needs a linear analog to digital converter with about 1 000 - 2 000 levels.

4) The table look up Fourier Transform algorithm is less complex than a Fast Fourier Transform of comparable accuracy.

LIST OF REFERENCES

1. Ball Computer Products DDOS. 1.6 Programmer's Reference Manual. 86-0140050 - 4, 1974.

2. Bell, Potter, Kopp, Green, Visible Speech; New York: Van Nostrand, 1947.

3. Bruel and Kjaer, Master Catalog; Copenhagen: 1974.

4. Data General Corp, How to Use the Nova Computers; D.G. NM-5, April 1971.

5. Gold, B. and Rader, C.M. Digital Processing of Signals; New York: McGraw-Hill, 1969.

6. Hanrahan, H.E. "The Principles and Error Characteristics of a Technique for Determining the Time Averages of Amplitude Quantised, Time Sampled Sequences". PhD Thesis, University of the Witwatersrand, Johannesburg: 1969.

7. _______ "Real Time Fourier Anrlysis of Amplitude Quantised Time Sampled by Computer or Special Purpose Instrument"; Trans SAIEE, 62: 141-146, 1971.

8. _______ "Time Averaging Measurement of Data that has been Logarithmically Quantised"; IEE Conf. Publ., 103 : 120, 1973.

9. _______ "A Digital Mean Square Measuring Technique Applicable to Equivalent Sound Level and Other Measurements"; 8th Int. Cong. on Acoustics, p.690, London, 1974.

10. _______ "Digital Measurement of Mean Square Pressure and other Time Averages using Logarithmic Analog to Digital Converters." Report D2 - 76 Department of Electrical Engineering, University of the Witwatersrand, Johannesburg, 1976.

11. Hanrahan, H.E. and Veradi, F. "Interface and Control System for an Analogic MP6912 Analog to Digital Converter and a Nova 1200 Digital Computer." Laboratory Handbook, Department of Electrical Engineering, University of the Witwatersrand, Johannesburg, 1974.

12. Hewlett-Packard. Electronic Instrumentation and Systems Catalog, 1975.

13. Lee, Y.W. Statistical Theory of Communication; New York: Wiley, 1960.

14. Steiglitz, K. Introduction to Discrete Systems; New York: Wiley, 1974.

15. Texas Instruments: The TTL Data Book for Design Engineers, Dallas, 1973.

16. Van Olst, R. "Nova Fast Fourier Transform Calculation." Department of Electrical Engineering, University of the Witwatersrand, Johannesburg, April, 1972.

17. Walker, A.J. "Studies on Fast Pseudo-Random Number Generators with Arbitrary Frequency Distributions." Internal Research Report, Department of Electrical Engineering, University of the Witwatersrand, Johannesburg, February, 1975.

18. Welch, P.D. "A Fixed Point FFT Error Analysis." IEEE Trans Audio and Electroacoustics AU17 - 2: 151-157 June 1969.

A1 Integration Formulae and Integration Weights

1.1 Input Signal Channel Integration Formula

The Logarithmic Analog to Digital Converter LADC quantises the input signal into 3dB levels. The quantising is illustrated in Fig.A1.1.

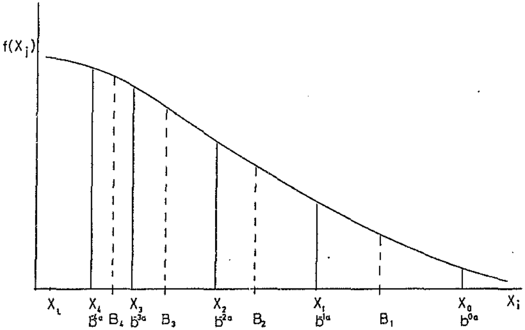


Fig. A1.1 LADC Thresholds

The tangent rule for integration is used.[5]

X_i are the thresholds of the LADC

b is the base of the logarithm

a is the index of the logarithm.

The abscissa axis must be divided into integration panels, B_i, to implement the integration.

These panels are defined

$$X_i = \frac{B_i + B_{i+1}}{2} \qquad \text{A1.1}$$

The quantiser threshold is placed in the centre of the panel.

$$X_i = X_o b^{-ia} \qquad \text{A1.2}$$

$$B_i = B_o b^{-ic} \qquad \text{A1.3}$$

$$X_o = \frac{B_o + B_1}{2} = 1 \qquad \text{A1.4}$$

(The maximum threshold is defined to be 1).

$$\frac{B_o + B_1}{2} \approx 1 \qquad \text{A1.5}$$

$$\frac{B_o + B_o b^{-a}}{2} = 1 \qquad \text{A1.6}$$

$$B_o = \frac{2}{1 + b^{-a}} \qquad \text{A1.7}$$

$$B_i = \frac{2b^{-ia}}{1 + b^{-a}} \qquad \text{A1.8}$$

The area, a_i of a panel i

$$a_i = (B_i - B_{i+1})\, f(X_i) \qquad \text{A1.9}$$

The area under the curve from X_L, the lowest threshold to X_o is

$$\int_{X_L}^{X_o} f(x)dx \doteq \sum_{i=0}^{M_x - 1} a_i \qquad \text{A1.10}$$

M_x is the number of thresholds.

$$\sum_{i=o}^{M_x - 1} a_i = \sum_{i=0}^{M_x - 1} (B_i - B_{i+1})\, f\,(X_i) \qquad \text{A1.11}$$

$$= \sum_{i=0}^{M_x - 1} h_x w_i \; f(Xi) \qquad \text{A1.12}$$

$$h_x = \frac{2(1 - b^{-a})}{1 + b^{-a}} \qquad \text{A1.13}$$

$$w_i = b^{-ia} \qquad \text{A1.14}$$

h_x is the integration constant and w_i is the integration weight.

The area of the panel between $X = 0$ and $X = X_L$ is small and is neglected.

Thus:

$$\int_0^{X_o} (x)dx. \doteq \frac{2(1 - b^{-a})}{(1 + b^{-a})} \sum_{i=0}^{M_x - 1} b^{-ia} f(X_i) \qquad \text{A1.15}$$

The notation used for the subscript in w_i and the index in b^{ia} is not compatible with the sinusoidal subscript notation in A1.2. Normally when the subscript is set to zero it is associated with zero on the abscissa axis. In the above case, zero index is associated with X_o, the maximum abscissa value. The following index numbering modification is introduced to allow a zero numbered subscript to be associated with the weight at X_L.

$$i' = (M_x - 1) - i$$

and $\quad w_i = b^{-i'a}$

The integration weights in table A1.1 are calculated using the above formula.

Table A1.1 The Signal Channel Integration Weights

i	w_i
0	0,000000000000
1	0,002762135866
2	0,00390625
3	0,005524271729
4	0,0078125
5	0,0110485434
6	0,015625
7	0,0220970869
8	0,03125
9	0,0441941738
10	0,0625
11	0,0883883476
12	0,125
13	0,1767766954
14	0,25
15	0,3535533905
16	0,5
17	0,7071066711
18	1,0

$w_i = b^{-i'a}$

$b = 2,0$

$a = 0,5$

$i' = (M_x - 1) - i$

1.2 The Sinusoidal Channel Integration Formula

1.2.1 Introduction

The problem of generating sine and cosine waves in a digital system is usually complicated. The Table look up Method simplifies this by incorporating the sinusoidal wave values into the integration formula. Thus *only the linear in time arguments* of the sine and cosine wave need be generated. This section shows how the linear arguments are manipulated mathematically to produce a sinusoidal output.

1.2.2 The Sinusoidal Argument

From chapter 2

$$e^{-j\frac{2\pi nk}{N}} = \cos\left(\frac{2\pi nk}{N}\right) - j\sin\left(\frac{2\pi nk}{N}\right) \qquad \text{A1.16}$$

where N = number of time samples in the DFT

n = index of time sample $n = 0,1 \; \ldots\ldots \; n - 1$

k = index of harmonic $k = 0,1 \; \ldots\ldots \; N - 1$

When taking the periodicity of the sinusoidal wave into account, the maximum value of (nk) is N - 1.

When $\frac{\pi}{2}$ symmetry is also taken into account the maximum value of nk is $\frac{N}{4}$.

The term 'argument' is used for the term (nk). However, the strict definition of argument is $\left(\frac{2\pi nk}{N}\right)$. *In this work nk* and machine code values that are proportional to (nk) are termed 'arguments'.

1.2.3 'Quantising' the Sine Waves

The quantisation and integration formulae are discussed. It must be noted that no actual sine wave quantiser exists and 'quantising' is implemented by the mathematical algorithm.

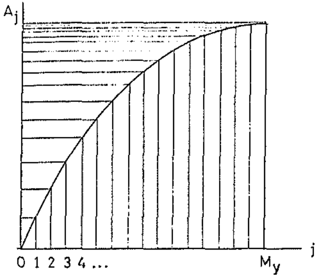


Fig.A1.2 Linearly quantised arguments

$$A_j = \sin\left(\frac{j}{M_y}\frac{\pi}{2}\right) \qquad j = 0 \ldots M_y \qquad \text{A1.17}$$

$$Y_j = \frac{A_j + A_{j+1}}{2}$$

The integration formula used is the very simple tangent rule.

The 'quantisation' levels are given by A_j.

The abscissas are given by Y_j

The area of panel j is a_j.

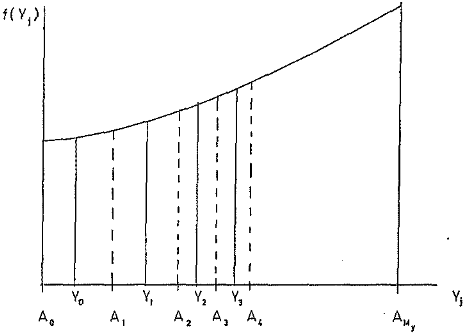


Fig. A1.3 The 'Quantisation' Levels in the Sinusoidal Channel

$$a_j = (A_{j+1} - A_j)\ f\ (Y_j) \tag{A1.18}$$

$$\int_a^1 f(y)\ dy \doteq \sum_{j=0}^{M_y - 1} a_j \tag{A1.19}$$

$$\sum_{j=0}^{M_y - 1} a_j = \sum_{j=0}^{M_y - 1} (A_{j+1} - A_j)\ f\ (Y_j) \tag{A1.20}$$

$$= \sum_{j=0}^{M_y - 1} h_y u_j\ f(Y_j) \tag{A.21}$$

$$h_y = 1 \tag{A1.22}$$

$$u_j = (A_{j+1} - A_j) = \sin(\frac{(j+1)\pi}{M_y 2}) - \sin(\frac{j\pi}{M_y 2}) \qquad \text{A1.23}$$

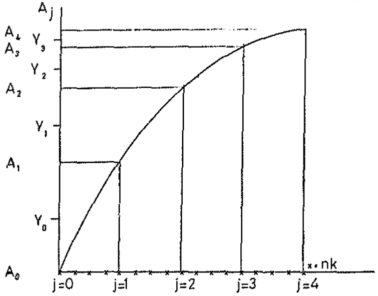


Fig.A1.4 Y_j and its corresponding arguments, nk.

$(\frac{N}{4})$, the number of arguments in the range $0 - \frac{\pi}{2}$ is greater than M_y, the number of sinusoidal 'quantising' levels. To facilitate computing the following definition is made.

$$2^q M_y = \frac{N}{4} \qquad q = 0, 1, 2, \ldots.. \qquad \text{A1.24}$$

If an argument, nk, is generated such that

$$(2^q)(j) \leqslant (nk) < (2^q)(j+1) \qquad \text{A1.25}$$

the corresponding integration weight is u_j

This is illustrated in Fig.A1.4 for $N = 64$, $M_y = 4$ and $q = 2$.

Notes on Definition

1. The Nova 1200 machine code value of the argument

equivalent to $\frac{\pi}{2}$, necessitates the following modification to equation A1.25 for the case $j = M_y - 1$

$$(2^q)(M_y - 1) \leqslant nk \leqslant (2^q)(M_y) \qquad \text{A1.26}$$

2. In chapter (2.5) the argument defining $\sin\pi$ is defined as -0. As the sine wave enters the third quadrant (cosine wave enters second quadrant), the first 'quantised' level has a nominal value of -Y. Thus the arguments that define this level must have negative signs.

3. The 'quantised' abscissas Y_j *are of no importance, but* U_j, the associated integration weight (with w_i) forms the basis of the STV.

Table A1.2 The Sinusoidal Channel Integration Weights

j	u_j
0	0,0980171401
1	0,0970731818
2	0,0951943551
3	0,0923987554
4	0,0887133045
5	0,0841734959
6	0,078823051
7	0,0727134975
8	0,065903672
9	0,0584591587
10	0,0504516523
11	0,0419582682
12	0,033060803
13	0,0238449447
14	0,0143994464
15	0,0048152733

A2 Derivation of the Stored Table of Values

The STV is derived from the integration weights in Tables A1.1 and A1.2. The 2-D array has its elements calculated according to the following equation.

$$S(s,r) = \sum_{i=0}^{s} \sum_{j=0}^{r} w_i u_j$$

where $s = 0,1,\ldots\ldots M_x - 1$ $(M_x = 19)$

and $r = 0,1,\ldots\ldots M_y - 1$ $(M_y = 16)$

$$\text{eg. } S(2,3) = \sum_{i=0}^{2} \sum_{j=0}^{3} w_i u_j = (w_o+w_1+w_2)(u_o+u_1+u_2+u_3)$$

A2.1 The Machine Code for the Stored Table Values

From Tables A2.1 and A2.2, the max. value in the STV is

$$S(18,15) = \sum_{i=0}^{18} \sum_{j=0}^{15} u_j w_i = 3,407545177$$

This value is defined to be equivalent to the machine code

$$40000_8 = 2^{14} = 16384_{10}$$

The remaining values in the STV are adjusted to this scale multiplying them by

$$(D_x D_y) = \frac{16384}{3,407545177} = 4808,1533$$

The machine code values of the STV are given in A4.15

From Al.1 $h_x = 0{,}3431458$

From Al.2 $h_y = 1{,}0.$

$$\frac{h_x h_y}{D_x D_y} \doteq \frac{1}{(2048)(7)}$$

Table A2.1 Input Signal Channel Weights

s = I	$\sum_{i=0}^{s} w_i$
0	0,0000000000
1	0,002762135866
2	0,006668385866
3	0,0121926576
4	0,0200051576
5	0,031053701
6	0,046678701
7	0,0687757879
8	0,100025788
9	0,1442199619
10	0,2067199619
11	0,2951083096
12	0,4201083096
13	0,596885005
14	0,846885005
15	1,200438396
16	1,700438396
17	2,407545177
18	3,407545177

Table A2.2 Sinusoidal Channel Weights

$r = K$	$\sum_{j=0}^{r} u_j$
0	0,0980171401
1	0,1950903219
2	0,290284677
3	0,3826834324
4	0,4713967369
5	0,5555702328
6	0,6343932838
7	0,7071067813
8	0,7730104533
9	0,8314696120
10	0,8819212643
11	0,9238795325
12	0,9569403355
13	0,9807852802
14	0,9951847266
15	1,0000000000

Table A2.3 Stored Table of Values (STV)

$S(I_n, K_{nk})$

S(row, column)

$D_x D_y$ = 4808,1533

I	k = 0	K = 1	K = 2	K = 3
0	0,0000	0,0000	0,0000	0,0000
1	1,3017259	2,5909162	3,8551549	5,0822651
2	3,1426918	6,2551196	9,3073079	12,269858
3	5,7461907	11,437045	17,017757	22,434572
4	9,4280754	18,765358	27,921923	36,809576
5	14,635026	29,129117	43,342683	57,13882
6	21,998795	43,785743	65,151015	85,888826
7	32,412744	64,513355	95,992674	126,54749
8	47,140283	93,826607	139,60933	184,04751
9.	67,96818	135,28182	201,29265	265,36485
10	97,4233257	193,90833	288,52598	380,36488
11	139,079	276,81868	411,89248	542,99939
12	197,98916	394,07169	586,35914	772,99944
13	281,3007	559,89249	833,09227	1098,2685
14	399,12101	794,3985	1182,0255	1558,2687
15	565,74409	1126,0401	1675,4918	2208,8071
16	801,38471	1595,0521	2373,3584	[illegible]28,8073
17	1134,6308	2258,3353	3360,291	4429,8841
18	1605,9121	3196,3593	4756,0243	6269,8845

I	K = 0	K = 1	K = 2	K = 3
0	0,0000	0,0000	0,0000	0,0000
1	1,3017259	2,5909162	3,8551549	5,0822651
2	3,1426918	6,2551196	9,3073079	12,269858
3	5,7461907	11,437045	17,017757	22,434572
4	9,4280754	18,765358	27,921923	36,809576
5	14,635026	29,129117	43,342683	57,13882
6	21,998795	43,785743	65,151015	85,888826
7	32,412744	64,513355	95,992674	126,54749
8	47,140283	93,826607	139,60933	184,04751
9.	67,96818	135,28182	201,29265	265,36485
10	97,4233257	193,90833	288,52598	380,36488
11	139,079	276,81868	411,89248	542,99939
12	197,98916	394,07169	586,35914	772,99944
13	281,3007	559,89249	833,09227	1098,2685
14	399,12101	794,3985	1182,0255	1558,2687
15	565,74409	1126,0401	1675,4918	2208,8071
16	801,38471	1595,0521	2373,3584	3128,8073
17	1134,6308	2258,3353	3360,291	4429,8841
18	1605,9121	3196,3593	4756,0243	6269,8845

I	K = 4	K = 5	K = 6	K = 7
0	0,000	0,0000	0,000	0,000
1	6,2604308	7,3783054	8,4251236	9,3908029
2	15,114245	17,813074	20,340355	22,671749
3	27,635333	32,569952	37,190907	41,45369
4	45,342736	53,439222	61,021065	68,015235
5	70,384686	82,952711	94,721865	105,57878
6	105,79949	124,69125	142,38218	158,70187
7	155,88361	183,71849	209,78408	233,82931
8	226,71322	267,19557	305,10471	340,07549
9	326,88148	385,25006	439,90852	490,33037
10	468,54069	552,20423	630,54978	702,82273
11	668,87697	788,31294	900,1571	1003,3321
12	952,19541	1122,2212	1281,4396	1428,3168
13	135,8682	1594,4389	1820,6545	2029,336
14	1919,505	2262,2556	2583,2196	2879,3055
15	2720,8506	3206,691	3661,6494	4081,3438
16	3854,1244	4542,3243	5186,7795	5781,2827
17	5456,8155	6431,195	7343,6393	8185,3594
18	7723,363	9102,4616	0393,899	11585,237

I	K = 8	K = 9	K = 10	K = 11
0	0,000	0,0000	0,000	0,000
1	10,266043	1,042415	11,538163	12,269674
2	24,784795	26,659151	28,276768	29,622061
3	45,317253	48,744382	51,702081	54,161853
4	74,354385	79,977454	84,830306	88,866183
5	115,41892	124,14751	131,6805	137,94532
6	173,49319	186,61366	197,93695	207,35398
7	255,62265	274,95418	291,63778	305,5127
8	371,77118	399,89647	424,15068	444,33002
9	536,0301	576,56751	611,55233	640,64747
10	768,32715	826,43209	876,57813	918,28211
11	1096,8446	1179,7937	1251,381	1310,9165
12	1561,4387	1679,5229	1781,4326	1866,1858
13	2218,474	2386,2467	2531,0387	2651,4552
14	3147,6622	3385,705	3591,1419	3761,9937
15	4461,7329	4799,1526	5090,3543	5332,5324
16	6320,1093	6798,0692	7210,5607	7553,6095
17	8948,2506	9824,9643	10208,985	10694,687
18	12665,003	13622,797	14449,398	15136,841

I	K = 12	K = 13	K = 14	K = 15
0	0,000	0,000	0,000	0,0000
1	12,708741	13,025417	13,216649	13,2806
2	30,682079	31,446613	31,908297	32,062689
3	56,100022	57,49792	58,342075	58,62437
4	92,046237	94,339841	93,74892	96,188068
5	142,88166	146,44198	148,59197	149,31095
6	214,77409	220,12582	223,3576	224,43834
7	316,4454	324,33057	329,09223	330,68458
8	460,23026	471,69826	478,6235	480,93938
9	663,57289	680,10777	690,09277	693,43185
10	951,14261	974,84314	989,1553	993,94145
11	1357,8274	1391,6616	1412,0933	1418,9259
12	1932,9668	1981,1324	2010,2184	2019,9451
13	2746,3368	2814,7699	2856,095	2869,9145
14	3896,6157	3993,7114	4052,3451	4071,9529
15	5523,3558	5660,9865	5744,0983	5771,8918
16	7823,9136	8018,8695	8136,5985	8175,9685
17	11077,393	11353,419	11520,104	11575,846
18	15678,509	16069,185	16305,105	16384,000

A.3 The Discrete Fourier Transform using the Table Look Up Algorithm

A3.1 Introduction

This appendix describes the modifications made to the 'Columns method' algorithm in 3.3.2 by exploiting the $\frac{\pi}{2}$ symmetry of the sinusoidal wave and shuffling the order of the I_n samples. A notation is introduced and a 128 point DFTTLU example is used to clarify the detailed explanation of the algorithm. The explanation is divided as follows:

1) The signed address generator.
2) The shuffling of the time samples, I_n
3) The table look up and accumulate routine
4) The calculation of the resultant harmonics

A3.2 Reducing the Number of Address Pairs from N^2 to $(\frac{N}{4})^2$

Fig.A3.1 shows the calculation steps normally needed to solve an 8 point DFTTLU. N constants from the STV have to be Accumulated to calculate one harmonic and N^2 constants have to be Accumulated to calculate N harmonics. The output of the SAG for N=8 is encoded as in Fig.A3.2 and $\frac{\pi}{2}$ is represented by $\frac{N}{4}$ ie. 2. The K_{nk} components of the address pairs are shown in Fig.A3.3. From this figure it can be seen that if the rows and columns of zeros are ignored, only the first $\frac{N}{4}$ rows or columns need be generated as the remaining addresses are similar

$$F_0 \propto S(I_0K_{00})+S(I_1K_{10})+S(I_2K_{20})+S(I_3K_{30})+S(I_4K_{40})+S(I_5K_{50})+S(I_6K_{60})+S(I_7K_{70})$$

$$F_1 \propto S(I_0K_{01})+S(I_1K_{11})+S(I_2K_{21})+S(I_3K_{31})+S(I_4K_{41})+S(I_5K_{51})+S(I_6K_{61})+S(I_7K_{71})$$

$$F_2 \propto S(I_0K_{02})+S(I_1K_{12})+S(I_2K_{22})+S(I_3K_{32})+S(I_4K_{42})+S(I_5K_{52})+S(I_6K_{62})+S(I_7K_{72})$$

$$F_3 \propto S(I_0K_{03})+S(I_1K_{13})+S(I_2K_{23})+S(I_3K_{33})+S(I_4K_{43})+S(I_5K_{53})+S(I_6K_{63})+S(I_7K_{73})$$

$$F_4 \propto S(I_0K_{04})+S(I_1K_{14})+S(I_2K_{24})+S(I_3K_{34})+S(I_4K_{44})+S(I_5K_{54})+S(I_6K_{64})+S(I_7K_{74})$$

$$F_5 \propto S(I_0K_{05})+S(I_1K_{15})+S(I_2K_{25})+S(I_3K_{35})+S(I_4K_{45})+S(I_5K_{55})+S(I_6K_{65})+S(I_7K_{75})$$

$$F_6 \propto S(I_0K_{06})+S(I_1K_{16})+S(I_2K_{26})+S(I_3K_{36})+S(I_4K_{46})+S(I_5K_{56})+S(I_6K_{66})+S(I_7K_{76})$$

$$F_7 \propto S(I_0K_{07})+S(I_1K_{17})+S(I_2K_{27})+S(I_3K_{37})+S(I_4K_{47})+S(I_5K_{57})+S(I_6K_{67})+S(I_7K_{77})$$

Fig.A3.1 Calculation Steps for 8 point DFTTLU

K_{nk} k=1	SAG output	Conventional Sine Argument
K_{01}	0	0
K_{11}	1	$\frac{\pi}{4}$
K_{21}	2	$\frac{\pi}{2}$
K_{31}	1	$\frac{3\pi}{4}$
K_{41}	-0	π
K_{51}	-1	$\frac{5\pi}{4}$
K_{61}	-2	$\frac{3\pi}{2}$
K_{71}	-1	$\frac{7\pi}{4}$

Fig.A 3.2 SAG Output N=8

k \ n	0	1	2	3	4	5	6	7
0	0	0	0	0	0		0	0
1	0	1	2	1	-0	-1	-2	-1
2	0	2	-0	-2	0	2	-0	-2
3	0	1	-2	1	-0	-1	2	-1
4	0	-0	0	-0	0	-0	0	-0
5	0	-1	2	-1	-0	1	-2	1
6	0	-2	-0	2	0	-2	-0	2
7	0	-1	-2	-1	-0	1	2	1

Fig.A 3.3 K_{nk} Components of Address Pairs N=8

Consider Fig.A3.3. The results of exploiting the symmetry are:

1) The addresses in columns $2n+1$, $n=0 \ldots \frac{N}{2}-1$ have the same absolute values.

2) The addresses in columns $4n+2$, $n=0 \ldots \frac{N}{4}-1$ have the same absolute values.

3) The addresses in columns $4n$, $n=0 \ldots \frac{N}{4}-1$ have the same absolute values.

4) The above 3 points also apply to the corresponding rows of addresses.

5) Only the first $\frac{N}{4}$ rows in the first $\frac{N}{4}$ columns need be generated if the rows and columns of zeros are ignored.

If some of the rows in some of the columns are considered it is possible to extract columns of K_{nk} values that are:

1) Identical in magnitude and sign

2) Identical in magnitude but opposite in sign.

These columns are termed similar. If the odd numbered rows in the odd numbered columns are considered, the resulting columns are similar. The same applies to rows $4k+2$, $k=0 \ldots \frac{N}{4}-1$ in columns $2n+1$, $n=0 \ldots \frac{N}{2}-1$ and rows $4k$, $k=0 \ldots \frac{N}{4}-1$ in columns $2n+1$, $n=0 \ldots \frac{N}{2}-1$. This can be extended to columns $4n+2$ and $4n, n=0 \ldots \frac{N}{4}-1$. Thus there are nine sets of similar columns. Each column in each set of columns is associated with more than one time sample. Thus one column of K_{nk} addresses can be used to derive several columns of address pairs $I_n\ K_{nk}$. For example consider the odd numbered rows in column 1. Samples I_1, I_3, I_5, I_7 share the same column of K_{nk} but K_{nk} has opposite signs for I_5 and I_7. These time samples are also termed similar. This can be extended to the remaining eight sets of similar columns.

The similar time samples follow a pattern similar to the reverse bit notation used in the FFT [5]. The time samples are shuffled in much the same way that they are shuffled in the F.F.T algorithm. This is not apparent in this simple 8 point example and, in A3.5, a 128 point DFTTLU example shows the similar solumns and the shuffled time samples. To simplify explanation, a notation for the nine sets of similar columns is introduced in A3.4.

A3.3 Shuffling the Order of the Time Samples I_n

The time samples are stored in the I_n array in the order in which they are sampled. In A3.5 it is shown that one of the first $\frac{N}{4}$ or $\frac{N}{8}$ columns of K_{nk} addresses can be paired with 4 or 8 I_n values respectively to form 4 or 8 columns of address pairs. The 4 or 8 I_n values do not occur in the sampled order and follow a bit reversed order. The notation in A3.5 and the example in A3.5 facilitate the explanation of the Shuffling Routine in A3.9.

A3.4 Notation

The following points are repeated for clarity:

1) Given an N point DFT/DFTTLU the time samples, I_n, are numbered from n=0 .... N-1.

2) The harmonics, F_k, are numbered from k=0 ... N-1.

3) N (in this case) is a positive integer power of 2.

4) The columns of address pairs (or K_{nk} values) are numbered according to the number of their associated time sample (I_n, n=0 .... N-1).

5) The rows of address pairs are numbered according to the frequency index k, k=0 .... N-1.

Odd numbered time samples and their corresponding columns are termed OTS.

The even numbered columns that are NOT multiples of 4 are termed E$\bar{4}$TS.

The remaining even numbered columns are termed E4TS.

The rows are numbered in a similar way.

The odd numbered rows contribute to the odd numbered harmonics.

The rows are termed OH.

Similarly, the even-numbered rows that are NOT multiples of 4 are termed E$\bar{4}$H.

The remaining rows are termed E4H.

The terminology is used to define the 9 sets of columns of arguments in the following way:

eg. The set of the odd numbered rows in the odd numbered columns of addresses is called OTS/OH.

Similarly, the even numbered rows that are NOT multiples of 4 in the even numbered *columns that are multiples of 4 are* termed E4TS/E$\bar{4}$H.

Thus the 9 sets of columns of addresses are listed as follows:

1) OTS/OH
2) OTS/E$\bar{4}$H
3) OTS/E4H
4) E$\bar{4}$ TS/OH
5) E$\bar{4}$TS/E$\bar{4}$H
6) E$\bar{4}$TS/E4H
7) E4TS/OH
8) E4TS/E$\bar{4}$H
9) E4TS/E4H.

A3.5 A 128 point DFTTLU example

A3.5.1 *The Signed Address Generator SAG output*

The 8 point DFTTLU example is not very useful as there are only 2 points in the range $0 - \frac{\pi}{2}$ rad. A 128 point DFTTLU has points

$$0,1,2 \quad \ldots\ldots \quad 29,\ 30,\ 31,\ 32 \text{ in the range } 0 - \frac{\pi}{2}$$

This notation is used in Fig.A3.4.1 and Fig.A3.4.2 which list the columns of K_{nk} values for the imaginary and real harmonics respectively (cf. Fig.A3.3).

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
0	2	4	6	8	10	.	.	.	.	.	.	.	.	.	30	32	30	28	.	.	.	.	.				.			4	2	-0
0	3	6	9	12	15	.	.			3	31	28	25	.	.			.	7	4	1	-2	-5		.	.	.	.	.	.	-29	-32
0	4	8	12	.	.	.	28	32	28	.	.	.	.	.	.	-0	-4	-8		.	.	.	-28	-32	-28	.	.	.	.	-8	-4	0
0	5	.	.	.	.	.	.		.	.	.	.	.	.	.	.		.		.	.	.	.	.	.	.	.	.	.	.	.	.
0	6	.	.		.	.			.	.	.	.	.	.	.	.	.		.	.		.	.	.	.	.	.	.	.	.	.	.
.	.	.	.		.	.	.		.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.		.	.	.		.	.	.				.	.			.		.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.																									
.																																
.																																
.																																
.																																
.																																
0	16	32	16	-0	-16	32	-16	0	.	.	.	.	.	.		-	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
.																																
0	30																															
0	31																															
0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0

Fig A 3.4.1 SIGNED ARGUMENTS / ADDRESSES (sine) N = 128

32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	-0
32	30	28	26												2	0	-2														-30	-32
32	29	26	23							2	-1	-4								-28	-31	-30							-9	-6	-3	0
32	28	24					4	0	-4						-28	-32	-28						-4	0	4							32
32	27					2	-3					-28	-3	-26					-1	4						30						-0
32	26																															
32	16	-0	-16	-32	-16	0	16																									
32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32	-0	-32	0	32

Fig. A 34.2 SIGNED ARGUMENTS /ADDRESSES (cosine) N = 128

A3.5.2 Shuffling the Time Samples

Fig.A3.5 lists the columns of K_{nk} components of the address pairs that are equal or opposite signed eg. if the set OTS/OH(sine) is considered, columns 1, 63, 65, 127 have K_{nk} values of equal magnitudes but columns 65, 127 have opposite signs to columns 1, 63. Clearly time samples $I_1, I_{63}, I_{65}, I_{127}$ are similar (cf. A3.2). If the set OTS/E4H(sine) is considered columns 9, 23, 41, 55, 73, 87, 105, 119 have K_{nk} values of equal magnitude but the signs of columns 41, 55, 105, 119 are opposite to columns 9, 23, 73, 87. Time samples $I_9, I_{23}, I_{41}, I_{55}, I_{73}, I_{87}, I_{105}, I_{119}$ are similar.

In this case 8 columns of K_{nk} values are similar if even numbered harmonics, which are not multiples of 4 are considered. The remaining 7 sets are interpreted in a similar manner.

The columns of signs in Fig.A3.5 show which columns of K_{nk} values are opposite signed. Consider the listing needed OTS/OH and consider the first column of numbers:

	sine	cosine
1	+	+
63	+	-
65	-	-
127	-	+

This states that the columns of K_{nk} values, 1, 63, 65, 127 have the same magnitude but that columns 65 and 127 are opposite signed to 1 63 when the sine (imaginary) harmonics are considered. This listing also shows that time samples 1, 63, 65 and 127 are similar. The remaining 8 sets of listings are interpreted in the same way.

A3.5.3 Formulae to Shuffle the Time Samples

The nine sets of shuffled I_n in Fig.A3.5 are generated with the following formulae. Consider the formula for the OTS/OH set:

ie.
$$1 + 2n$$
$$\left(\frac{N}{2} - 1\right) - 2n \qquad n = 0\ldots.\frac{N}{8} - 1$$
$$\left(\frac{N}{2} + 1\right) + 2n$$
$$(N - 1) - 2n$$

Here $N = 128$ and consider thecase for $n = 0$.

ie. 1

63

65

127

columns of K_{nk} values numbered 1, 63, 65 and 127 have the same magnitudes and time samples I_1, I_{63}, I_{65}, I_{127} shuffled out of the I_n array.

ODD TIME SAMPLES

Odd	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	+	+
harmonics	63	61	59	57	55	53	51	49	47	45	43	41	39	37	35	33	+	-
	65	67	69	71	73	75	77	79	81	83	85	87	89	91	93	95	-	-
	127	125	123	121	119	117	115	113	111	109	107	105	103	101	99	97	-	+
E$\overline{4}$	1	3	5	7	9	11	13	15									+	+
harmonics	31	29	27	25	23	21	19	17									+	-
	33	35	37	39	41	43	45	47									-	-
	63	61	59	57	55	53	51	49									-	+
	65	67	69	71	73	75	77	79									+	+
	95	93	91	89	87	85	83	81									+	-
	97	99	101	103	105	107	109	111									-	-
	127	125	123	121	119	117	115	113									-	+
E4	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	+	+
harmonics	33	35	37	39	41	43	45	47	49	51	53	55	57	59	61	63	+	+
	65	67	69	71	73	75	77	79	81	83	85	87	89	91	93	95	+	+
	97	99	101	103	105	107	109	111	113	115	117	119	121	123	125	127	+	+

Fig.A3.5.1 Groups of Shuffled Time Samples including sign patterns for N = 128

$E\bar{4}$ TIME SAMPLES

0										
Odd	2	6	10	14	18	22	26	30	+	+
harmonics	62	58	54	50	46	42	38	34	+	-
	66	70	74	78	82	86	90	94	-	-
	126	122	118	114	110	106	102	98	-	+
$E\bar{4}$	2	6	10	14					+	+
harmonics	30	26	22	18					+	-
	34	38	42	46					-	-
	62	58	54	50					-	+
	66	70	74	78					+	+
	94	90	86	82					+	-
	98	102	106	110					-	-
	126	122	118	114					-	+
E4	2	6	10	14	18	22	26	30	+	+
harmonics	34	38	42	46	50	54	58	62	+	+
	66	70	74	78	82	86	90	94	+	+
	48	102	106	110	114	118	122	126	+	+

Fig.A3.5 : Groups of Shuffled Time Samples including sign patterns for N = 128

E4 TIME SAMPLES

										sin	cos
Odd	0	4	8	12	16	20	24	28	(32)	+	+
harmonic	(64)	60	56	52	48	44	40	36	32	+	-
	64	68	72	76	80	84	88	92	(96)	-	-
	(128)	124	120	116	112	108	104	100	96	-	+
$\overline{E4}$	0	4	8	12	16	20	24	28		+	+
harmonic	32	36	40	44	48	52	56	60		-	-
	64	68	72	76	80	84	88	92		+	+
	96	100	104	108	112	116	120	124		-	-
E4											
harmonic	0	4	8	12	16	20	24	28		+	+
	32	36	40	44	48	52	56	60		+	+
	64	68	72	76	80	84	88	92		+	+
	96	100	104	108	112	116	120	124		+	+

Fig.A3.5.3 Groups of Shuffled Time Samples including sign patterns for N = 128

Table A 3.1 Formulae to Shuffle Time Samples

OTS/OH $n=0,1,\ldots(N/8)-1$ $1+2n$ $(\frac{N}{2}-1)-2n$ $(\frac{N}{2}+1)+2n$ $(N-1)-2n$	E$\bar{4}$TS/OH $n=0,1,\ldots(N/16)-1$ $2+4n$ $(\frac{N}{2}-2)-4n$ $(\frac{N}{2}+2)+4n$ $(N-2)-4n$	E4TS/OH $n=0,1,\ldots(N/16)-1$ $0+4n$ $\frac{N}{2}-4n$ $\frac{N}{2}+4n$ $N-4n$
OTS/E$\bar{4}$H $n=0,1,\ldots(N/16)-1$ $1+2n$ $(\frac{N}{4}-1)-2n$ $(\frac{N}{4}+1)+2n$ $(\frac{N}{2}-1)-2n$ $(\frac{N}{2}+1)+2n$ $(\frac{3N}{4}-1)-2n$ $(\frac{3N}{4}+1)+2n$ $(N-1)-2n$	E$\bar{4}$TS/E$\bar{4}$H $n=0,1,\ldots(N/32)-1$ $2+4n$ $(\frac{N}{4}-2)-4n$ $(\frac{N}{4}+2)+4n$ $(\frac{N}{2}-2)-4n$ $(\frac{N}{2}+2)+4n$ $(\frac{3N}{4}-2)-4n$ $(\frac{3N}{4}+2)+4n$ $(N-2)-4n$	E4TS/E$\bar{4}$H $n=0,1,\ldots(N/16)-1$ $0+4n$ $\frac{N}{4}+4n$ $\frac{N}{2}+4n$ $\frac{3N}{4}+4n$
OTS/E4H $n=0,1,\ldots(N/8)-1$ $1+2n$ $(\frac{N}{4}+1)+2n$ $(\frac{N}{2}+1)+2n$ $(\frac{3N}{4}+1)+2n$	E$\bar{4}$TS/E4H $n=0,1,\ldots(N/16)-1$ $2+4n$ $(\frac{N}{4}+2)+4n$ $(\frac{N}{2}+2)+4n$ $(\frac{3N}{4}+2)+4n$	E4TS/E4H $n=0,1,\ldots(N/16)-1$ $0+4n$ $\frac{N}{4}+4n$ $\frac{N}{2}+4n$ $\frac{3N}{4}+4n$

A3.6 *Algorithm to Generate the Signed Addresses (K_{nk})*

A3.6.1 The Four State UP/DOWN Counter

The generator, a software program, must simulate a 4 state UP/DOWN courter. Consider the K_{nk} *values needed for the imaginary* harmonic components. The states are listed in Table A3.2

Table A3.2 The 4 states of the K_{nk} generator.

State	Radian Argument		Signed Radian Argument		Output of Signed Address Generator	
1	$0 - \frac{\pi}{2}$	+	$0 - \frac{\pi}{2}$		+	$0 - \frac{N}{4}$
2	$\frac{\pi}{2} - \pi$	+	$\frac{\pi}{2} - 0$		+	$\frac{N}{4} - 0$
3	$\pi - \frac{3\pi}{2}$	-	$0 - \frac{\pi}{2}$		-	$0 - \frac{N}{4}$
4	$\frac{3\pi}{2} - 2\pi$	-	$\frac{\pi}{2} - 0$		-	$\frac{N}{4} - 0$

To facilitate explanation, *the radian equivalents of* the K_{nk} values are used and clearly $\frac{\pi}{2} \equiv \frac{N}{4}$. In A3.7 the machine code values of K_{nk} are discussed.

A3.6.2 The Starting and Incrementing Arguments

The algorithm requires two input values.

1) The starting argument
2) The incrementing argument.

The starting argument is the first value in a column of K_{nk} values. The incrementing argument is added to the starting argument and the resulting subtotal is the next K_{nk} value in the column. The incrementing argument is repeatedly added to the subtotal to give the remaining K_{nk} values. The problem of having a subtotal greater than $\frac{\pi}{2}$ is discussed in the next section. Assuming the problem of changing from state 1 to state 2 is overcome, the K_{nk} values in state 2 are obtained by repeatedly subtracting the incrementing argument from the subtotal. The problem of having a subtotal less than 0 is also discussed in the next section.

A3.6.3 Subtotal greater than $\frac{\pi}{2}$ (overflow) less than 0, (underflow) and incrementing arguments greater than $\frac{\pi}{2}$

It must be noted that the sign component of the argument is only added to the argument at a later stage. The values here are in the 'unsigned' form.

The maximum value that the register can hold is $\frac{\pi}{2}$. Values greater than $\frac{\pi}{2}$ will appear in the 'overflowed' form while value less than 0 will appear in 'the underflowed' form.

Once overflow or underflow takes place the resulting subtotal must be modified in order to be compatible with the next state. If $\frac{\pi}{2}$ is chosen in the above manner, the modification is simply implemented by taking the '$\frac{\pi}{2}$'s complement' of the overflow or underflow value.

Consider overflow from state 1 to state 2.

eg. $\frac{\pi}{3} + \frac{\pi}{3} = \underline{\frac{\pi}{6}\ \text{overflow}}$ (correct answer is $\frac{2\pi}{3}$)

The $\frac{\pi}{2}$'s complement of $\frac{\pi}{6}$ is $\frac{\pi}{3}$ and.

$\sin\ (\frac{\pi}{3}) \cong \sin\ (\frac{2\pi}{3})$

Consider underflow from state 2 to state 3

eg $\frac{\pi}{6} - \frac{\pi}{3} = \underline{\frac{\pi}{3}\ \text{overflow}}$

The $\frac{\pi}{2}$'s complement of $\frac{\pi}{3}$ is $\frac{\pi}{6}$. In this case state 3 is negative and the output would be $-\frac{\pi}{6}$. By definition

$\sin\ (-\frac{\pi}{6}) \equiv -\sin\ (\frac{\pi}{6})$

In Fig.A3.6 the output of an SAG is given where the starting argument is zero and the incrementing argument is $\frac{\pi}{3}$.

Conventional Argument	0	$\frac{\pi}{3}$	$\frac{2\pi}{3}$	π	$\frac{4\pi}{3}$	$\frac{5\pi}{3}$	2π	State
Signed Argument	0	$\frac{\pi}{3}$	OVF					1
			$\frac{\pi}{3}$	UNF				2
				-0	$-\frac{\pi}{3}$	OVF		3
						$-\frac{\pi}{3}$	UNF	4
							0	1

Fig.A3.6 SAG Output

Adding $\frac{\pi}{2}$ is implemented by taking the $\frac{\pi}{2}$'s complement of the subtotal and moving to the next state. Adding π is implemented by moving to the next state but one.

Adding $\frac{3\pi}{2}$ is implemented by taking the $\frac{\pi}{2}$'s complement of the subtotal and moving to the next state but two.

Increments greater than $\frac{\pi}{2}$ are written in the overflowed form.

eg. an increment of $\frac{3}{4}\pi$ is written as $\frac{\pi}{4}$ and is added by taking the $\frac{\pi}{2}$'s complement of the subtotal and adding $\frac{\pi}{4}$ to the result.

A3.6.4 Flowchart of Algorithm to Generate K_{nk} values

There are four similar algorithms for the four cases of incrementing arguments.

Case 1.	$0 < \text{Inc} < \frac{\pi}{2}$	Fig.A3.7
Case 2.	$\frac{\pi}{2} < \text{Inc} < \pi$	Fig.A3.8
Case 3.	$\pi < \text{Inc} < \frac{3}{2}\pi$	Fig.A3.9
Case 4.	$\frac{3\pi}{2} < \text{Inc} < 2\pi$	Fig.A3.10.

Step 3 processes the argument to conform to the simusoidal 'quantising levels' defined in A1.2.3. The subtotal must not be destroyed in the processing and it is stored in a 'scratch pad' memory location. The sign bit is added in the last step of the processing and is placed in the most significant bit (MSB) position. The cosine value is derived in step 3 by taking the $\frac{\pi}{2}$'s complement of the sine argument and setting the signbit (MSB) to the sign of the following state. The algorithm is entered at step 3 and the start argument is stored in the scratch pad. The Assembler program is in A4.5.

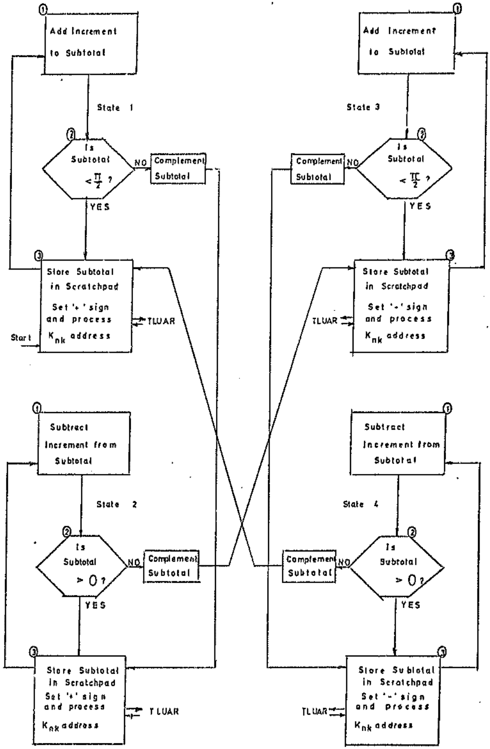


Fig.A3.7 Case1 $0 < \text{inc.} < \frac{\pi}{2}$

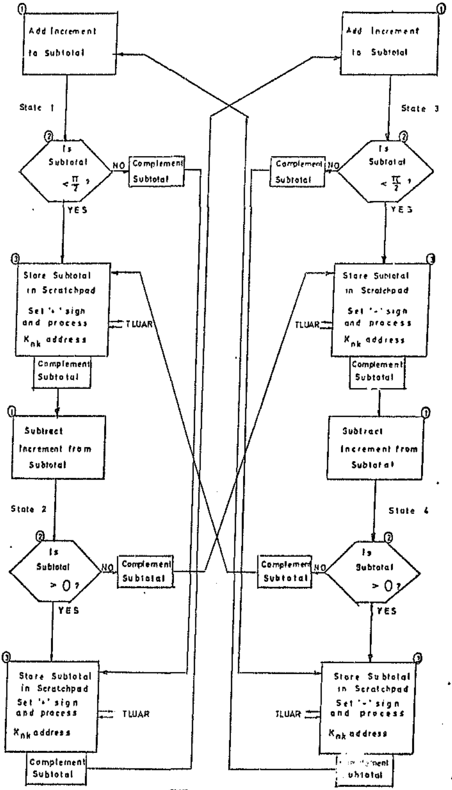


Fig. A 3.8 Case 2 $\frac{\pi}{2} < \text{inc.} < \pi$

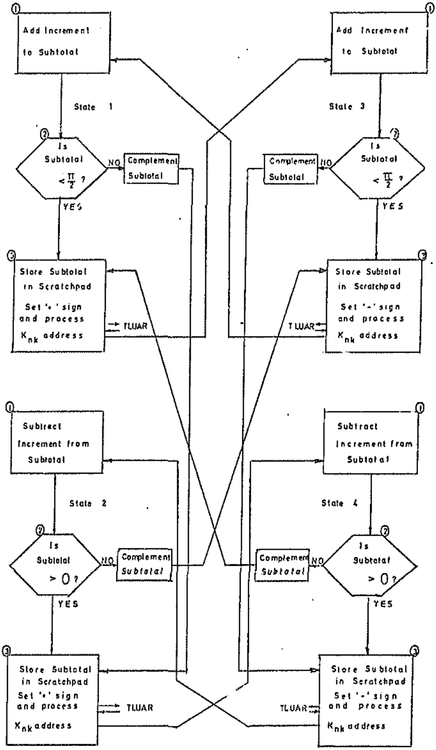


Fig. A3.9 Case 3 $\pi < \text{inc.} < \frac{3\pi}{2}$

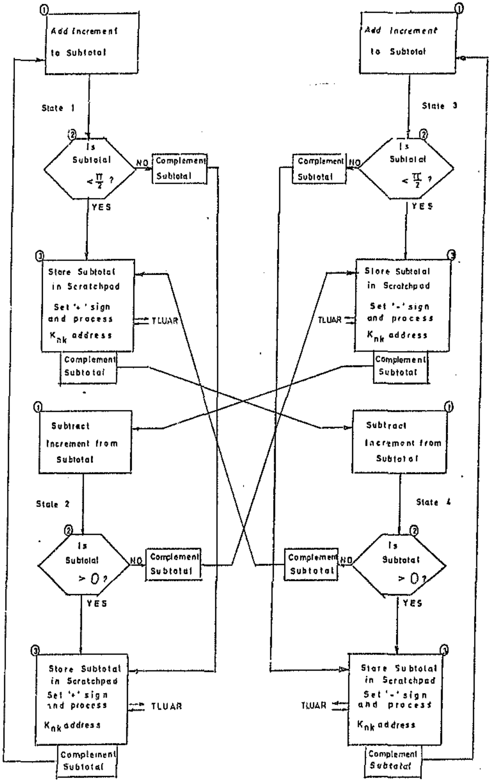


Fig. A 3.10 Case 4 $\frac{3\pi}{2} < \text{inc.} < 2\pi$

A3.7 Expressing the Radian Arguments in Nova 1200 Machine Code

A knowledge of Nova 1200 machine code[4] is assumed and the Assembler program that implements the algorithm in A3.6 is described in A4.5. This section shows how the radian arguments are expressed in terms of machine code and how overflow (subtotals > $\frac{\pi}{2}$) or underflow can be detected using the CARRY bit of the machine. The resulting machine code values of K_{nk} are then modified to facilitate the addressing of the Stored Table of Values (STV).

A3.7.1 Machine Code

The Nova 1200 has a 16 bit word and a CARRY bit which changes state when overflow or underflow occurs. The radian angles are defined in terms of the machine code as follows:

$$0 \text{ rad} \equiv 0_8$$

$$\frac{\pi}{2} \text{ rad} \equiv 0_8 \text{ overflow CARRY bit changes state}$$

Thus taking the two's (2's) complement of a machine code number is equivalent to taking the $\frac{\pi}{2}$'s complement (A3.6.3). Confusion arises because 0 and $\frac{\pi}{2}$ have the same machine code value (ignoring the change of state of the carry bit). However, Fig.A3.4 (for N = 128) shows that this only occurs in 3 of the 9 sets of similar columns - in those sets that extract the even multiples of 4 harmonics. In these 3 sets bo 0 and $\frac{\pi}{2}$ can occur in the same column. This is overcome by the definition of Y_{M-1} and the corresponding integration weight u_{M-1} in A1.2 which includes $\frac{\pi}{2}$ in the final quantising level. This is not correct mathematically as $\frac{\pi}{2}$ is the threshold of the next quantising level Y_M. The algorithm in A3.6 does not take this into account and the simple modification is given later in this section.

Consider an N point DFTTLU

$0 - 2\pi$ is divided into N steps

$0 - \frac{\pi}{2}$ is divided into $\frac{N}{4}$ steps

The 16 bit computer word must have $\frac{N}{4}$ steps. If N = 128 the machine code steps are:

0_8, $4\ 000_8$, $10\ 000_8$ ..... $174\ 000_8$

The final column in the set OTS/OH is:

$4\ 000_8$, $14\ 000_8$, $24\ 000_8$, $34\ 000_8$ .....

Clearly the starting value is $4\ 000_8$ and the incrementing value is $10\ 000_8$.

The layout of the STV is given in A3.10.2 and the K_{nk} values must be processed and signed to address the STV. This requires the K_{nk} values to be in integral multiples of 40_8. There are 16 "sinusoidal quantising abscissas", and the K_{nk} output of the SAG must be in the form

$\pm 0_8$ $\pm 40_8$, $\pm 100_8$ ..... $\pm 700_8$ $\pm 740_8$

This is done by truncating all but the 4 most significant bits (MSB) of the machine code and shifting the remaining 4 bits 7 places to the right.

eg. $14\ 000_8$ $= 0\ 001\ 100\ 000\ 000\ 000_2$

Truncating $0\ 001$ ————————

Shift right 7 X $0\ 000\ 000\ 000\ 100\ 000_2$

$= 40_8$

The machine code equivalent to $\frac{\pi}{2}$ is 0_8 overflow. This value must be converted into 740_8 in terms of the definition in A1.2.4. The algorithm takes the one's complement of 0_8 and then truncates and shifts it as before.

eg one's complement of 0_8	$= 1\ 111\ 111\ 111\ 111\ 111_2$
Truncating	1 111 ———————
Shift right 7X	$0\ 000\ 000\ 111\ 100\ 000_2$
	$= 740_8$

The cosine argument is obtained from the sine argument by taking the two's complement.

eg sine argument $4\ 000_8$	$= 0\ 000\ 100\ 000\ 000\ 000_2$
two's complement	$1\ 111\ 100\ 000\ 000\ 000_2$
	$= 174\ 000_8$

The result must still be truncated and shifted.

A3.8 Derivation of Starting and Incrementing Arguments of K_{nk} for SAG

Each column in each of the 9 sets of columns need 2 inputs:

1) The starting argument SH

2) The incrementing argument SI.

A3.8.1 The Starting Argument

The DFTTLU for N = 128 is used as an example and consider the set of columns of arguments OTS/OH.

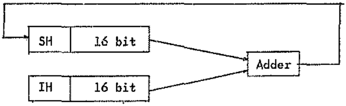
The starting arguments of each column in this set are listed in machine code.

$4\ 000_8$	$14\ 000_8$	$24\ 000_8$	$34\ 000_8$
$44\ 000_8$	$54\ 000_8$	$64\ 000_8$	$74\ 000_8$
$104\ 000_8$	$114\ 000_8$	$124\ 000_8$	$134\ 000_8$
$144\ 000_8$	$154\ 000_8$	$164\ 000_8$	$174\ 000_8$

The difference between each starting argument is $10\ 000_8$

This difference is termed IH.

The values are derived in the following way



The value SH (in this case $4\ 000_8$) is loaded into address SH.

The value IH (in this case $10\ 000_8$) is loaded into address IH.

IH is added to SH and the result is placed in SH to form the next value of SH.

This is repeated until all the start addresses in the set OTS/OH have been generated.

There are 9 sets of columns and SH and IH are listed below for N = 128.

Table A3.3

OTS/OH	SH1	$4\ 000_8$	IH1	$10\ 000_8$
OTS/E$\bar{4}$H	SH2	$10\ 000_8$	IH2	$20\ 000_8$
OTS/E4H	SH3	0_8	IH3	0_8
E$\bar{4}$TS/OH	SH4	$10\ 000_8$	IH4	$20\ 000_8$
E$\bar{4}$TS/E$\bar{4}$H	SH5	$20\ 000_8$	IH5	$40\ 000_8$
E$\bar{4}$TS/E4H	SH6	0_8	IH6	0_8
E4TS/OH	SH7	0_8	IH7	$20\ 000_8$
E4TS/E$\bar{4}$H	SH8	0_8	IH8	$40\ 000_8$
E4TS/E4H	SH9	0_8	IH9	0_8

A3.8.2 The Incrementing Argument

Once again the DFTTLU for N = 128 used as an example.

Consider the set of columns of arguments OTS/OH. The incrementing arguments, SI, of each column in the set are listed in machine code.

10 000	$30\ 000_8$	$50\ 000_8$	$70\ 000_8$
$110\ 000_8$	$130\ 000_8$	$150\ 000_8$	$170\ 000_8$
*$10\ 000_8$	$30\ 000_8$	$50\ 000_8$	$70\ 000_8$
$110\ 000_8$	$130\ 000_8$	$150\ 000_8$	$170\ 000_8$

*The arguments exceed $\frac{\pi}{2}$ and the 'overflow' form is used.

The difference between these values is 20 000_8. This ence is termed II.

The derivation of these values is similar to the derivation for SH, in this case II is added to SI to generate the next value.

<u>Table A3.4</u>

The values of SI and II are listed below for N = 128

OTS/OH	SI1	10 000_8	II1	20 000_8
OTS/E$\bar{4}$H	SI2	20 000_8	II2	40 000_8
OTS/E4H	SI3	20 000_8	II3	40 000_8
E$\bar{4}$TS/OH	SI4	40 000_8	II4	40 000_8
E$\bar{4}$TS/E$\bar{4}$H	SI5	40 000_8	II5	100 000_8
E4TS/E4H	SI6	40 000_8	II6	100 000_8
E4TS/OH	SI7	0_8	II7	40 000_8
E4TS/E$\bar{4}$H	SI8	0_8	II8	100 000_8
E4TS/E4H	SI9	0_8	II9	100 000_8

The basic step value for N = 128 is 4 000_8.

All the above values are derived from 4 000_8 by shifting it the required number of places to the left.

ie. 4 000_8 → 10 000_8 → 20 000_8 → 40 000_8 → 100 000_8

All these starting and incrementing arguments are calculated in subprogram CCL which is covered in A4.8.

A3.8.3 <u>The Starting and Incrementing Arguments and the SAG</u>

Consider the case where the set of columns OTS/OH is being generated.

1) SH1, IH1, S11, II1 are calculated by the CCL subprogram and passed to the start/inc. argument generator.

2) The starting and incrementing arguments are generated and passed to the SAG.

3) The SAG generates a column of K_{nk} values which are processed with the shuffled I_n values to form 4 columns of address pairs.

4) After the column of K_{nk} values have been processed. The start/inc. argument generator generates the starting and incrementing argument of the next column (column 3) and the above steps are repeated.

A3.9 Shuffling Routine

A3.9.1 Introduction

The shuffling routine must sort out the 4 or 8 time samples associated with each column in each of the 9 sets of similar columns by implementing the formulae given in A3.5.3. If the DFTTLU for N = 128 is considered and the first column in the set OTS/OH is being generated the time samples associated with that column are I_1, I_{63}, I_{65} and I_{127}. The next column in that set (column 3) is associated with time samples I_3, I_{61}, I_{67}, I_{125}.

A3.9.2 The Algorithm

The time samples are stored in the I_n array as they are sampled. Their addresses in the array (designated MB) correspond to their position in time (ie. I_n is stored in address n in MB). The algorithm must address and extract the required 4 or 8 similar time samples and place these in 4 or 8 consecutive locations in a data block termed SB. At a later stage these values can be easily accessed by the Table Look Up and Accumulate Routine (TLUAR) (A3.10).

The algorithm need only generate the formulae in A3.5.3 and this is implemented as follows:

The Shuffling Routine must be given the first set of shuffled I_n values in each of the 9 sets of columns. (These shuffled I_n values are calculated by the CCL subprogram.)

Fig. 3.11

The First Group of Shuffled Time Samples

(cf A3.5.3)

Set	n
OTS/OH	$1, \frac{N}{2} - 1, \frac{N}{2} + 1, N - 1$
OTS/E$\bar{4}$H	$1, \frac{N}{4} - 1, \frac{N}{4} + 1, \frac{N}{2} - 1, \frac{N}{2} + 1, \frac{3N}{4} - 1, \frac{3N}{4} + 1, N - 1$
OTS/E4H	$1, \frac{N}{4} + 1, \frac{N}{2} + 1, \frac{3N}{4} + 1$
E$\bar{4}$TS/OH	$2, \frac{N}{2} - 1, \frac{N}{2} + 1, N - 2$
E$\bar{4}$TS/E$\bar{4}$H	$2, \frac{N}{4} - 2, \frac{N}{4} + 2, \frac{N}{2} - 2, \frac{N}{2} + 2, \frac{3N}{4} - 2, \frac{3N}{4} + 2, N - 2$
E$\bar{4}$TS/E4H	$2, \frac{N}{4} + 2, \frac{N}{2} + 2, \frac{3N}{4} + 2$
E4TS/OH	$0, \frac{N}{2}, \frac{N}{2}, N$
E4TS/E$\bar{4}$H	$0, \frac{N}{4}, \frac{N}{2}, \frac{3N}{4}$
E4TS/E4H	$0, \frac{N}{4}, \frac{N}{2}, \frac{3N}{4}$

For example, if the numbers are required for the second column in the set OTS/OH.

2 would be added to the first element in the column

ie. $1 + 2$

2 would be subtracted from the second element in the column.

ie. $(\frac{N}{2} - 1) - 2$

2 would be added to th third element in the column.

ie. $(\frac{N}{2} + 1) + 2$

2 would be subtracted from the fourth element in the column.

in $(N - 1) - 2$

The Assembler program is given in A4.4

A3.10 Table Look Up and Accumulate Routine TLUAR

A3.10.1 Introduction

This routine performs 2 steps.

1) Derives the STV address from the time sample, In, and the signed address, K_{nk}.

2) Obtains the contents of that address and adds/subtracts its to/from its corresponding Accumulator, depending on the signum function.

It must be noted each of the two outputs of the SAG (the sine and cosine arguments) is associated with 4 or 8 shuffled time samples and 8 or 16 address pairs are passed to the TLUAR. The TLUAR stores the cosine K_{nk} address and first processes the sine address pairs. It then retrieves the cosine address and processes it. The format of the STV can be manipulated to simplify the addressing of the STV and the derivation of the signum function. This allows the signum function to be manipulated to take account of the similar columns of K_{nk} addresses. (ie. columns with the same magnitudes but opposite signs.)

A3.10.2 The Format of the STV

The STV is a two dimensional (2 - D) array $S(I_n\ K_{nk})$ where I_n and K_{nk} are the two addresses needed to specify any location in the array. The LADC output, I_n, has 19 levels and a sign bit. (ie. 19 positive levels symmetric with 19 negative levels). The SAG output is quantised into 16 levels and a sign bit (ie. 16 positive levels symmetric with 16 negative levels.)

The sign bits contribute to the signum function and the level addresses are associated with the 19 X 16 2 - D array in Fig.A3.12.

I \ K	0	1	2	3	....				15
0									
1									
2									
⋮									
⋮									
18									

Fig.A3.12 The Two Dimensional Array

In the computer core, 2 - D arrays are transposed into 1 - D arrays (ie. data blocks). The columns of the 2 - D array are placed successively in the core as in Fig.A3.13.

I	K	Contents
0	0	
1	0	
2 . . . 18	0 . . . 0	
0	1	
1	1	
2 . . . 18	1 . . . 1	
- - -	- - -	
0	15	
1	15	
2 . . . 18	15 . . . 15	

Fig.A3.13 2 - D array transposed into 1 - D array.

If the contents of location (3,2) are required from the 1 - D *array* (*ie.* $I_n = 3$, $K_{nk} = 2$), the location is $3 + (2 \times 19) = 41_{10}$ locations from the *first location*. The machine code for 19 is

OCTAL COMPUTER ADDRESS	I OCTAL	K OCTAL	STV CONTENTS
0	0	0	
1	1	0	
⋮	⋮	⋮	
22	22	0	
⋮ 37	⋮ 37	⋮ 0	Waste
40	0	1	
41	0	1	
⋮	⋮	⋮	
62	22	1	
⋮ 77	⋮ 37	⋮ 1	Waste
100	0	2	
101 ⋮ 737	1 ⋮ 37	2 ⋮ 16	
740	0	17	
741	1	17	
⋮	⋮	⋮	
762	22	17	
⋮ 777	⋮ 37	⋮ 17	Waste

Fig. A 3.14   1-D Array. 40_8 Format

10011_2 which is not an easy number for the computer to manipulate. The next largest and easily manipulated number is 32_{10} ie. $100\ 000_2$ and this is chosen as the basis of the 1 - D array. The K_{nk} addresses are thus expressed as integral multiples of 32_{10} ie. 40_8. This results in a waste of core between locations 20_{10} and 32_{10} which are empty. The core is formatted as in Fig.A3.14.

Another method to overcome the problem of the 19 locations in a column of the array is to modify the I_n value instead of the K_{nk} value. The I_n values would be ± 0, $\pm 16_{10}$, $\pm 32_{10}$,...$\pm 288_{10}$. The K_{nk} values would still have to be manipulated to have values between 0_{10} and 15_{10}. Thus both I_n and K_{nk} would have to be manipulated and it was decided rather to waste the core space and manipulate one of the address pair components.

The two address pair components are added using the normal two's complement arithmetic. The magnitude of the result gives the STV location and the sign bit of the result gives the result of the signum function. If the STV is loaded from location X, X is also added to the address pair total to give the STV address.

A3.10.3 Incorporating the Similar Columns of K_{nk} Values into the Signum Function

This is best explained with the aid of the example for N = 128, Fig.A3.5 and the flowchart in Fig.A3.16. The OTS/OH set is considered with the sine K_{nk} value.

The shuffling routine has loaded SB with I_1, I_{63}, I_{65} and I_{127}. The SAG has generated a sine and cosine K_{nk} which are

I_n Sign Bit	K_{nk} Sign Bit	Address Pair Sign Bit	Accumulator Operation
0	0	0	Add
0	1	1	Subtract
1	0	1	Subtract
1	1	0	Add

0 ≡ Positive

1 ≡ Negative

Fig. A 3.15 Signum Function

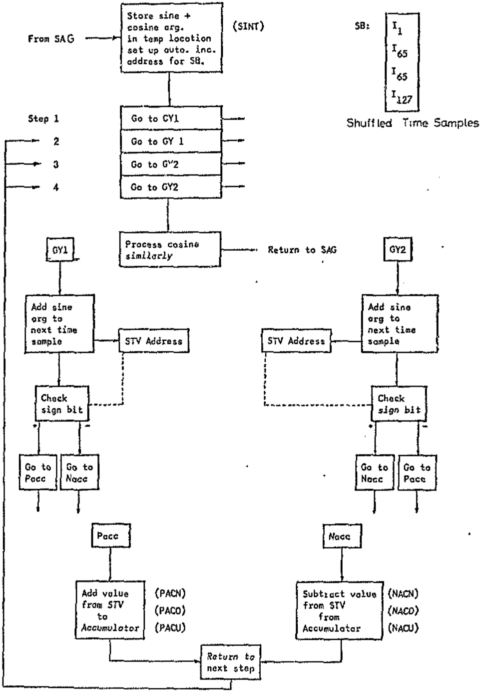


Fig.A3.1 6 Generation and Modification of Signum Function stored in a temporary location as they must not be destroyed during the calculation.

It is added to the sine K_{nk} via path GY1. If the sign bit of the result is positive, the value in the STV, addressed by the magnitude of the result, is added to the Accumulator via path 'PACC'. If the sign bit is negative the STV value is subtracted from the Accumulator via path 'NACC'. The routine proceeds to step 2 and the process is repeated for I_{63}. The next time sample, I_{65}, has a column of K_{nk} arguments with opposite signs. At step 3, I_{65} is added to the sine K_{nk} via path GY2. If the sign bit of the result is positive, the STV constant, addressed by the magnitude of the result, is subtracted from the Accumulator via path 'NACC'. If the sign bit is negative, the STV constant is added to the Accumulator via path 'PACC'. The remaining step is processed similarly. The cosine K_{nk} has the order of the steps changed to GY1, GY2, GY2, GY1.

The step sequences for the remaining sets of columns are given in Fig.A3.5 where '+' and '-' correspond to GY1 and GY2 respectively.

A3.10.4 Accumulating the STV Constant and Checking for Overflow (OVF) or Underflow (UNF)

The STV constants and the Accumulator contents can be either positive or negative numbers. (The particular integration weights used to derive the STV constants are positive in this case but the algorithm is written for the general case.) Checks must be made for OVF or UNF when addition or subtraction occurs. It must be noted that the STV constant is added to or subtracted from the Accumulator. Table A3.5 lists the checks that must be made.

Table A3.5

OVF and UNF Check

Sign Accumulator	Sign STV	Arith. Operation	Check
+	+	Add	OVF
+	-	Add	OK
-	+	Add	OK
-	-	Add	UNF
+	+	Sub	OK
+	-	Sub	OVF
-	+	Sub	UNF
-	-	Sub	OK

Before addition or subtraction can take place, two checks must be made.

1) Are the sign bits of the Accumulator and STV constant the same?

2) Is the Accumulator sign bit positive?

The first check is made using the 'Exclusive OR' routine.

The subprogram that implements the routine can be found in section A4.7.5.

A3.10.5 The Format and Addressing of the Accumulators

The accumulators are divided into 6 blocks.

1a	Odd Imaginary Harmonics	1b	Odd Real Harmonics
2a	$E\bar{4}$ Imaginary Harmonics	2b	$E\bar{4}$ Real Harmonics
3a	E4 Imaginary Harmonocs	3b	E4 Real Harmonics

The auto incrementing address function[+] is used extensively to address the Accumulators. The auto-incrementing addresses are set up in the program that generates the start and incrementing arguments (A4.6). On entering the TLUAR, the auto incrementing location addresses the required Accumulator and stores its contents in a temporary location (see A4.7.6). The 4 or 8 STV constants are added to or subtracted from the temporary location according to the routine described in A3.10.3. The contents of the temporary location are then returned to the correct Accumulator. The method of obtaining the STV address is covered in A3.10.3 and the Assembler program is covered in A4.7.1.

A3.10.5 Calculating the Resultant Harmonics

The format of the Accumulator blocks in the core is given in Fig.A3.17. The real harmonic component and the imaginary harmonic component are denoted a_k and b_k respectively. The harmonics are not in order but are sorted into groups of odd numbered, even multiples of four and the remaining even numbered harmonics. The routine must extract the harmonic components from the Accumulator block in the correct sequence and calculate the resultant (see A4.9).

$$\text{ie.} \quad F_k \propto \sqrt{a_k^{\ 2} + b_k^{\ 2}}$$

The resultant must still be divided by 7 and adjusted by the overall scale factor (see Chapter 4.4)

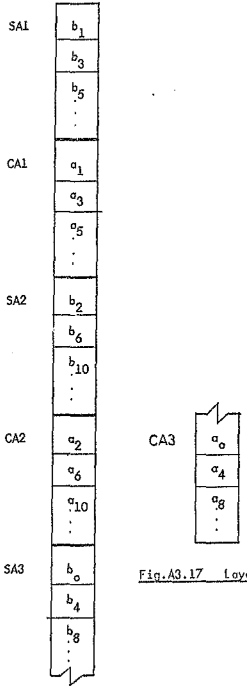


Fig.A3.17 Layout of Accumulators

A4 The Assembler Programme

A4.1 Introduction

The Assembler programme consists of a number of subprograms each covering an aspect of the algorithm presented in Appendix A3 Each subprogram can be run separately with simulated input data so that programming errors can be detected and the subprogram can be modified or corrected without affecting the remaining subprograms. Most of the algorithms in A3 (eg. the SAG) are implemented with nine similar subprograms; one for each set of columns of arguments. This results in a large number of similar, but simple, subprograms which is not efficient programming but it does ease detection of errors.

The subprograms perform the following tasks.

1. Initialize the Accumulator data blocks. ie. load zero into the Accumulator.

2. Calculate constants and the first group of shuffled time sample addresses. (CCL)

3. Sample the signal with the analog to digital converter and store the outputs in the arrays.

4. Shuffle the order of the time samples.

5. Generate the K_{nk} arguments with the signed address generator routine.

6. Combine the K_{nk} and I_n values to form the address pairs of the constant in the STV; obtain the constant and add/subtract it to/from the correct Accumulator.

7. Display or analyse the resulting harmonics.

The Run Program (A4.12) sets the number of bits that represent a constant in the STV and the calls CCL to calculate the constants and addresses for a given N. The Run Program then calls MCN (A4.11), the main control program, which completes the Discrete Fourier Transform.

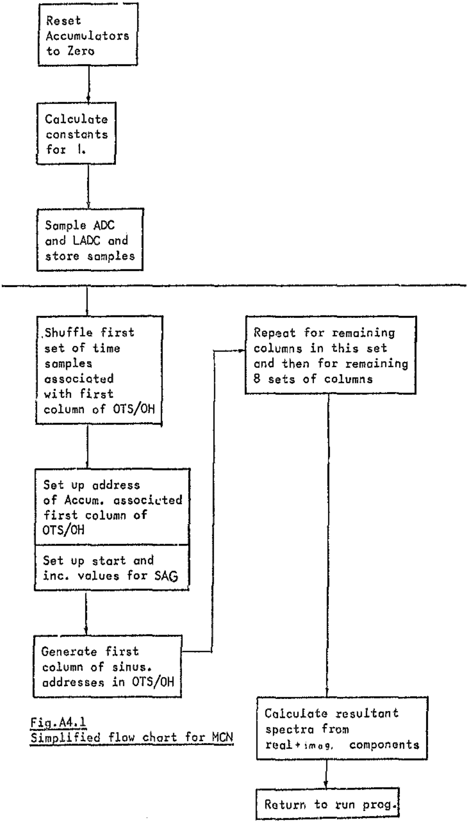


Fig.A4.1
Simplified flow chart for MCN

A4.2 Subprogram to Reset Accumulators

This subprogram loads zero into each location in the Accumulator blocks SA1, CA1, SA2, CA2, SA3, CA3. The number of locations in each block is given by CT1, CT2 and CT3, where CT1 is associated with SA1 and CA1, CT2 with SA2 and CA2, and CT3 with SA3 and CA3. CT1, CT2, and CT3 are calculated in the CCL subprogram.

```
P21:    STA 3,RP21      Save return.
        LDA 2,.CT1 )
                   )    Set up index
        STA 2,.CT1 )
        LDA 2,SA1       Load address of SA1 block
        STA 2,20        in auto inc. location.
        LDA 2,CA1       Load address of CA1 block.
        STA 2,21        in auto inc. location.
        LDA 1,Z         Load zero into register 1
P211:   STA 1,@20       Store zero into next location in SA1
                        block.
        STA 1,@21       Store zero into next location in CA1
                        block.
        DSZ .CT1        Decre ment index
        JMP P211        If index is zero skip  this instruction;
                        if greater than zero go to P211 and
                        initialize next pair of locations in
                        SA1 and CA1.
```

```
                 .TITL   .P21
                 .ENT    SP21
                 .EXTD   SA1,SA2,SA3,CA1,CA2,CA3
                 .EXTD   CT1,CT2,CT3
                 .ZREL
                 .LOC    164
SP21:    P21
                  .NREL
P21      STA     3,RP21

         LDA     2,CT1
         STA     2,.CT1
         LDA     2,SA1
         STA     2,20
         LDA     2,CA1
         STA     2,21
         LDA     1,Z
P211:    STA     1,@20
         STA     1,@21
         DSZ     .CT1
         JMP     P211

         LDA     2,CT2
         STA     2,.CT2
         LDA     2,SA2
         STA     2,20
         LDA     2,CA2
         STA     2,21
         LDA     1,Z
P212:    STA     1,@20
         STA     1,@21
         DSZ      CT2
         JMP     P212

         LDA     2,CT3
         STA     2,.CT3
         LDA     2,SA3
         STA     2,20
         LDA     2,CA3
         STA     2,21
         LDA     1,Z
P213:    STA     1,@20
         STA     1,@21
         DSZ      CT3
         JMP     P213

         JMP     @RP21

 CT1:    0
 CT2:    0
 CT3:    0
Z:       0
RP21:    0
.END
```

A4.3 The Sampling Subprogram

This sampling subprogram is used in the error analysis (A6) and it controls both the ADC and the LADC. Normally the LADC is used and the sections of the program that control the ADC are deleted. The detailed description of the ADC and the LADC is given in references 11 and 10 respectively. Both the ADC and the LADC are triggered simultaneously by an external pulse and the linearly quantised and logarithmically quantised values of the signal are stored in blocks R1 and MB respectively.

```
          LDA 2,MB25 )    Set the auto inc. address for block
                     )
          STA 2,22   )    MB
          LDA 2,N         Set the number of pairs of samples
          STA 2,.N        to be taken.
          LDA 0,C         Send out control signal to
          DOA 0,21        ADC.

          LDA 2,R1        Set the auto. inc. address for
          STA 2,21        block R1
P251:     SKPDN 21  )     Wait until the ADC has processed
                    )
          JMP .-1   )     a sample
          DIAC 1,21       Load the encoded sample value into
                          reg. 1.
          STA 1,@21       Store contents of reg. 1 via auto.
```

	inc. address in R1.
SKPDN 22	Wait until the LADC has processed
JMP .-1	a sample.
DIAC 1,22	Load the encoded sample value into
	reg. 1.
STA 1,@22	Store the contents of reg.1 via auto.
	inc. address in MB.
DSZ .N	Reduce N by 1. If N is zero return
	to MCN program.
JMP P251	If N is not zero, go to P251:
JMP @RP25	Return to MCN.

```
                .TITL   .P25
                .ENT    SP25
                .EXTD   N,R1
                .EXTN   .MB
                .ZREL
                .LOC    166
SP25:   P25
                .NREL
P25:    STA     3,RP25

        LDA     2,MB25
        STA     2,22
        LDA     2,N
        STA     2,.N
        LDA     0,C
        DOA     0,21
        LDA     2,R1
        STA     2,21

P251:   SKPDN   21
        JMP     .-1
        DIAC    1,21
        STA     1,@21
        SKPDN   22
        JMP     .-1
        DIAC    1,22
        STA     1,@22
        DSZ     .N
        JMP     P251

        JMP     @RP25

C:      60
MB25:   .MB
.N:     0
RP25:   0
.END
```

A4.4 The Shuffling Subprograms

There are nine subprograms numbered from P31 to P39, each providing a set of shuffled I_n values for each of the nine sets of similar columns of K_{nk} values. 4 or 8 I_n values are shuffled out of MB and are placed consecutively in SB before the starting and incrementing arguments are generated for the column of K_{nk} values. The program has 2 entry points, P3 X 1 and P3 X 2 (X = 1 ..... 9) and is initially called through P3 X 1. The subsequent shuffled I_n values are obtained by calling P3 X 2.

Subprogram P37 contains a modification as I_o, $I_{N/4}$, $I_{N/2}$ and $I_{3N/4}$ are not similar - I_o and $I_{N/2}$ are similar and $I_{N/4}$ and $I_{3N/4}$ are similar.

The subprogram P31 is explained (set OTS/OH)

```
P311:   STA 3,RP31      Save return
        LDA 3,SB31 )    Set auto. inc. address of
                   )
        STA 3,20   )    block SB.
        LDA 2,MB5       Get the address of the first
                        shuffled time sample (I1)
        STA 2,.MB5      Store this address in a temporary
                        location
        LDA 0,0,2       Load the value of I1 into reg. 0.
        STA 0,@20       Store contents of reg. 0 into first
                        location in SB.
```

Locations MB5, MB14, MB7 and MB16 in program CCL contain the addresses of the first set of shuffled time samples ie I_1, $I_{N/2-1}$, $I_{N/2+1}$ and I_{N-1}. These addresses are stored in temporary locations .MB5, .MB14, .MB7 and .MB16. In part P312 these addresses are processed according to the formulae in A3.5.3.

```
P312:   STA 3, RP31      Save return
        LDA 1,N2.        Load reg 1 with N2. N2. corresponds
                         to 2 which is either added to or
                         subtracted from the shuffled address
                         to obtain the next shuffled address.

        LDA 3,SB31 )     Set auto. inc. address of
                   )
        STA 3,20   )     block SB.
        LDA 2,.MB        Get the first shuffled address
                         stored in the temporary location
        ADDZ 1,2         Add 2 to obtain next shuffled
                         address.
        STA. 2,. MB      Save new address in temporary
                         location.
        LDA 0,0,2        Load reg. 0 with the contents of the
                         new shuffled address.
        STA 0,@20        Store reg. 0. contents in first
                         location in SB.
```

.MB5 now contains the address of I_{1+2}.

```
LDA 2, .MB14
SUBO 1,2.       Subtract 2 to obtain the next
                shuffled address.
STA 2, .MB14
LDA 0,0,2.
STA 0,@20       Store the reg. 0. contents in
                second location in SB.
```

.MB14 now contains address $I_{N/2-1-2}$.

The remaining two shuffled addresses are processed and SB will contain the following time sample values.

SB: I_{1+2}
$I_{N/2-1-2}$
$I_{N/2+1+2}$
I_{N-1-2}.
in that order.

```
                .TITL   .PI1
                .ENT    SPI11,SPI12
                .EXTN   .SB
                .EXTD   MB5,MB14,MB7,MB16
                .ZREL
                .LOC    200
SPI11.  PI11
SPI12:  PI12
                .NREL
PI11:   STA     3,RPI1
        LDA     3,SBI1
        STA     3,20

        LDA     2,MB5
        STA     2,.MB5
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB14
        STA     2,.MB14
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB7
        STA     2,.MB7
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB16
        STA     2,.MB16
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI1

PI12:   STA     3,RPI1
        LDA     1,N2
        LDA     3,SBI1
        STA     3,20

        LDA     2,.MB5
        ADDZ    1,2
        STA     2,.MB5
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB14
        SUBO    1,2
        STA     2,.MB14
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB7
        ADDZ    1,2
        STA     2,.MB7
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB16
        SUBO    1,2
        STA     2,.MB16
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI1
RPI1.   0
.MB5:   0
.MB14:  0
.MB7:   0
.MB16:  0
N2:     2
SBI1:   .SB
.END
```

```
        .TITL   .PI2
        .ENT    SPI21,SPI22
        .EXTN   .SB
        .EXTD   MB5,MB13,MB6,MB14,MB7,MB15,MB8,MB16
        .ZREL
        .LOC    202
SPI21:  PI21
SPI22:  PI22
        .NREL
PI21:   STA     3,RPI2
        LDA     3,SBI2
        STA     3,20

        LDA     2,MB5
        STA     2,.MB5
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB13
        STA     2,.MB13
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB6
        STA     2,.MB6
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB14
        STA     2,.MB14
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB7
        STA     2,.MB7
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB15
        STA     2,.MB15
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB8
        STA     2,.MB8
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB16
        STA     2,.MB16
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI2

PI22:   STA     3,RPI2
        LDA     1,N2
        LDA     3,SBI2
        STA     3,20

        LDA     2,.MB5
        ADDZ    1,2
        STA     2,.MB5
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB13
        SUBO    1,2
        STA     2,.MB13
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB6
        ADDZ    1,2
        STA     2,.MB6
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB14
        SUBO    1,2
        STA     2,.MB14
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB7
        ADDZ    1,2
        STA     2,.MB7
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB15
        SUBO    1,2
        STA     2,.MB15
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB8
        ADDZ    1,2
        STA     2,.MB8
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB16
        SUBO    1,2
        STA     2,.MB16
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI2
RPI2:   0
.MB5:   0
.MB13:  0
.MB6:   0
.MB14:  0
.MB7:   0
.MB15:  0
.MB8:   0
.MB16:  0
N2:     2
SBI2:   .SB
        .END
```

```
                  .TITL   .P33
                  .ENT    SP331,SP332
                  .EXTN   .SB
                  .EXTD   MB5,MB6,MB7,MB8
                  .ZREL
                  .LOC    204
SP331:  P331
SP332:  P332
                  .NREL
P331:   STA       3,RP33
        LDA       3,SB33
        STA       3,20

        LDA       2,MB5
        STA       2,.MB5
        LDA       0,0,2
        STA       0,@20

        LDA       2,MB6
        STA       2,.MB6
        LDA       0,0,2
        STA       0,@20

        LDA       2,MB7
        STA       2,.MB7
        LDA       0,0,2
        STA       0,@20

        LDA       2,MB8
        STA       2,.MB8
        LDA       0,0,2
        STA       0,@20

        JMP       @RP33

P332:   STA       3,RP33
        LDA       1,N2
        LDA       3,SB33
        STA       3,20

        LDA       2,.MB5
        ADDZ      1,2
        STA       2,.MB5
        LDA       0,0,2
        STA       0,@20

        LDA       2,.MB6
        ADDZ      1,2
        STA       2,.MB6
        LDA       0,0,2
        STA       0,@20

        LDA       2,.MB7
        ADDZ      1,2
        STA       2,.MB7
        LDA       0,0,2
        STA       0,@20

        LDA       2,.MB8
        ADDZ      1,2
        STA       2,.MB8
        LDA       0,0,2
        STA       0,@20

        JMP       @RP33
RP33:   0
.MB5:   0
.MB6:   0
.MB7:   0
.MB8:   0
N2:     2
SB33:   .SB
        .END
```

```
                .TITL   .PI4
                .ENT    SPI41,SPI42
                .EXTN   .SB
                .EXTD   MB9,MB18,MB11,MB20
                .ZREL
                .LOC    200
SPI41:  PI41
SPI42:  PI42
                .NREL
PI41:   STA     3,RPI4
        LDA     3,SBI4
        STA     3,20

        LDA     2,MB9
        STA     2,.MB9
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB18
        STA     2,.MB18
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB11
        STA     2,.MB11
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB20
        STA     2,.MB20
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI4

PI42:   STA     3,RPI4
        LDA     1,N4
        LDA     3,SBI4
        STA     3,20

        LDA     2,.MB9
        ADDZ    1,2
        STA     2,.MB9
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB18
        SUBO    1,2
        STA     2,.MB18
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB11
        ADDZ    1,2
        STA     2,.MB11
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB20
        SUBO    1,2
        STA     2,.MB20
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI4
RPI4:   0
.MB9:   0
.MB18:  0
.MB11:  0
.MB20:  0
N4:     4
SBI4:   .SB
        .END
```

```
        .TITL   .P35
        .ENT    SP351,SP352
        .EXTN   .SB
        .EXTD   MB9,MB17,MB10,MB18,MB11,MB19,MB12,MB20
        .ZREL
        .LOC    210
SP351:  P351
SP352:  P352
        .NREL
P351:   STA     3,RF35
        LDA     3,SB35
        STA     3,20

        LDA     2,MB9
        STA     2,.MB9
        LDA     0,0,2
        STA     0,020

        LDA     2,MB17
        STA     2,.MB17
        LDA     0,0,2
        STA     0,020

        LDA     2,MB10
        STA     2,.MB10
        LDA     0,0,2
        STA     0,020

        LDA     2,MB18
        STA     2,.MB18
        LDA     0,0,2
        STA     0,020

        LDA     2,MB11
        STA     2,.MB11
        LDA     0,0,2
        STA     0,020

        LDA     2,MB19
        STA     2,.MB19
        LDA     0,0,2
        STA     0,020

        LDA     2,MB12
        STA     2,.MB12
        LDA     0,0,2
        STA     0,020

        LDA     2,MB20
        STA     2,.MB20
        LDA     0,0,2
        STA     0,020

        JMP     @RF35

P352:   STA     3,RF35
        LDA     1,N4
        LDA     3,SB35
        STA     3,20

        LDA     2,.MB9
        ADDZ    1,2
        STA     2,.MB9
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB17
        SUBO    1,2
        STA     2,.MB17
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB10
        ADDZ    1,2
        STA     2,.MB10
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB18
        SUBO    1,2
        STA     2,.MB18
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB11
        ADDZ    1,2
        STA     2,.MB11
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB19
        SUBO    1,2
        STA     2,.MB19
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB12
        ADDZ    1,2
        STA     2,.MB12
        LDA     0,0,2
        STA     0,020

        LDA     2,.MB20
        SUBO    1,2
        STA     2,.MB20
        LDA     0,0,2
        STA     0,020

        JMP     @RF35
RF35:   0
.MB9:   0
.MB17:  0
.MB10:  0
.MB18:  0
.MB11:  0
.MB19:  0
.MB12:  0
.MB20:  0
N4:     4
SB35:   .SB
.END
```

```
                .TITL   .PI6
                .ENT    SPI61,SPI62
                .EXTN   .SB
                .EXTD   MB9,MB10,MB11,MB12
                .ZREL
                .LOC    212
SPI61:  PI61
SPI62:  PI62
                .NREL
PI61:   STA     3,RPI6
        LDA     3,SBI6
        STA     3,20

        LDA     2,MBS
        STA     2,.MBS
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB10
        STA     2,.MB10
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB11
        STA     2,.MB11
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB12
        STA     2,.MB12
        LDA     0,0,2
        STA     0,@20

        JMP     @RP36

PI62:   STA     3,RP36
        LDA     1,N4
        LDA     3,SB36
        STA     3,20

        LDA     2,.MB9
        ADDZ    1,2
        STA     2,.MB9
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB10
        ADDZ    1,2
        STA     2,.MB10
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB11
        ADDZ    1,2
        STA     2,.MB11
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB12
        ADDZ    1,2
        STA     2,.MB12
        LDA     0,0,2
        STA     0,@20

        JMP     @RPI6
RPI6:   0
.MB9:   0
.MB10:  0
.MB11:  0
.MB12:  0
N4:     4
SBI6:   .SB
.END
```

```
                .TITL    P37
                 ENT     SP371,SP372
                .ENTN    .SB
                .EXTD    MB1,MB3,MB21
                .ZREL
                .LOC     214
SP371:  P371
SP372:  P372
                .NREL
P371:   STA      3,RP37
        LDA      3,SB37
        STA      3,20

        LDA      2,MB1
        STA      2,.MB1
        LDA      0,0,2
        STA      0,@20

        LDA      2,MB3
        STA      2,.MB3
        LDA      0,0,2
        STA      0,@20

        LDA      2,MB3
        STA      2,.MB.3
        LDA      0,0,2
        STA      0,@20

        LDA      2,MB21
        STA      2,.MB21
        LDA      0,0,2
        STA      0,@20

        JMP      @RP37

P372:   STA      3,RP37
        LDA      1,N4
        LDA      3,SB37
        STA      3,20

        LDA      2,.MB1
        ADDZ     1,2
        STA      2,.MB1
        LDA      0,0,2
        STA      0,@20

        LDA      2,.MB3
        SUBO     1,2
        STA      2,.MB3
        LDA      0,0,2
        STA      0,@20

        LDA      2,.MB.3
        ADDZ     1,2
        STA      2,.MB.3
        LDA      0,0,2
        STA      0,@20

        LDA      2,.MB21
        SUBO     1,2
        STA      2,.MB21
        LDA      0,0,2
        STA      0,@20
```

```
                 JMP      @RP37
        RP37:    0
        .MB1:    0
        .MB.3:   0
        .MB3:    0
        .MB21:   0
        N4:      4
        SB37:    .SB
        .END
        ---
        ---
        ---
```

```
                        TITL    .P38
                        .ENT    SP381,SP382
                        .EXTN   .SB
                        .EXTD   MB1,MB2,MB3,MB4
                        .ZREL
                        .LOC    210
SP381:      P381
SP382:      P382
                        .NREL
P381:       STA         3,RP38
            LDA         3,SB38
            STA         3,20

            LDA         2,MB1
            STA         2,.MB1
            LDA         0,0,2
            STA         0,@20

            LDA         2,MB2
            STA         2,.MB2
            LDA         0,0,2
            STA         0,@20

            LDA         2,MB3
            STA         2,.MB3
            LDA         0,0,2
            STA         0,@20

            LDA         2,MB4
            STA         2,.MB4
            LDA         0,0,2
            STA         0,@20

            JMP         @RP38

P382:       STA         3,RP38
            LDA         1,N4
            LDA         3,SB38
            STA         3,20

            LDA         2,.MB1
            ADDZ        1,2
            STA         2,.MB1
            LDA         0,0,2
            STA         0,@20

            LDA         2,.MB2
            ADDZ        1,2
            STA         2,.MB2
            LDA         0,0,2
            STA         0,@20

            LDA         2,.MB3
            ADDZ        1,2
            STA         2,.MB3
            LDA         0,0,2
            STA         0,@20

            LDA         2,.MB4
            ADDZ        1,2
            STA         2,.MB4
            LDA         0,0,2
            STA         0,@20

                    JMP         @RP38
RP38:               0
.MB1:               0
.MB2:               0
.MB3:               0
.MB4:               0
N4:                 4
SB38:               .SB
.END
----
----
----
```

```
                .TITL   .PI39
                .ENT    SPI391,SPI392
                .EXTN   .SB
                .EXTD   MB1,MB2,MB3,MB4
                .ZREL
                .LOC    220
SPI391: PI391
SPI392: PI392
                .NREL
PI391:  STA     3,RP39
        LDA     3,SB39
        STA     3,20

        LDA     2,MB1
        STA     2,.MB1
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB2
        STA     2,.MB2
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB3
        STA     2,.MB3
        LDA     0,0,2
        STA     0,@20

        LDA     2,MB4
        STA     2,.MB4
        LDA     0,0,2
        STA     0,@20

        JMP     @RP39

PI392:  STA     3,RP39
        LDA     1,N4
        LDA     3,SB39
        STA     3,20

        LDA     2,.MB1
        ADDZ    1,2
        STA     2,.MB1
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB2
        ADDZ    1,2
        STA     2,.MB2
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB3
        ADDZ    1,2
        STA     2,.MB3
        LDA     0,0,2
        STA     0,@20

        LDA     2,.MB4
        ADDZ    1,2
        STA     [illegible]
        LDA     0,0,2
        STA     0,@20

        JMP     @RP39
RP39:   0
.MB1:   0
.MB2:   0
.MB3:   0
.MB4:   0
N4:     4
SB39:   .SB
        .END
```

A4.5 Signed Address. Generator Subprogram (K_{nk})

This subprogram follows the flowchart in A3.6.4. The following values are passed to this subprogram from the Initialize SAG subprogram (A4.6).

.H5 the start argument.

.I5 the increment argument

.C5 the number of K_{nk} addresses in a column of K_{nk} addresses.

CSR, MSK1, MSK2 are calculated by the CCL subprogram (A4.8) and modify the K_{nk} values into a multiple of 40_8. Table A4.1 lists the names of the SAG subprograms and is derived as follows.

1) The numbers in the name (ie 61, 62, 63, 64, 611, 612) give the entry point of the TLUAR (A4.7).

ie. this statement in a SAG subprogram, X.61 (X = A,B).

```
JSR     @A61
```

means that the particular column of arguments being generated have a sign pattern.

sine +,+,-,-

cosine +,-,-,+

(See fig. A3.5)

Table A4 1 SAG Subprogram Names

SET	$0 < inc < \frac{\pi}{2}$	$\frac{\pi}{2} < inc < \pi$	$\pi < inc < \frac{3\pi}{2}$	$\frac{3\pi}{2} < inc < 2\pi$	$inc = 0$	$inc = \frac{\pi}{2}$	$inc = \pi$	$inc = \frac{3\pi}{2}$
OTS OH	A61	B61						
OTS E$\bar{4}$H	A62	B62						
OTS E4H	C64	D64	E64	F64				
E$\bar{4}$TS OH	A61	B61						
E$\bar{4}$TS E$\bar{4}$H	A62	B62						
E$\bar{4}$TS E4H	C64	D64	E64	F64				
E4TS OH	A61	B61			I611	J61	K612	
E4TS E$\bar{4}$H	A63	B63	G63	H63	I63	J63	K63	L63
E4TS E4H	C64	D64	E64	F64	I64	M64	N64	O64

Table A42 SAG Increment and Output Values

TYPE	INCREMENT	K_{nk} OUTPUT EQUIVALENT TO
A	$0 < inc < \frac{\pi}{2}$	K_{nk} has all values except those equivalent to 0 and $\frac{\pi}{2}$ radians
B	$\frac{\pi}{2} < inc < \pi$	
G	$\pi < inc < \frac{3\pi}{2}$	
H	$\frac{3\pi}{2} < inc < 2\pi$	
C	$0 < inc < \frac{\pi}{2}$	K_{nk} has all values
D	$\frac{\pi}{2} < inc < \pi$	
E	$\pi < inc < \frac{3\pi}{2}$	
F	$\frac{3\pi}{2} < inc < 2\pi$	
I	$inc = 0$	sine $K_{nk} = 0$ cosine $K_{nk} = \frac{\pi}{2}$
J	$inc = \frac{\pi}{2}$	$K_{nk} = \frac{\pi}{4}$
K	$inc = \pi$	$K_{nk} = 0$ or $\frac{\pi}{4}$
L	$inc = \frac{3\pi}{2}$	$K_{nk} = \frac{\pi}{4}$
M	$inc = \frac{\pi}{2}$	$K_{nk} = 0$ or $\frac{\pi}{2}$
N	$inc = \pi$	sine $K_{nk} = 0$ cosine $K_{nk} = \frac{\pi}{2}$
O	$inc = \frac{3\pi}{2}$	$K_{nk} = 0$ or $\frac{\pi}{2}$

The Assembler codesfor program types ABGH are given first. Types CDEF are explained next. The fina' group, MNO are simple programs that have outputs equivalent to 0 on $\frac{\pi}{2}$ radians.

A4.5.1 <u>Types ABGH</u>

The instructions for state 1 are discussed and the differences between state 1 and the remaining states are listed.

<u>Case 1 State 1</u>

A61:	STA 3,RA61	Save return to MCN
	LDA 0,.H5	Load start argument (arg) into 'scratch pad ' memory.
	JMP OUPP	Go to output process. This corresponds to step 3 in flowchart
SUPP:	LDA 2,.I5	Get inc. arg.
	LDA 0,.H5	Get current subtotal.
1	ADDZ 2,0, SNC	Add inc to the subtotal to form the new current subtotal If CARRY bit is non zero skip next instruction (overflow occurs).
	JMP OUPP	Go to output process (state 1).
	NEGZ 0,0	Carry bit was non zero ie. arg $> \frac{\pi}{2}$
2	JMP ODNP	To to output process of state 2.
OUPP;	STA 0,.H5	Store current K_{nk} subtotal in scratch pad.
	MOVZ 0,1	Move K_{nk} into reg 1 for processing

	NEGZ 0,2	Add $\frac{\pi}{2}$ to sin K_{nk} to obtain cosine K_{nk}. Store in reg 2.
	LDA 3,MSK1	Load reg 3 with truncating mask
	ANDZ 3,1	Truncate sine K_{nk}
	JSR SR	This subroutine shifts the sine and cosine K_{nk} 6 places to the right
3	MOVZR 1,1 MOVZR 2,2	Set the sign bit in the CARRY bit and shift once more to the right.
	JSR @SP61	Go to TLUAR
	DSZ .C5	Return from TLUAR here. Decrement control counter and skip next instruction if .C5 is zero.
4	JMP SUPP	Go to state 1 and generate the next K_{nk} value.
	JMP OTP	Return to MCN as the required number of K_{nk} values have been generated.

The marked statements, numbered from 1 to 4, change in the remaining states. The changes are listed below.

<u>Case 1 State 2</u>

1	SUBO 2,0, SNC JMP ODNP	Subtract inc from subtotal. Go to state 2 output process.

2	JMP	OUPN	Go to state 3 output process.
3	MOVZR	1,1	Set sine sign bit positive
	MOVZR	2,2	Set cosine sign bit negative
4	JMP	SDNP	Go to beginning of state 2

<u>Case 1 State 3</u>

1	ADDZ	2,0,SNC	Add increment to K_{nk} subtotal
	JMP	OUPN	Go to state 3 output process
2	JMP	ODNN	Go to state 4 output process
3	MOVOR	1,1	Set sine sign bit negative
	MOVOR	2,2	Set cosine sign bit negative
4	JMP	SUPN	Go to beginning of state 3

<u>Case 1 State 4</u>

1	SUBO	2,0,SNC	Subtract inc from K_{nk} sub-total
	[illegible]	[illegible]	Go to state 4 output process
2	JMP	OUPP	Go to state 1 output process
3	MOVOR	1,1	Set sine sign bit positive
	MOVZR	2,2	Set cosine sign bit negative
4	JMP	SDNN	Go to beginning of state 4

<u>Case 2</u>

The incrementing argument has values between $\frac{\pi}{2}$ and π.
The 'overflowed form' of the increment is used in this case. (see A3.8.2) The marked statements 2 and 3 are the same as before.

Case 2 State 1

1	NEGZ	0,0	Add $\frac{\pi}{2}$ to current K_{nk} subtotal.
	ADDZ	2,0 SNC	Add inc to subtotal to form next K_{nk} value.
	JMP	OUPP	
4	JMP	SDNP	Go to beginning of state 2.

Case 2 State 2

1	NEGZ	0,0	Subtract $\frac{\pi}{2}$ from current K_{nk} subtotal.
	SUBO	2,0,SNC	Subtract inc from subtotal to form next K_{nk} value.
	JMP	ODNP	
4	JMP	SUPN	Go to beginning of state 3.

Case 2 State 3

1	NEGZ	0,0	Add $\frac{\pi}{2}$ to current K_{nk} subtotal
	ADDZ	2,0,SNC	
	JMP	OUPN	
4	JMP	SDNN	Go to beginning of state 4

Case 2 State 4

1	NEGZ	0,0	Subtract $\frac{\pi}{2}$ from current K_{nk} subtotal.
	SUBO	2,0,SNC	
	JMP	ODNN	
4	JMP	SUPP	Go to beginning of state 1

Case 3

The incrementing argument has values between π and $\frac{3\pi}{2}$.
The 'overflowed' form of the incrementing *argument is used.* π is added by simply skipping one state.

Case 3 State 1

4 JMP SUPN *Go to beginning of state 3.*

Case 3 State 2

4 JMP SDNN Go to beginning of state 4

Case 3 State 3

4 JMP SUPP Go to beginning of state 1

Case 3 State 4

JMP SDNP Go to beginning of state 2

Case 4

The *incrementing arguments* have values between $\frac{3\pi}{2}$ and 2π
The 'overflowed' form of the incrementing argument is used. $\frac{3\pi}{2}$ is added by skipping two states.

Case 4 State 1

1 NEGZ 0,0 Add $\frac{3\pi}{2}$ to current K_{nk} subtotal.

ADDZ 2,0,SNC Add increment to K_{nk} subtotal

JMP OUPP

4	JMP	SDNN	Go to beginning of state 4.

<u>*Case 4 State 2*</u>

1	NEGZ	0,0	Subtract $\frac{3\pi}{2}$ from current K_{nk} subtotal
	SUBO	2,0,SNC	Subtract increment from K_{nk} subtotal
	JMP	ODNP	
4	JMP	SUPP	Go to beginning of state 1

<u>*Case 4 State 3*</u>

1	NEGZ	0,0	Add $\frac{3\pi}{2}$ to current K_{nk} subtotal
	ADDZ	2,0,SNC	Add increment to current K_{nk} subtotal
	JMP	OUPN	
4	JMP	SDNP	Go to beginning of state 2

<u>*Case 4 State 4*</u>

1	NEGZ	0,0	Subtract $\frac{3\pi}{2}$ from current K_{nk} subtotal
	SUBO	2,0,SNC	Subtract increment from current K_{nk} subtotal
	JMP	ODNN	
4	JMP	SUPN	Go to beginning of state 3

A4.5.2 <u>Programs Types CDEF</u>

These programs are similar to types ABGH but K_{nk} can have values equivalent to 0 radians and $\frac{\pi}{2}$ radians. The machine code equivalent to 0 and $\frac{\pi}{2}$ radians is 0_8 and 177777_8 (see definition A1.2.3). 0 radians occurs in states 2 and 4 because of the properties of the machine code, but, by definition, 0 radians is the first K_{nk} argument in states 1 and 3. Thus the instruction to output 0 radians (positive) is in state 4 and the instruction for 0 radians (negative) is in state 2. This problem does not occur with $\frac{\pi}{2}$ radians and the method for dealing with this is given in A3.7.1.

The statements that differ from those in types ABEH are explained.

Case 1 State 1

C64:	STA	3,RC64	Save return to MCN.
	LDA	0,.H5	Load starting arg. into scratch pad memory
	JMP	TDNN	Go to state 4 output routine (start arg $\equiv$ 0 rad.)
SUPP:	LDA	2,.I5	
	LDA	0.H5	
	ADDZ	2,0,SNC	Add inc. to current K_{nk}. Skip next statement if OVF.
	JMP	OUPP	Go to state 1 output routine.
	MOVZ	0,0SNR	Overflow took place. Check whether subtotal is either $> \frac{\pi}{2}$ or $= \frac{\pi}{2}$ Skip next instruction if result is $> \frac{\pi}{2}$

	JMP	TUPP	Result is equal to $\frac{\pi}{2}$
			Go to special routine.
	NEGZ	ODNP	Go to state 2 output process
TUPP:	COMZ	0,1	Take one's complement of K_{nk} argument
	MOVZ	0,2	Cosine K_{nk} is the same as the
			uncomplemented $K_{nk} \equiv \frac{\pi}{2}$
	LDA	3,MSK2	Load special truncating mask
	ANDZ	3,1	Mask out bits in positions
	ANDZ	3,2	6,7,8
	MOVZR	1,1	Set sine sign bit positive
	MOVOR	2,2	Set cosine sign bit negative
	JSP	@SP64	Go to TLUAR
	DSZ	.C5	Decrement control index and skip
			next instruction is result is zero.
	JMP	FUPP	
	JMP	OTP	Return to MCN
FUPP:	LDA	0,.I5	Reg. 0 contains the current K_{nk}
			argument which is zero. Adding the
			increment is implemented by loading
			it into reg. 0.
	NEGZ	0,0	The result of this 'addition' always
			$> \frac{\pi}{2}$ and the 2'S complement must be
			taken
	JMP	ODNP	Go to state 2 output process.

OUPP: This set of instructions is the same as in A4.5.1.

<u>Case 1 State 2</u>

SDNP:	LDA	2,I5	
	LDA	0,.H5	
	SUBO	2,0,SNC	Subtract the increment from the current sine K_{nk} argument. If the sign bit is non zero (result negative) skip the next instruction.
	JMP	ZDNP	
	NEGZ	0,0	Result is negative. Take the 2^{s} complement and go to the state 3 output process
	JMP	OUPN	
ZDNP:	MOVZ	0,0,SZR	Check whether subtotal is ≥ 0. If equal to zero skip next instruction.
	JMP	ODNP	If subtotal > 0 go to state 2 output process.
TDNP:	MOVZ	0,1	sine K_{nk} arg $\equiv 0$ rad
	COMZ	0,2	Add $\frac{\pi}{2}$ to sine K_{nk} to derive. cosine K_{nk}. Here sine $K_{nk} \equiv 0$ and the one's complement is taken to derive the machine code , 740_8.
	LDA	3,MSK2	Load special truncating mask
	ANDZ	3,1	Mask out bits in positions 6,7,8
	ANDZ	3,2	
	MOVOR	1,1	Set sine sign bit negative
	MOVOR	2,2	Set cosine sign bit negative

```
        JSR   @SP64        Go to TLUAR
        DSZ   .C5
        JMP   FDNP
        JMP   OTP
FDNP:   LDA   0,.I5        Same explanation as FUPP.
        JMP   OUPN         Go to state 3 output process
LDNP:                      Same as ODNP in A5.4.1.
```

The remaining 2 state and 3 cases are not explained as they follow a similar pattern to the previous cases in A5.4.1.

```
                .TITL   .A61
                .ENT    SA61
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP61
                .ZREL
                .LOC    300
SA61:   A61
                 NREL
A61:    STA     3,RA61
        LDA     0,.H5
        JMP     OUPP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

OTP:    JMP     @RA61

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
RA61:   0
C5:     0
RSR:    0
.END
```

```
                .TITL   .B61
                .ENT    SB61
                .ENTO   .H5,.I5,.C5
                .ENTO   MSK1,CSR,SP61
                .ZREL
                 LOC    301
SB61:   B61
                 NREL
B61:    STA     3,RB61
        LDA     0,.H5
        JMP     OUFP
SUFP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUFP
        NEGZ    0,0
        JMP     ODNP
OUFP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP61
        DSZ     .C5
        JMP     SUFP
        JMP     OTP

OTP:    JMP     @RB61

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR

RB61:   0
C5:     0
RSR:    0
.END
```

```
                .TITL   .I611
                .ENT    SI611
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK2,SP611
                .ZREL
                .LOC    320
SI611:  I611
                 NREL
I611:   STA     3,RI611
TUPP:   LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP611
        DSZ      C5
        JMP     TUPP
        JMP     OTP
OTP:    JMP     @RI611
RI611:  0
.END
---
---
---

                .TITL   .K612
                .ENT    SK612
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK2,SP612
                .ZREL
                .LOC    322
SK612:  K612
                 NREL
K612:   STA     3,RK612
TDNP:   LDA     0,.H5
        COMZ    0,1
        MOVZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP612
        DSZ     .C5
        JMP     TDNN
        JMP     OTP
TDNN:   LDA     0,.H5
        COMZ    0,1
        MOVZ    0 2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP612
        DSZ     .C5
        JMP     TDNP
        JMP     OTP
OTP:    JMP     @RK612
RK612:  0
.END
---
---
```

```
                      .TITL    .J61
                      .ENT     SJ61
                      .EXTD    .H5, I5,.C5
                      .EXTD    MSK1,CSR,SP61
                      .ZREL
                      .LOC     321
SJ61:     J61
                      .NREL
J61:      STA         3,RJ61
          LDA         0,.H5
          JMP         OUPP
SUPP:     LDA         2,.I5
          LDA         0,.H5
          NEGZ        0,0
          ADDZ        2,0,SNC
          JMP         OUPP
          NEGZ        0,0
          JMP         ODNP
OUPP:     STA         0,.H5
          MOVZ        0,1
          NEGZ        0,2
          LDA         3,MSK1
          ANDZ        3,1
          ANDZ        3,2
          JSR         SR
          MOVZR       1,1
          MOVZR       2,2
          JSR         @SP61
          DSZ         .C5
          JMP         SDNP
          JMP         OTP

SDNP:     LDA         2,.I5
          LDA         0,.H5
          NEGZ        0,0
          SUBO        2,0,SNC
          JMP         ODNP
          NEGZ        0,0
          JMP         OUPN
ODNP:     STA         0,.H5
          MOVZ        0,1
          NEGZ        0,2
          LDA         3,MSK1
          ANDZ        3,1
          ANDZ        3,2
          JSR         SR
          MOVZR       1,1
          MOVOR       2,2
          JSR         @SP61
          DSZ         .C5
          JMP         SUPN
          JMP         OTP

SUPN:     LDA         2,.I5
          LDA         0,.H5
          NEGZ        0,0
          ADDZ        2,0,SNC
          JMP         OUPN
          NEGZ        0,0
          JMP         ODNN
OUPN:     STA         0,.H5
          MOVZ        0,1
          NEGZ        0,2
          LDA         3,MSK1
          ANDZ        3,1
          ANDZ        3,2
          JSR         SR
          MOVOR       1,1
          MOVOR       2,2
          JSR         @SP61
          DSZ         .C5
          JMP         SDNN
          JMP         OTP

SDNN:     LDA         2,.I5
          LDA         0,.H5
          NEGZ        0,0
          SUBO        2,0,SNC
          JMP         ODNN
          NEGZ        0,0
          JMP         OUPP
ODNN:     STA         0,.H5
          MOVZ        0,1
          NEGZ        0,2
          LDA         3,MSK1
          ANDZ        3,1
          ANDZ        3,2
          JSR         SR
          MOVOR       1,1
          MOVZR       2,2
          JSR         @SP61
          DSZ         .C5
          JMP         SUPP
          JMP         OTP
OTP:      JMP         @RJ61
SR:       STA         3,RSR
          LDA         3,CSR
          STA         3,C5
R1:       MOVZR       1,1
          MOVZR       2,2
          DSZ         C5
          JMP         R1
          JMP         @RSR
RJ61:     0
RSR:      0
C5:       0
.END
```

```
---
                .TITL   .A62
                .ENT    SA62
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP62
                .ZREL
                .LOC    302
SA62:   A62
                .NREL
A62:    STA     3,RA62
        LDA     0,.H5
        JMP     OUPP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SDNP
        JMP     OTP
```

```
SUPN.   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

OTP:    JMP     @RA62

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
P1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     P1
        JMP     @RSR
RA62:   0
C5:     0
RSR:    0
 END
---
---
```

```
        .TITL   .B62
        .ENT    SB62
        .EXTD   .H5,.I5,.C5
        .EXTD   MSK1,CSR,SP62
        .ZREL
        .LOC    302
SB62:   B62
        .NREL
B62:    STA     3,PB62
        LDA     0,.H5
        JMP     OUPP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUFN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSP     SP
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SUFN
        JMP     OTP

SUFN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUFN
        NEGZ    0,0
        JMP     ODNN
OUFN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP62
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

OTP:    JMP     @PB62

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
RB62:   0
C5:     0
RSR:    0
.END
```

```
                .TITL   .A63
                .ENT    SA63
                .EXTD   H5,.I5,.C5
                .EXTD   MSK1,CSR,SF63
                 ZREL
                .LOC    310
SA63.   A63
                .NREL
A63:    STA     3,RA63
        LDA     0,.H5
        JMP     OUPP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SUPN.   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SDNN.   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

OTP:    JMP     @RA63

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     .-1
        JMP     @RSR
RA63.   0
C5.     0
RSR:    0
 END
```

```
                .TITL   E63
                .ENT    SB63
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP63
                .ZREL
                .LOC    311
SB63:   B63
                .NREL
B63:    STA     3,RB63
        LDA     0,H5
        JMP     OUPP
SUPP:   LDA     2,I5
        LDA     0,H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SDNP
        JMP     OTP

SDNP:   LDA     2,I5
        LDA     0,H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SUPN
        JMP     OTP

SUPN:   LDA     2,I5
        LDA     0,H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SDNN
        JMP     OTP

SDNN:   LDA     2,I5
        LDA     0,H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SUPP
        JMP     OTP

OTP:    JMP     @RB63

SP:     STA     3,RSP
        LDA     3,CSR
        STA     3,CS
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     CS
        JMP     R1
        JMP     @RSP
RB63:   0
CS:     0
RSP:    0
 .END
```

```
                .TITL   G63
                .ENT    SG63
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP63
                .ZREL
                .LOC    312
SG63:   G63
                .NREL
G63:    STA     3,RG63
        LDA     0,.H5
        JMP     ODNP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

OTP:    JMP     @RG63

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
RG63:   0
RSR:    0
C5:     0
        .END
```

```
                TITL    H63
                .ENT    SH63
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP63
                .ZREL
                .LOC    313
SH63:   H63
                .NREL
H63:    STA     3,RH63
        LDA     0,.H5
        JMP     ODNP
SUPP.   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SDNN.   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

OTP:    JMP     @RH63

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1.     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
RH63    0
RSR     0
C5:     0
 .END
```

```
                .TITL   .I63
                .ENT    SI63
                .EXTD   .H5, .I5, .C5
                .EXTD   MSK2, SR63
                .ZREL
                .LOC    314
SI63:   I63
                .NREL
I63:    STA     3,RI63
TUFF:   LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SR63
        DSZ     .C5
        JMP     TUFF
        JMP     OTP
OTP:    JMP     @RI63
RI63:   0
 .END
```

```
                .TITL   .J63
                .ENT    SJ63
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK1,CSR,SP63
                 ZREL
                .LOC  .  315
SJ63:   J63
                 NREL
J63:    STA     3,RJ63
        LDA     0,.H5
        JMP     OUPP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SDNP
        JMP     OTP

SDNP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SDNN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP63
        DSZ      C5
        JMP     SUPP
        JMP     OTP

OTP:    JMP     @RJ63

SR:     STA     3,RSR
        LDA     1,CSR
        STA     3,C5
R1:     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
RJ63:   0
RSR:    0
C5:     0
 .END
```

```
                        TITL    M63
                        ENT     SM63
                        EXTD     H5,.I5,.C5
                        EXTD    MSK2 SR63
                        ZREL
                        LOC     116
SM63:       M63
                        NREL
M63.        STA       3,RM63
TONP        LDA       0  H5
            COMZ      0,1
            MOVZ      0,2
            STA       0, H5
            LDA       1,MSK2
            ANDZ      3,1
            ANDZ      3,2
            MOVZR     1,1
            MOVOR     2,2
            JSR       @SR63
            DSZ       .C5
            JMP       TONN
            JMP       OTP
TONN.       LDA       0,.H5
            COMZ      0,1
            MOVZ      0,2
            STA       0  H5
            LDA       3,MSK2
            ANDZ      3,1
            ANDZ      3,2
            MOVOR     1,1
            MOVZR     2,2
            JSR       @SR63
            DSZ        C5
            JMP       TONP
            JMP       OTP
OTP         JMP       @RM63
RM63.       0
.END
```

```
        TITL    .L63
        ENT     SL63
        EXTD    .H5,.I5,.C5
        EXTD    MSK1,CSR,SF63
        ZREL
        LOC     317
SL63:   L63
        .NREL
L63:    STA     3,RL63
        LDA     0,.H5
        JMP     ODNP
SUPP:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPP
        NEGZ    0,0
        JMP     ODNP
OUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SDNP    LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNP
        NEGZ    0,0
        JMP     OUPN
ODNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SUPP
        JMP     OTP

SUPN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     OUPN
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SDNP
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ODNN
        NEGZ    0,0
        JMP     OUPP
ODNN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SF63
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

OTP:    JMP     @RL63

SR:     STA     3,RSR
        LDA     3,CSR
        STA     3,CS
R1      MOVZR   1,1
        MOVZR   2,2
        DSZ     CS
        JMP     R1
        JMP     @RSR
RL63:   0
RSR:    0
CS      0
  END
```

```
                TITL    C64
                .ENT    SI64
                .EXTC   .H5, IS, C5
                .EXTD   MSK1,MSK2,SIF,SR64
                ZREL
                .LOC    304
BC64:   C64
                NREL
C64:    STA     3,RC64
        LDA     0,.H5
        JMP     TDWN
SUPP:   LDA     2,.IS
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     QUPP
        MOVZ    0,0,SNR
        JMP     TUPP
        NEGZ    0,0
        JMP     ODNP
TUPP:   COMZ    0,1
        MOVZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SR64
        DSZ     .C5
        JMP     FUPP
        JMP     OTR
FUPP:   LDA     0,.IS
        NEGZ    0,0
        JMP     QDNP
QUPP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SR64
        DSZ     .C5
        JMP     SUPP
        JMP     OTR
```

```
SDNP:   LDA     2,.IS
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     ZDNP
        NEGZ    0,0
        JMP     QUPN
ZDNP:   MOVZ    0,0,SZR
        JMP     ODNP
TDNP:   MOVZ    0,1
        COMZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SR64
        DSZ     .C5
        JMP     FDNP
        JMP     OTR
FDNP:   LDA     0,.IS
        JMP     QUPN
QDNP:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SR64
        DSZ     .C5
        JMP     SDNP
        JMP     OTR
```

```
OTP:    JMP     @RC64
RC64:   0

SUPN:   LDA     2,.I5
        LDA     0,.H5
        ADDZ    2,0,SNC
        JMP     OUPN
        MOVZ    0,0,SNR
        JMP     TUPN
        NEGZ    0,0
        JMP     ODNN
TUPN:   COMZ    0,1
        MOVZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     FUPN
        JMP     OTP
FUPN:   LDA     0,.I5
        NEGZ    0,0
        JMP     ODNN
OUPN:   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     SUPN
        JMP     OTP

SDNN:   LDA     2,.I5
        LDA     0,.H5
        SUBO    2,0,SNC
        JMP     IDNN
        NEGZ    0,0
        JMP     OUPP
IDNN:   MOVZ    0,0,SZR
        JMP     ODNN
TDNN    MOVZ    0,1
        COMZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     FDNN
        JMP     OTP
FDNN    LDA     0,.I5
        JMP     OUPP
ODNN.   STA     0,.H5
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SP
        MOVOR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     SDNN
        JMP     OTP

SP      STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1.     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
C5:     0
RSR     0
 END
```

```
              .TITL    .E64
              .ENT     SE64
              .EXTD    .H5, .I5, .C5
              .EXTD    MSK1, MSK2, CSR, SP64
              .IFEL
              .LOC     306
SE64:   E64
              .NFEL
E64:    STA    3, RE64
        LDA    0, .H5
        JMP    TONN
SUPP:   LDA    2, .I5
        LDA    0, .H5
        ADDZ   2, 0, SNC
        JMP    OUPP
        MOVZ   0, 0, SNR
        JMP    TUPP
        NEGZ   0, 0
        JMP    OONP
TUPP:   COMZ   0, 1
        MOVZ   0, 2
        LDA    3, MSK2
        ANDZ   3, 1
        ANDZ   3, 2
        MOVZR  1, 1
        MOVOR  2, 2
        JSR    @SP64
        DSZ    .C5
        JMP    FUPP
        JMP    OTP
FUPP:   LDA    0, .I5
        NEGZ   0, 0
        JMP    OONN
OUPP:   STA    0, .H5
        MOVZ   0, 1
        NEGZ   0, 2
        LDA    3, MSK1
        ANDZ   3, 1
        ANDZ   3, 2
        JSR    SR
        MOVZR  1, 1
        MOVZR  2, 2
        JSR    @SP64
        DSZ    .C5
        JMP    SUPN
        JMP    OTP

SONP:   LDA    2, .I5
        LDA    0, .H5
        SUBO   2, 0, SNC
        JMP    ZONP
        NEGZ   0, 0
        JMP    OUPN
ZONP:   MOVZ   0, 0, SZR
        JMP    OONP
TONP:   MOVZ   0, 1
        COMZ   0, 2
        LDA    3, MSK2
        ANDZ   3, 1
        ANDZ   3, 2
        MOVOR  1, 1
        MOVOR  2, 2
        JSR    @SP64
        DSZ    .C5
        JMP    FONP
        JMP    OTP
FONP:   LDA    0, .I5
        JMP    OUPP
OONP:   STA    0, .H5
        MOVZ   0, 1
        NEGZ   0, 2
        LDA    3, MSK1
        ANDZ   3, 1
        ANDZ   3, 2
        JSR    SR
        MOVZR  1, 1
        MOVOR  2, 2
        JSR    @SP64
        DSZ    .C5
        JMP    SONN
        JMP    OTP
```

```
OTP:    JMP     @RE64
RE64.   0

SUFN:   LDA     2, IS
        LDA     0, HS
        ADDZ    2,0,SNC
        JMP     OUFN
        MOVZ    0,0,SNR
        JMP     TUFN
        NEGZ    0,0
        JMP     OONN
TUFN    COMZ    0,1
        MOVZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @RE64
        DSZ      C5
        JMP     FUFN
        JMP     OTP
FUFN.   LDA     0, IS
        NEGZ    0,0
        JMP     OONP
OUFN.   STA     0, HS
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @RE64
        DSZ      C5
        JMP     SUFP
        JMP     OTP

SONN    LDA     2, IS
        LDA     0, HS
        SUBO    2,0,SNC
        JMP     ZONN
        NEGZ    0,0
        JMP     OUFP
ZONN:   MOVZ    0,0,SZR
        JMP     OONN
TONN:   MOVZ    0,1
        COMZ    0,2
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @RE64
        DSZ      C5
        JMP     FONN
        JMP     OTP
FONN:   LDA     0, IS
        JMP     OUFN
OONN.   STA     0, HS
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MSK1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SR
        MOVZR   1,1
        MOVZR   2,2
        JSR     @RE64
        DSZ      C5
        JMP     SONP
        JMP     OTP

SR.     STA     3,RSR
        LDA     3,CSR
        STA     3,C5
R1.     MOVZR   1,1
        MOVZR   2,2
        DSZ     C5
        JMP     R1
        JMP     @RSR
C5      0
RSR.    0
.END
```

```
                .TITL   .F64
                .ENT    SF64
                .EXTD   HS, IS, CS
                .EXTD   MEM1,MEM2,CSR,SF64
                .ZREL
                .LOC    207
SF64:   F64
                .NREL
F64:    STA     3,RF64
        LDA     0,HS
        JMP     TDNF
SUFF:   LDA     2,IS
        LDA     0,HS
        NEGZ    0,0
        ADDZ    2,0,SNC
        JMP     DUFF
        MOVZ    0,0,SNR
        JMP     TUFF
        NEGZ    0,0
        JMP     ODNF
TUFF:   COMZ    0,1
        MOVZ    0,2
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SF64
        DSZ     CS
        JMP     FUFF
        JMP     OTF
FUFF:   LDA     0,IS
        JMP     QUFF
QUFF:   STA     0,HS
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MEM1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SF
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SF64
        DSZ     CS
        JMP     SDNF
        JMP     OTF

SDNF:   LDA     2,IS
        LDA     0,HS
        NEGZ    0,0
        SUBO    2,0,SNC
        JMP     ZDNF
        NEGZ    0,0
        JMP     QUFF
ZDNF:   MOVZ    0,0,SZR
        JMP     ODNF
TDNF:   MOVZ    0,1
        COMZ    0,2
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVOR   1,1
        MOVOR   2,2
        JSR     @SF64
        DSZ     CS
        JMP     FDNF
        JMP     OTF
FDNF:   LDA     0,IS
        NEGZ    0,0
        JMP     ODNF
ODNF:   STA     0,HS
        MOVZ    0,1
        NEGZ    0,2
        LDA     3,MEM1
        ANDZ    3,1
        ANDZ    3,2
        JSR     SF
        MOVZR   1,1
        MOVOR   2,2
        JSR     @SF64
        DSZ     CS
        JMP     SUFF
        JMP     OTF
```

```
OTP:     JMP     @RF64
RF64:    0

SUFN:    LDA     2,.IS
         LDA     0,.HS
         NEGZ    0,0
         ADDZ    2,0,SNC
         JMP     OUFN
         MOVZ    0,0,SNR
         JMP     TUFN
         NEGZ    0,0
         JMP     ODNN
TUFN:    COMZ    0,1
         MOVZ    0,2
         LDA     3,MSK2
         ANDZ    3,1
         ANDZ    3,2
         MOVOR   1,1
         MOVZR   2,2
         JSR     @SP64
         DSZ     .CS
         JMP     FUFN
         JMP     OTP
FUFN:    LDA     0,.IS
         JMP     OUFN
OUFN:    STA     0,.HS
         MOVZ    0,1
         NEGZ    0,2
         LDA     3,MSK1
         ANDZ    3,1
         ANDZ    3,2
         JSR     SR
         MOVOR   1,1
         MOVOR   2,2
         JSR     @SF64
         DSZ     .CS
         JMP     SDNF
         JMP     OTP

SDNN:    LDA     2,.IS
         LDA     0,.HS
         NEGZ    0,0
         SUBO    2,0,SNC
         JMP     ZDNN
         NEGZ    0,0
         JMP     OUPF
ZDNN:    MOVZ    0,0,SZR
         JMP     ODNN
TDNN:    MOVZ    0,1
         COMZ    0,2
         LDA     3,MSK2
         ANDZ    3,1
         ANDZ    3,2
         MOVZR   1,1
         MOVZR   2,2
         JSR     @SF64
         DSZ     .CS
         JMP     FDNN
         JMP     OTP
FDNN:    LDA     0,.IS
         NEGZ    0,0
         JMP     ODNN
ODNN:    STA     0,.HS
         MOVZ    0,1
         NEGZ    0,2
         LDA     3,MSK1
         ANDZ    3,1
         ANDZ    3,2
         JSR     SR
         MOVOR   1,1
         MOVZR   2,2
         JSR     @SF64
         DSZ     .CS
         JMP     SUFN
         JMP     OTP

SR:      STA     3,RSR
         LDA     3,CSR
         STA     3,CS
R1:      MOVZR   1,1
         MOVZR   2,2
         DSZ     CS
         JMP     R1
         JMP     @RSR
CS:      0
RSR:     0
         END
```

```
                .TITL   I64
                .ENT    SI64
                .EXTD   .H5,.I5,.C5
                .EXTD   MSK2,SR64
                .ZREL
                .LOC    333
SI64:   I64
                .NREL
I:      STA     3,RI64
TUR:    LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SR64
        DSZ     .C5
        JMP     TURR
        JMP     OTR
OTR:    JMP     @RI64
RI64    0
.END
---
---
---
```

```
                .TITL   .M64
                .ENT    SM64
                .EXTD    H5,.I5,.C5
                .EXTD   MSK2,SP64
                .ZREL
                 LOC    I24
SM64.   M64
                 NREL
M64:    STA     3,RM64
TUFF    LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TDNF
        JMP     OTP
TDNF:   LDA     0,.H5
        COMZ    0,1
        MOVZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TUFN
        JMP     OTP
TUFN:   LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TDNN
        JMP     OTP
TDNN.   LDA     0,.H5
        COMZ    0,1
        MOVZ    0,2
        STA     0,.H5
        LDA     3,MSK2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TUFF
        JMP     OTP

OTP:    JMP     @RM64
RM64    0
 END
```

```
                        .TITL   .N64
                        .ENT    SN64
                        .EXTD   .H5,.I5,.C5
                        .EXTD   MSK2,SP64
                        .ZREL
                        .LOC    325
SN64:       N64
                        .NREL
N64:        STA         3,RN64
TUFP:       LDA         0,.H5
            MOVZ        0,1
            COMZ        0,2
            STA         0,.H5
            LDA         3,MSK2
            ANDZ        3,1
            ANDZ        3,2
            MOVZR       1,1
            MOVZR       2,2
            JSR         @SP64
            DSZ         .C5
            JMP         TUFN
            JMP         OTP
TUFN:       LDA         0,.H5
            MOVZ        0,1
            COMZ        0,2
            STA         0,.H5
            LDA         3,MSK2
            ANDZ        3,1
            ANDZ        3,2
            MOVOR       1,1
            MOVOR       2,2
            JSR         @SP64
            DSZ         .C5
            JMP         TUFP
            JMP         OTP

OTP:        JMP         @RN64
RN64:       0
 END
---
---
---
```

```
                .TITL   O64
                .ENT    SO64
                .EXTD   .H5, .I5, .C5
                .EXTD   MEM2, SP64
                ZREL
                LOC     326
SO64:   O64
                NREL
O64     STA     3,RO64
TUFF:   LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TDNN
        JMP     OTF
TDNF    LDA     0,.H5
        COMZ    0,1
        MOVZ    0,2
        STA     0,.H5
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TUFF
        JMP     OTF
TUFN:   LDA     0,.H5
        MOVZ    0,1
        COMZ    0,2
        STA     0,.H5
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TDNF
        JMP     OTF
TDNN    LDA     0,.H5
        COMZ    0,1
        MOVZ    0,2
        STA     0,.H5
        LDA     3,MEM2
        ANDZ    3,1
        ANDZ    3,2
        MOVZR   1,1
        MOVZR   2,2
        JSR     @SP64
        DSZ     .C5
        JMP     TUFN
        JMP     OTF

OTF:    JMP     @RO64
RO64    0
 END
```

A4.6 Subprogram to Generate Starting and Incrementing Arguments for the SAG

There are nine subprograms numbered from Q41 to Q49, each program providing starting and incrementing arguments for each of the nine sets of columns of K_{nk} arguments. The subprogram is executed before a column of K_{nk} arguments is generated. The subprogram is initially entered through SP4X1 (X = 1, ... 9) and subsequently entered through SP4X2 for the remaining starting and incrementing arguments. The auto incrementing address locations of the Accumulators, which are needed by the TLUAR, are also set up by the subprogram. Subprogram Q41 is only explained as the remaining subprograms are similar.

```
P411: STA    3,RP41        Save return to MCN
      LDA    1,SH1         Set first start arg. (calculated by
                           CCL).
      LDA    2,SI          Get first inc. arg. (calculated by
                           CCL).
      JMP    P410          Go to statement P410.
P412: STA    3,RP41        Save return to MCN
      LDA    1,.SHI        Get current start arg.
      LDA    3,IH1  )      Generate next current start arg.
                    )
      ADDZ   3,1    )
      LDA    2,.SI1        Get current inc. arg.
      LDA    0,II1  )      Generate next current inc. arg.
                    )
      ADDZ   0,2    )
```

```
P410 : STA    1,.SHI        store current start arg.
       STA    2,.SI1        store current inc. arg.
       STA    1,.H5         pass current start arg. to SAG
       STA    2,.I5         pass current inc. arg. to SAG.
       LDA    1,SA1         These instructions set
       STA    1,20          the auto. inc. address
       STA    1,21          Locations of the Accumulators
       LDA    1,CA1         which will be used by
       STA    1,22          the TLUAR.
       STA    1,23
       LDA    1,CT1         Passthe number of
       STA    1,.C5         Knk arg. that must be generated
                            to the SAG.
       JMP    @RP41         Return to MCN.
.SH1   Current starting arg. scratch pad memory.
.SI1   Current inc arg. scratch pad memory.
RP41   Location of return address.
```

```
          TITL    041
         .ENT     SF411,SF412
          EXTD     H5, I5, C5
         .EXTD    SH1,IH1,SI1,II1,CT1,SR1,CA1
          ZREL
         .LOC     222
SF411:   P411
SF412:   P412
          NREL
P411:    STA      3,RP41
         LDA      1,SH1
         LDA      2,SI1
         JMP      P410

P412.    STA      3,RP41
         LDA      1,.SH1
         LDA      3,IH1
         ADDZ     3,1
         LDA      2,.SI1
         LDA      0,II1
         ADDZ     0,2

P410.    STA      1,.SH1
         STA      2,.SI1
         STA      1,.H5
         STA      2,.I5

         LDA      1,SR1
         STA      1,20
         STA      1,21
         LDA      1,CA1
         STA      1,22
         STA      1,23

         LDA      1,CT1
         STA      1,.C5

         JMP      @RP41
.SH1:    0
.SI1     0
RP41     0
 END
```

```
         .TITL    042
         .ENT     SF421,SF422
          EXTD     .H5, .I5, .C5
         .EXTD    SH2,IH2,SI2,II2,
                    CT2, SA2, CA2,
         .ZREL
         .LOC     224
SF421    P421
SF422:   P422
          NREL
P421.    STA      3,RP42
         LDA      1,SH2
         LDA      2,SI2
         JMP      P420

P422     STA      3,RP42
         LDA      1,.SH2
         LDA      3,IH2
         ADDZ     3,1
         LDA      2,.SI2
         LDA      0,II2
         ADDZ     0,2

P420.    STA      1,.SH2
         STA      2,.SI2
         STA      1,.H5
         STA      2,.I5

         LDA      1,SR2
         STA      1,20
         STA      1,21
         LDA      1,CA2
         STA      1,22
         STA      1,23

         LDA      1,CT2
         STA      1,.C5

         JMP      @RP42
.SH2:    0
.SI2     0
RP42:    0
.END
```

```
                    .TITL   .P43
                    .ENT    SP431,SP432
                    .EXTD   .H3,.I3,.C3
                    .EXTD   SH3,IH3,SI3,II3,CT3,SA3,CA3
                    .ZREL
                    .LOC    220
SP431:   P431
SP432:   P432
                    .NREL
P431:    STA        3,RP43
         LDA        1,SH3
         LDA        2,SI3
         JMP        P430
P432:    STA        3,RP43
         LDA        1,.SH3
         LDA        3,IH3
         ADDZ       3,1
         LDA        2,.SI3
         LDA        0,II3
         ADDZ       0,2
P430:    STA        1,.SH3
         STA        2,.SI3
         STA        1,.H3
         STA        2,.I5

         LDA        1,SA3
         STA        1,20
         STA        1,21
         LDA        1,CA3
         STA        1,22
         STA        1,23

         LDA        1,CT3
         STA        1,.C5

         JMP        @RP43
.SH3:    0
.SI3:    0
RP43:    0
.END
```

```
                    .TITL   .P44
                    .ENT    SP441,SP442
                    .EXTD   .H7,.I3,.C3
                    .EXTD   SH4,IH4,SI4,II4,
                              CT1,SA1,CA1
                    .ZREL
                    .LOC    220
SP441:   P441
SP442:   P442
                    .NREL
P441:    STA        3,RP44
         LDA        1,SH4
         LDA        2,SI4
         JMP        P440

P442:    STA        3,RP44
         LDA        1,.SH4
         LDA        3,IH4
         ADDZ       3,1
         LDA        2,.SI4
         LDA        0,II4
         ADDZ       0,2

P440:    STA        1,.SH4
         STA        2,.SI4
         STA        1,.H5
         STA        2,.I5

         LDA        1,SA1
         STA        1,20
         STA        1,21
         LDA        1,CA1
         STA        1,22
         STA        1,23

         LDA        1,CT1
         STA        1,.C5

         JMP        @RP44
.SH4:    0
.SI4:    0
RP44:    0
.END
```

```
                TITL    046
                .ENT    SP461, SP462
                .EXTD   .H5, I5, C5
                .EXTD   SH6, IH6, SI6, II6, CT3, SA3, CA3
                .ZREL
                .LOC    234
SP461:  P461
SP462.  P462
                .NREL
P461:   STA     3, RP46
        LDA     1, SH6
        LDA     2, SI6
        JMP     P460

P462    STA     3, RP46
        LDA     1, SH6
        LDA     3, IH6
        ADDZ    3, 1
        LDA     2, .SI6
        LDA     0, II6
        ADDZ    0, 2

P460.   STA     1, SH6
        STA     2, .SI6
        STA     1, .H5
        STA     2, .I5

        LDA     1, SA3
        STA     1, 20
        STA     1, 21
        LDA     1 CA3
        STA     1, 22
        STA     1, 23

        LDA     1, CT3
        STA     1, .C5

        JMP     @RP46
.SH6:   0
.SI6.   0
RP46:   0
 END
```

```
                .TITL   .045
                .ENT    SP451, SP452
                .EXTD   .H5, .I5, C5
                .EXTD   SH5, IH5, SI5, II5 &
                           CT2, CA2, SA2
                .ZREL
                .LOC    232
SP451:  P451
SP452:  P452
                .NREL
P451:   STA     3, RP45
        LDA     1, SH5
        LDA     2, SI5
        JMP     P450

P452:   STA     3, RP45
        LDA     1, SH5
        LDA     3, IH5
        ADDZ    3, 1
        LDA     2, .SI5
        LDA     0, II5
        ADDZ    0, 2

P450.   STA     1, .SH5
        STA     2, SI5
        STA     1, H5
        STA     2, .I5

        LDA     1, SA2
        STA     1, 20
        STA     1, 21
        LDA     1, CA2
        STA     1, 22
        STA     1, 23

        LDA     1, CT2
        STA     1, .C5

        JMP     @RP45
.SH5.   0
.SI5:   0
RP45.   0
.END
```

```
                .TITL   .048
                .ENT    SP481,SP482,SP483
                .EXTD   .H5,.I5,.C5
                .EXTD   SH8,IH8,SI8,II8,CT2,SA2,CA2
                .ZREL
                .LOC    240
SP481.  P481
SP482.  P482
SP483   P483
                .NREL
P481:   STA     3,RP48
        LDA     1,SH8
        LDA     2,SI8
        JMP     P480

P483:   STA     3,RP48
        LDA     1,.SH8
        LDA     3,IH8
        SUBO    3,1
        JMP     P484

P482:   STA     3,RP48
        LDA     1,.SH8
        LDA     3,IH8
        ADDZ    3,1
P484.   LDA     2,.SI8
        LDA     0,II8
        ADDZ    0,2

P480.   STA     1,.SH8
        STA     2,.SI8
        STA     1,.H5
        STA     2,.I5

        LDA     1,SA2
        STA     1,20
        STA     1,21
        LDA     1,CA2
        STA     1,22
        STA     1,23

        LDA     1,CT2
        STA     1,.C5

        JMP     @RP48
.SH8:   0
.SI8.   0
RP48:   0
.END
```

```
                .TITL   .047
                .ENT    SP471,SP472
                .EXTD   .H5,.I5,.C5
                .EXTD   SH7,IH7,SI7,II7,CT1
                                SA1,CA1
                .ZREL
                .LOC    226
SP471.  P471
SP472.  P472
                NREL
P471:   STA     3,RP47
        LDA     1,SH7
        LDA     2,SI7
        JMP     P470

P472:   STA     3,RP47
        LDA     1,.SH7
        LDA     3,IH7
        ADDZ    3,1
        LDA     2,.SI7
        LDA     0,II7
        ADDZ    0,2

P470:   STA     1,.SH7
        STA     2,.SI7
        STA     1,.H5
        STA     2,.I5

        LDA     1,SA1
        STA     1,20
        STA     1,21
        LDA     1,CA1
        STA     1,22
        STA     1,23

        LDA     1,CT1
        STA     1,.C5

        JMP     @RP47
.SH7:   0
.SI7    0
RP47    0
END
```

```
                        .TITL   643
                        .ENT    SF431,SF432
                        .EXTD   .H5, I5, .C5
                        .EXTD   SH3 IH3 SI3.II3 CTI.SAI,CAI
                        .ZREL
                        .LOC    143
SF431:  F431
SF432:  F432
                        .NREL
F431:   STA     3,RF43
        LDA     1,SH3
        LDA     2,SI3
        JMP     F430

F432:   STA     3,RF43
        LDA     1, SH3
        LDA     3,IH3
        ADDZ    3,1
        LDA     2, SI3
        LDA     0,II3
        ADDZ    0,2

F430    STA     1, SH3
        STA     2,.SI3
        STA     1,.H5
        STA     2,.I5

        LDA     1,SAI
        STA     1,20
        STA     1,21
        LDA     1,CAI
        STA     1,22
        STA     1,23

        LDA     1,CTI
        STA     1,.C5

        JMP     @RF43
.SH3:   0
 SI3    0
RF43:   0
.END
```

A4.7 The Table Look Up and Accumulate Routine TLUAR

The subprogram is called by the SAG subprogram to process the sine and cosine K_{nk} values. After the TLUAR has performed its tasks, detailed in A3.10, control is returned to the SAG subprogram. The TLUAR consists of the following subroutines.

1) CINT/SINT
2) GY1/GY2
3) PACC/NACC

as well as the main routine.

In A3.10.2 the STV address is simply calculated by adding the two components of the address pair I_n and K_{nk} and the address of the first location in the STV. Subroutine SINT performs the first part of this addition by adding the address of the first location in the STV to the sine K_{nk} (argument) value. CINT adds the cosine K_{nk} to the address of the first location in the STV. This partial sum is termed the base address (BTLU). GY1/GY2 adds the LADC output to the base address to form the final STV address (see A3.10.4). PACC/NACC adds/subtracts the STV constant to/from the required Accumulator.

The main routine consists of six parts, each part corresponding to entry points described in A4.6. (ie part is associated with a sign pattern in Fig.A3.5).

The main routine performs the following tasks.

1) Store the return address of the SAG.
2) Calls SINT

3) Stores the contents of the particular Accumulator being processed in a temporary location.

4) Calls GY1/GY2 in the required order and increments or decrements the contents of the temporary location according to the value of the signum function

5) Returns the contents of the temporary location to the Accumulator.

These steps are then repeated for the cosine K_{nk} values.

A4.7.1 <u>Subroutine SINT</u>

```
SINT: STA   3,RINT       Save main routine return address
      STA   1,SC         Store sine Knk in scratch pad.
      STA   2,CC         Store cosine Knk in scratch pad
      LDA   3,SB6  )     Set auto inc. address for SB.
                   )
      STA   3,24   )
      LDA   3,TLU        Get address of first location of STV
      LDA   1,SC         Get sine Knk from scratch pad
      ADDZ  3,1          Add sine Knk address to first location
                         address of STV to obtain base address
      STA   1,BTLU       Store base address
      JMP   @RINT        Return to main routine
```

A4.7.2 <u>Subroutine CINT</u>

This is similar to SINT but sets up the base address for the cosine K_{nk}.

A4.7.3 Subroutine CINT2 and SINT2

These are similar to CINT and SINT and are used for the special case in P37.

A4.7.4 Subroutines GY1 and GY2

These subroutines generate the signum function and modify it for the 'opposite signs' case.

```
GY1:  STA    3,RGY        Save return to main routine
      STA    0,TAC        Store Accumulator contents in
                          temporary location
      LDA    3,@24        Get the next shuffled I_n value from
                          SB
      LDA    1,BTLU       Get base address of STV
      ADDZL  1,3,SNC      Calculate final STV address and
                          check sign bit
      JMP    PACC         If sign bit is positive go to
                          incrementing subroutine
      JMP    NACC         If sign bit is negative go to
                          decrementing routine
```

GY2 has the last two instructions swapped

```
ie.   ADDZL  1,2,SNC
      JMP    NACC         If sign bit is positive go to
                          decrementing routine
      JMP    PACC         If sign bit is negative go to
                          incrementing routine.
```

Register 3 contains the STV address and the signum function sign bit which have been shifted one place to the left so as to check the sign bit. PACC and NACC must first restore this address by shifting it one place to the right and destroying the sign bit as it has served its purpose.

A4.7.5 Subroutines PACC and NACC

These subroutines perform the following tasks

1) Restore the final STV address
2) Check the sign bits of the STV constant and Accumulator contents to check for overflow (OVF) underflow (UNF)
3) Add or subtract the STV constant to or from the Accumulator value
4) Write an error message if OVF/UNF occurs

```
PACC:  MOVZR  3,3      Restore STV address and eliminate the
                       sign bit.
       LDA    1,0,3    Get the STV constants and load them
                       into reg. 1.
       STA    1,T1D    Load STV constants into a temporary
                       location
       LDA    2,MSK    Get mask that will extract sign bits
                       of STV constant and Accumulator value
       LDA    0,TAC    Get Accumulator value which is stored
                       in a temporary location.
       AND    2,0      Get Accumulator sign bit
       AND    2,1      Get STV sign bit
```

```
        MOV     1,2        )
                           )
        ANDZL   0,2        )
                           )     Exclusive OR of sign bits
        ADD     0,1        )
                           )
        SUB     2,1,SZR    )

        JMP     PACN             sign bits are different
                                 There is no risk of OVF/UNF
        MOVZL#  0,0,SZC          The sign bits are the same.  Check
                                 the Accumulator sign bit
        JMP     PACU             Accumulator sign bit negative.
                                 UNF could take place.
        JMP     PACO             Accumulator sign bit positive OVF
                                 could take place
```

PACU is a subroutine that adds TID (STV constant) to TAC (Accumulator value) and checks for underflow. PACO is a subroutine adds TID to TAC and checks for overflow. PACN adds TID to TAC and does not check for overflow or underflow as neither will occur in this case.

```
PACN:   LDA     2,TID            Get STV constant
        LDA     0,TAC            Get Accumulator value
        ADDZ    2,0              Add
        JMP     @RGY             Return to main routine
PACO:   LDA     2,TID
        LDA     0,TAC
        ADDZL   2,0,SZC          Add, Shift left and check CARRY bit.
                                 If zero, no OVF and skip next
                                 instruction
```

```
        JMP     OVFP            OVF.  Go to error flag routine.
        MOVR    0,0             Restore result
        JMP     @RGY
PACU:   LDA     2,TID
        LDA     0,TAC
        ADDZL   2,0,SNC         Add.  Shift left and check CARRY
                                bit.  If one (negative result), no
                                UNF and skip next instruction
        JMP     UNFP            UNF.  Go to error flag routine
        MOVR    0,0             Restore result
        JMP     @RGY
```

The NACC subroutine is similar to the PACC subroutines but subtraction instead of addition takes place. The sign bits of the Accumulator and the STV constant are also exclusively ORed and the subroutine chooses one of three subtraction routines NACN (no OVF/UNF check necessary), NACO (check OVF) and NACU (check underflow) to decrement the Accumulator.

A4.7.6 <u>The Main Routine</u>

The main routine has to entry points P61, P62, P63, P64, P611 and P612, one for each of the 6 sign patterns in Fig.A3.5. Consider the routine for the set OTS/OH.

```
P61:    STA     3,RP6           Save the SAG return
        JSR     SINT            See A4.7.1
```

```
LDA    0,@20           Get the contents of the Accumulator
                       that has its address set in Q41 and
                       load them into reg 0.   The second
                       statement in GY1/GY2 stores the
                       contents of reg 0 in a temporary
                       location.
JSR    GY1  )
            )          This is equivalent to the sign pattern
JSR    GY1  )
            )          associated with the set OTS/OH.
JSR    GY2  )
            )
JSR    GY2  )
STA    0,@21           Reg. 0 contains the incremented or
                       decremented Accumulator value.   The
                       contents of reg. 0 are returned to
                       the Accumulator block.
```

The cosine main routine follows a similar pattern.

```
JSR    CINT
LOA    0,@22           Get Accumulator value
JSR    GY1  )
            )
JSR    GY2  )
            )          sign pattern for cosine OTS/OH.
JSR    GY2  )
            )
JSR    GY1  )
STA    0,@23           Return the incremented or decremented
                       Accumulator value to the Accumulator
                       block
JMP    @RP6            Return to SAG
```

```
                .TITL   PG1
                .ENT    SPG1 SPG2 SPG3 SPG4
                .ENT    SPG11 SPG12
                .EXTN   .TLU  SI
                .EXTN   .SPSI
                .EXTD   ?TLU
                .ZREL
                .LOC    150
SPG1:   PG1
SPG2:   PG2
SPG3:   PG3
SPG4:   PG4
SPG11:  PG11
SPG12:  PG12
TLU:    .TLU
SSG:    .SG
RPG:    0
                .NREL
PG1:    STA     3,RPG

        JSR     SINT

        LDA     0,@20
        JSR     GY1
        JSR     GY1
        JSR     GY2
        JSR     GY2
        STA     0,@21

        JSR     CINT

        LDA     0,@22
        JSR     GY1
        JSR     GY2
        JSR     GY2
        JSR     GY1
        STA     0,@23

        JMP     @RPG

PG11:   STA     3,RPG

        JSR     SINT

        LDA     0,@20
        JSR     GY1
        JSR     GY2
        STA     0,@21

        JSR     CINT

        LDA     0,@22
        JSR     GY1
        JSR     GY2
        STA     0,@23

        JMP     @RPG

PG12:   STA     3,RPG

        JSR     SINT2

        LDA     0,@20
        JSR     GY1
        JSR     GY2
        STA     0,@21

        JSR     CINT2

        LDA     0,@22
        JSR     GY1
        JSR     GY2
        STA     0,@23

        JMP     @RPG

PG2:    STA     3,RPG

        JSR     SINT

        LDA     0,@20
        JSR     GY1
        JSR     GY1
        JSR     GY2
        JSR     GY2
        JSR     GY1
        JSR     GY1
        JSR     GY2
        JSR     GY2
        STA     0,@21

        JSR     CINT

        LDA     0,@22
        JSR     GY1
        JSR     GY2
        JSR     GY2
        JSR     GY1
        JSR     GY1
        JSR     GY2
        JSR     GY2
        JSR     GY1
        STA     0,@23

        JMP     @RPG
```

```
P63:    STA     3,RP6

        JSR     SINT

        LDA     0,@20
        JSR     GY1
        JSR     GY2
        JSR     GY1
        JSR     GY2
        STA     0,@21

        JSR     CINT

        LDA     0,@22
        JSR     GY1
        JSR     GY2
        JSR     GY1
        JSR     GY2
        STA     0,@23

        JMP     @RP6

P64:    STA     3,RP6

        JSR     SINT

        LDA     0,@20
        JSR     GY1
        JSR     GY1
        JSR     GY1
        JSR     GY1
        STA     0,@21

        JSR     CINT

        LDA     0,@22
        JSR     GY1
        JSR     GY1
        JSR     GY1
        JSR     GY1
        STA     0,@23

        JMP     @RP6

GY1:    STA     3,RGY
        STA     0,TAC
        LDA     3,@24
        LDA     1,BTLU
        ADDZL   1,3,SNC
        JMP     FACC
        JMP     NACC

GY2     STA     3,RGY
        STA     0,TAC
        LDA     3,@24
        LDA     1,BTLU
        ADDZL   1,3,SNC
        JMP     NACC
        JMP     FACC

SINT:   STA     3,RINT
        STA     1,SC
        STA     2,CC
        LDA     3,SB6
        STA     3,24
        LDA     3,TLU
        LDA     1,SC
        ADDZ    3,1
        STA     1,BTLU
        JMP     @RINT

CINT:   STA     3,RINT
        LDA     3,SB6
        STA     3,24
        LDA     3,TLU
        LDA     1,CC
        ADDZ    3,1
        STA     1,BTLU
        JMP     @RINT

SINT2:  STA     3,RINT
        STA     1,SC
        STA     2,CC
        LDA     3,SB6
        INC     3,3
        STA     3,24
        LDA     3,TLU
        LDA     1,SC
        ADDZ    3,1
        STA     1,BTLU
        JMP     @RINT

CINT2.  STA     3,RINT
        LDA     3,SB6
        INC     3,3
        STA     3,24
        LDA     3,TLU
        LDA     1,CC
        ADDZ    3,1
        STA     1,BTLU
        JMP     @RINT

FACC:   MOVZR   3,3
        LDA     1,0,3
        STA     1,TID
        LDA     2,MSK
        LDA     0,TAC
        AND     2,0
        AND     2,1

        MOV     1,2
        ANDZL   0,2
        ADD     0,1
        SUB     2,1,SZR

        JMP     FACN
        MOVZL#  0,0,SZC
        JMP     FACU
        JMP     FACO
```

```
NACC.   MOVZR   3,3
        LDA     @1,0,3
        STA     1,TID
        LDA     2,MSK
        LDA     0,TAC
        AND     2,0
        AND     2,1

        MOV     1,2
        ANDZL   0,2
        ADD     0,1
        SUB     2,1,SNR

        JMP     NACN
        MOVZL#  0,0,SZC
        JMP     NACU
        JMP     NACO

FACN    LDA     2,TID
        LDA     0,TAC
        ADDZ    2,0
        JMP     @RGY

FACO:   LDA     2,TID
        LDA     0,TAC
        ADDZL   2,0,SZC
        JMP     OVFP
        MOVR    0,0
        JMP     @RGY

FACU    LDA     2,TID
        LDA     0,TAC
        ADDZL   2,0,SNC
        JMP     UNFP
        MOVR    0,0
        JMP     @RGY

NACN.   LDA     2,TID
        LDA     0,TAC
        SUBZ    2,0
        JMP     @RGY

NACO    LDA     2,TID
        LDA     0,TAC
        SUBZL   2,0,SZC
        JMP     OVFN
        MOVR    0,0
        JMP     @RGY

NACU    LDA     2,TID
        LDA     0,TAC
        SUBZL   2,0,SNC
        JMP     UNFN
        MOVR    0,0
        JMP     @RGY
```

```
SC:     0
CC:     0
TAC:    0
TID:    0
BTLU:   0
MSK:    100000
RGY.    0
RINT:   0
OVFP:   LDA     1,A1
        JSR     @BD
        JMP     OUA
OVFN:   LDA     1,A2
        JSR     @BD
        JMP     OUA
UNFP.   LDA     1,A3
        JSR     @BD
        JMP     OUA
UNFN.   LDA     1,A4
        JSR     @BD
        JMP     OUA
OUA:    LDA     1,20
        JSR     @BD
        LDA     1,21
        JSR     @BD
        LDA     1,22
        JSR     @BD
        LDA     1,23
        JSR     @BD
        JMP     @STLU
BD:     .EMBD
A1:     7
A2:     115
A3:     1411
A4.     17141
.END
```

A4.8 The CCL Subroutine

This subroutine calculates all the constants for the DFTTLU for a given value of N (ie. N = 128,256,512,1024) and will stop if N is not one of these values. The subprogram calculates the numbers of columns of K_{nk} values that fall into each of the four cases. These numbers are listed as C16, C161, C32, C321, C64 and C641. It then calculates the numbers of K_{nk} values in a column which are listed as CT1, CT2 and CT3. MB0, MB1 ... MB21 contain the first group of addresses of the shuffled time samples. These addresses are needed by the Shuffling Subprogram. The subprogram calculates the first set of starting and incrementing arguments required by the Q41 - Q49 subprograms ie SH1, S11, IH1, I11 ... II9.

```
              TITL    .COLD
              ENT     .MS  IS, CS TEI
              ENT     C16 C161 C32 C321 C64 C641
              ENT     CCAL IISH MEI NN.N MS
              ENT     SH1 SH2 SH3 IH1 IH2 IH3
              ENT     SI1 SI2 SI3 II1 II2 II3
              ENT     SH4, SH5 SH6 IH4, IH5, IH6
             .ENT     SI4 SI5 SI6 II4 II5 II6
             .ENT     SH7 SH8 SH9 IH7 IH8 IH9
             .ENT     SI7 SI8 SI9 II7 II8 II9
              ENT     ME0, ME1, ME2, ME3, ME4, ME5
             .ENT     ME6, ME7, ME8, ME9, ME10, ME11
             .ENT     ME12 ME13, ME14, ME15, ME16
             .ENT     ME17, ME18 ME19, ME20, ME21
              ENT     MSK1, MSK2, CT1, CT2, CT3, C??
              EXTN    .MS
             .ZREL
              LOC     42
CCAL.   .CCAL
IISH:   SH1            N       200
MS      .MS            MEI     ME0
SH1.    0              ME0     0
SH2     0              ME1     0
SH4     0              ME2     0
IH1.    0              ME3     0
SI1.    0              ME4     0
SH5.    0              ME5.    0
IH2.    0              ME6     0
IH4.    0              ME7     0
IH7:    0              ME8     0
SI2:    0              ME9     0
SI3     0              ME10    0
SI4.    0              ME11    0
II1:    0              ME12    0
IH5     0              ME13    0
IH8:    0              ME14    0
SI5:    0              ME15    0
SI6:    0              ME16    0
II2.    0              ME17    0
II3     0              ME18    0
II4.    0              ME19    0
II7     0              ME20    0
II5.    0              ME21.   0
II6.    0              CT1     0
II8     0              CT2     0
II9:    0              CT3     0
SH3     0              C16     0
SH6     0              C161    0
SH7     0              C32     0
SH8.    0              C321    0
SH9:    0              C64     0
IH3     0              C641    0
IH6     0              NN      0
IH9:    0              NF11    0
SI7:    0                                 .LOC    170
SI8.    0               MS     0
SI9     0               IS     0
MSK1.   170000          CS     0
MSK2    1700           TEI     10
CIP     0              ???     ???
```

```
                .NREL
.CCAL:  STA     3,RP11
        LDA     1,N

        LDA     2,M1
        ANDZ    1,2,SZR
        JMP     .+2
        JMP     A128

        JSR     @SR6
        LDA     2,M2
        ANDZ    1,2,SZR
        JMP     .+2
        JMP     A256

        JSR     @SR6
        LDA     2,M3
        ANDZ    1,2,SZR
        JMP     .+2
        JMP     A512

        JSR     @SR6
        LDA     2,M4
        ANDZ    1,2,SZR
        JMP     .+2
        JMP     A1024

        LDA     2,M5
        ANDZ    1,2,SZR
        HALT
        JMP     A2048

A128:   LDA     1,N4
        JMP     SRI

A256:   LDA     1,N5
        JMP     SRI

A512:   LDA     1,N6
        JMP     SRI

A1024:  LDA     1,N7
        JMP     SRI

A2048:  LDA     1,N8
        JMP     SRI

SRI:    STA     1,DNZ
        LDA     1,SIB
        MOVOR   1,1
.SRI:   MOVZR   1,1
        DSZ     DNZ
        JMP     .SRI
        STA     1,SH1

        LDA     2,IISH
        STA     2,20

        LDA     1,SH1
        MOVZL   1,1
        LDA     2,N4
        STA     2,.N4
A1:     STA     1,@20
        DSZ     .N4
        JMP     A1

        MOVZL   1,1
        LDA     2,N3
        STA     2,.N3
A2:     STA     1,@20
        DSZ     .N3
        JMP     A2

        MOVZL   1,1
        LDA     2,N3
        STA     2,.N3
A3:     STA     1,@20
        DSZ     .N3
        JMP     A3

        MOVZL   1,1
        LDA     2,N4
        STA     2,.N4
A4:     STA     1,@20
        DSZ     .N4
        JMP     A4

        LDA     1,N
        MOVZR   1,1
        MOVZR   1,1
        STA     1,CT1
        STA     1,ISF
        MOVZR   1,1
        STA     1,CT2
        STA     1,CT3
        MOVZR   1,1
        STA     1,C16
        STA     1,C161
        DSZ     C161
        MOVZR   1,1
        STA     1,C32
        STA     1,C321
        DSZ     C321
        MOVZR   1,1
        STA     1,C64
        STA     1,C641
        DSZ     C641

        LDA     1,CT1
        LDA     2,CT2
        LDA     3,CT3
        ADD     1,2
        ADD     2,3
        STA     3,N4

        LDA     2,MEI
        STA     2,21

        LDA     1,ME
        STA     1,MET

        LDA     1,N3
        STA     1,.N3

E:      LDA     1,MET
        INC     1,1
        STA     1,MET
        JSR     P110
        DSZ     .N3
        JMP     E

        LDA     1,ME
        LDA     0,ISF
        ADDZ    0,1
        STA     1,MST
        JSR     P110
        DSZ     MST
        JSR     P110

        LDA     1,MB15
        INC     1,1
        STA     1,MB21
        JMP     @RP11

P110:   STA     3,RP110
        LDA     2,N4
        STA     2,.N4
        LDA     1,MET
        LDA     0,ISF
        JMP     P111
P112:   ADDZ    0,1
P111:   STA     1,@20
        DSZ     .N4
        JMP     P112
        JMP     @RP110

M1:     177577
M2:     177377
M3:     176777
M4:     175777
M5:     173777
N3:     3
N4:     4
N5:     5
N6:     6
N7:     7
N8:     10
DNZ:    0
.N3:    0
.N4:    0
.N5:    0
ISF:    0
RP110:  0
MST:    0
SR6:    STA     3,RET
        LDA     3,TBI
        MOVZR   3,3
        STA     3,TBI
        JMP     @RET
RET:    0
        .END
```

A4.9 Calculation of the Resultant Harmonics

Fig.A3.17 shows the format of the Accumulator block in the core. This subprogram extracts the corresponding real and imaginary components in the correct sequences, calculates the resultant and divides it by 7 (see 4.4). Data General Software[4] is used to perform the multiplication, double word addition and division. Reference 18 gives a routine that computes the square root of a double word. The resulting harmonics are used by the error analysis described in A5. This subprogram is also used to compute the resultant harmonics of the FFT routine used in the error analysis.

Subroutine STR extracts the harmonic components in the correct order ie E4H, OH, $\bar{E4}$H,OH ... and subroutine PTG calculates the resultant, divides it by 7, and stores it in block RB. The resultant harmonics in RB are in the correct sequence and are readily available for analysis by the error program.

```
LDA   1,SA1
STA   1,20      The addresses of the first
LDA   1,CA1     locations in the
STA   1,21      Accumulator blocks are
LDA   1,SA2     loaded into the auto-
STA   1,22      incrementing address
LDA   1,CA2     locations
STA   1,23
LDA   1,SA3
STA   1,24
LDA   1,CA3
STA   1,25
LDA   1,RB
STA   1,26

LDA   1,NH      The index i  set up.
STA   1,NH      NH is the number of harmonics
                that will be processed.
                NH is decremented by one as each
                resultant harmonic is processed.
                When NH is zero control is returned
                to MCN.
```

```
        JSR     @CPB          This instruction is included if the
                              harmonics are being plotted.  CPB
                              positions the zero at
                              (X,Y) = (-1000_8, -1000_8) and makes a
                              small vertical mark.
STR:    LDA     1,@24         Get imaginary E4 component
        LDA     2,@25         Get real E4 component
        JSR     PTG           Go to resultant calculating routine

        LDA     1,@20         Get imaginary odd numbered component
        LDA     2,@21         Get real odd numbered component
        JSR     PTG
        LDA     1,@22         Get imaginary E4̄ component.
        LDA     2,@23         Get real E4̄ component
        JSR     PTG
        LDA     1,@20         Get imaginary odd-numbered component
        LDA     2,@21         Get real odd numbered component
        JSR     PTG
        JMP     STR
```

NH is decremented and tested for zero in PTG. The resultant is calculated using the Pythagorus Theorem. The squaring of the component is carried out by subroutine MPY, the addition by DADD and the square root, DSQR. The result of MPY is a double word, the high order bits are stored in reg. 0 and the low order in reg. 1.

```
PTG:  STA   3,RPTG       Save return to STR.
      STA   2,TC         store the real component in a
                         temporary location.
      MOV   1,2          Duplicate the imaginary component.
                         in reg. 2
      JSR   @MPY         Square imaginary component
      STA   0,HS         Store high order bits in HS.
      STA   1,LS         Store low order bits in LS.
      LDA   1,TC         Get real component from temporary
                         location.
      LDA   2,TC         Luplicate real component in reg 2.
      JSR   @MPY         Square real component.
      JSR   @DADD        Add squared real and imaginary
                         component.
AHS:  HS                 This statement gives the location of
                         the squared imaginary components to
                         DADD
      LDA   3,CKK        The double word sum must be less than
      AND   0,3,SZR      2^29.  If greater than 2^29, the square
      JMP   OSQR.        root of the double word sum will cause
                         a 2's complement 16 bit word to
                         overflow.  If the double word is
                         greater than 2^29, control is given to
                         to subroutine OSQR which prints an
                         error message '66' and then gives
                         control to subroutine STLU. (A4.12)
```

```
JSR   @DSQR          Take the square root of the double
                     word sum.
LDA   2,C17          Load 7 into divisor register
JSR   @DIVI          Divide square root result by 7.
STA   1,@26          Store the answer in RB.
```

If the result is to be plotted, the following 2 instructions are included.

```
JSR   @SRTB  )       See A4.13
             )
JSR   @DOR   )
DSZ   .NH            Decrement NH by one and test for zero.
JMP   @RPTG          NH not zero.  Return to STR for next
                     harmonic.
JSR   @CPE           NH is zero.  This statement is
                     included if the results are plotted.
                     The pen in positions at (X,Y) =
                     (0, -1000₈) and makes a small vertical
                     mark.
JMP   @RP75          Return to MCN.
```

```
                .TITL   .PP751
                .ENT    SP75
                .EXTN   .DADD,.MPY,.DSQR,.SDEC,.DIVI
                .EXTD   CA3,CA2 DOR,SP[illegible]
                .EXTD   SA1,SA2 SA3
                .EXTD   CA1,CA2,CA3
                .EXTD   NH,TBI,STLU,FE
                .ZREL
                .LOC    163
SP75:   P75
                .NREL
P75:    STA     3,RP75

        LDA     1,SA1
        STA     1,20
        LDA     1,CA1
        STA     1,21
        LDA     1,SA2
        STA     1,22
        LDA     1,CA2
        STA     1,23
        LDA     1,SA3
        STA     1,24
        LDA     1,CA3
        STA     1,25
        LDA     1,FE
        STA     1,26

        LDA     1,NH
        STA     1,.NH
        JSR     @CFE

STR:    LDA     1,@24
        LDA     2,@23
        JSR     PTG

        LDA     1,@20
        LDA     2,@21
        JSR     PTG

        LDA     1,@22
        LDA     2,@23
        JSR     PTG

        LDA     1,@20
        LDA     2,@23
        JSR     PTG

        JMP     STR
```

```
PTG:    STA     3,RPTG
        STA     2,TC

        MOV     1,3
        JSR     @MPY
        STA     0,HS
        STA     1,LS

        LDA     1,TC
        LDA     2,TC
        JSR     @MPY

        JSR     @DADD
AHS.    HS

        LDA     3,CK[illegible]
        AND     0,3,SZR
        JMP     DSOF
        JSR     @DSQR
        LDA     2,CK[illegible]
        JSR     @DIVI
        LDA     3,C[illegible]
        AND     1,3,SZR
        JMP     DSOF
        STA     1,@25
        JSR     @SPTB
        JSR     @DOR
        DSZ     .NH
        JMP     @RPTG
        JSR     @CA2
        JMP     @RP75

DSOF    LDA     1,EL
        JSR     @ED
        JMP     @STL

RP75:   0
HS:     0
LS.     0
TC.     0
RPTG.   0
.NH:    0
MPY:    .MPY
DADD:   .DADD
DSQR:   .DSQR
DIVI:   .DIVI
C17.    7
CK[illegible].  1[illegible]
CKM1.   [illegible]
EL:     102
ED      .SDEC
.END
```

```
                .TITL   P7521
                .ENT    SP75
                .EXTN   DADD, MPY, DSQR, EDEC, DIVI
                .EXTD   SA1, SA2, SA3
                .EXTD   CA1, CA2, CA3
                .EXTD   NH, TBI, ETLU, R6
                .ZREL
                .LOC    255
SP75:   P75
                .NREL
P75:    STA     3, RP75

        LDA     1, SA1
        STA     1, 20
        LDA     1, CA1
        STA     1, 21
        LDA     1, SA2
        STA     1, 22
        LDA     1, CA2
        STA     1, 23
        LDA     1, SA3
        STA     1, 24
        LDA     1, CA3
        STA     1, 25
        LDA     1, R6
        STA     1, 26

        LDA     1, NH
        STA     1, .NH

STR:    LDA     1, @24
        LDA     2, @25
        JSR     PTG

        LDA     1, @20
        LDA     2, @21
        JSR     PTG

        LDA     1, @22
        LDA     2, @23
        JSR     PTG

        LDA     1, @20
        LDA     2, @21
        JSR     PTG

        JMP     STR

PTG:    STA     3, RPTG
        STA     2, TC

        MOV     1, 2
        JSR     @MPY
        STA     0, HS
        STA     1, LS

        LDA     1, TC
        LDA     2, TC
        JSR     @MPY

        JSR     @DADD
AHS:    HS

        LDA     3, CHK
        AND     0, 3, SZR
        JMP     DSQR
        JSR     @DSQR
        LDA     2, C17
        JSR     @DIVI
        LDA     3, CHK1
        AND     1, 3, SZR
        JMP     OSQR
        STA     1, @26
        DSZ     .NH
        JMP     @RPTG
        JMP     @RP75

OSQR:   LDA     1, E5
        JSR     @ED
        JMP     @ETLU

RP75:   0
HS:     0
LS:     0
TC:     0
RPTG:   0
.NH:    0
MPY:    .MPY
DADD:   .DADD
DSQR:   .DSQR
DIVI:   .DIVI
C17:    7
CHK:    160000
CHK1:   170000
E5:     103
ED:     .EDEC
        .END
```

A4.10 The Data Blocks

The data blocks are required to store the large number of values in consecutive locations. These values can be easily addressed using the auto incrementing address facility. To save core, certain aress of core are overwritten with new data, eg. the I1 block associated with the FFT is also used for the DFTTLU accumulators.

MB: N time samples - output of LADC.

R1: N time samples - output of ADC.

I1: N imaginary time samples - set to zero.

The results of the FFT are loaded in R1 and I1.

R1: N Real FFT harmonic components

I1: N Imaginary FFT harmonic components.

SA1: $\frac{N}{4}$ Imaginary FFT harmonic components

SA1: $\frac{N}{4}$ Imaginary odd numbered DFTTLU harmonic components.

CA1: $\frac{N}{4}$ Real odd numbered DFTTLU harmonic components.

SA2: $\frac{N}{8}$ Imaginary even numbered (not multiples of 4) DFTTLU harmonic components.

CA2: $\frac{N}{8}$ Real even numbered (not multiples of 4) DFTTLU harmonic components.

SA3: $\frac{N}{8}$ Imaginary remaining even numbered DFTTLU harmonic components.

CA3: $\frac{N}{8}$ Real remaining even numbered DFTTLU harmonic components.

RB: $\frac{N}{2}$ Resultant DFTTLU harmonics.

SB: Current shuffled I_n values.

A4.11 The Main Control Program MCN

The main control program calls the subprograms in the required order. The program first resets the Accumulators by loading zero into the Accumulator. It then calls the sampling subprogram.

```
        JSR   @SP21            Zero Accumulators
        JSR   @SP25            Sample outputs of LADC and (ADC)
```

The first column in each of the nine sets of columns is solved separately as some of the subprograms are entered at a special point (eg. SP3 X 1, SP4 X 1). The subprogram associated with the OTS/OH is explained.

```
Q1.1:   JSR   @SP311           Shuffle time samples.
        JSR   @SP411           Get starting and incrementing
                               arguments.
        JSR   @SA61            Generate first columnof K_nk values.
        LDA   1,C161  )        Set up the index for the remaining
                      )
        STA   1,CCNT  )        columns in case 1 (0 < inc < π/2)
Q1.2:   JSR   @SP312           Get next group of shuffled time
                               samples
        JSR   @SP412           Get next starting and incrementing
                               arguments
        JSR   @SA61            Generate next column of K_nk addresses
        DSZ   CCNT    )        Repeat for all the columns in case 1
                      )
        JMP   Q1.2    )
```

The following 3 cases are treated similarly and the pattern is repeated for the remaining eight sets of columns.

```
         TITL    MCN4
         .ENT    MCON Q1.1
         EXTD    SP21, SP25
         .EXTD   SPI11, SPI12, SPI21, SPI22, SPI31, SPI32
         EXTD    SPI41, SPI42, SPI51, SPI52, SPI61, SPI62
         .EXTD   [illegible]
         EXTD    SP411, SP412, SP421, SP422, SP431, SP432
         EXTD    [illegible]
         EXTD    SP471, SP472, SP481, SP482, SP491, SP492
         .EXTD   [illegible]
         EXTD    [illegible]
         .EXTD   [illegible]
         EXTD    [illegible]
         .EXTD   [illegible]
         EXTD    [illegible]
         .EXTD   [illegible]
         EXTD    [illegible]
         EXTD    C16, C161, C32, C321, C64, C641
         .ZREL
         LOC     160
MCON:    .MCON
CCNT:    0
RCON:    0
         NREL

.MCON:   STA     3,RCON

         JSR     @SP25

Q1.1     JSR     @SP21
         JSR     @SPI11
         JSR     @SP411
         JSR     @SA61

         LDA     1,C161
         STA     1,CCNT
Q1.2:    JSR     @SPI12
         JSR     @SP412
         JSR     @SA61
         DSZ     CCNT
         JMP     Q1.2

         LDA     1,C16
         STA     1,CCNT
Q1.3:    JSR     @SPI12
         JSR     @SP412
         JSR     @SE61
         DSZ     CCNT
         JMP     Q1.3

Q2.1     JSR     @SPI21
         JSR     @SP421
         JSR     @SA62

         LDA     1,C321
         STA     1,CCNT
Q2.2:    JSR     @SPI22
         JSR     @SP422
         JSR     @SA62
         DSZ     CCNT
         JMP     Q2.2

         LDA     1,C32
         STA     1,CCNT
Q2.3     JSR     @SPI22
         JSR     @SP422
         JSR     @SE62
         DSZ     CCNT
         JMP     Q2.3

Q3.1.    JSR     @SPI31
         JSR     @SP431
         JSR     @SC64

         LDA     1,C321
         STA     1,CCNT
Q3.2:    JSR     @SPI32
         JSR     @SP432
         JSR     @SC64
         DSZ     CCNT
         JMP     Q3.2

         LDA     1,C32
         STA     1,CCNT
Q3.3     JSR     @SPI32
         JSR     @SP432
         JSR     @SD64
         DSZ     CCNT
         JMP     Q3.3

         LDA     1,C32
         STA     1 CCNT
Q3.4     JSR     @SPI32
         JSR     @SP432
         JSR     @SE64
         DSZ     CCNT
         JMP     Q3.4

         LDA     1,C32
         STA     1,CCNT
Q3.5     JSR     @SPI32
         JSR     @SP432
         JSR     @SF64
         DSZ     CCNT
         JMP     Q3.5

Q4.1     JSR     @SPI41
         JSR     @SP441
         JSR     @SA61

         LDA     1,C321
         STA     1,CCNT
Q4.2.    JSR.    @SPI42
         JSR     @SP442
         JSR     @SA61
         DSZ     CCNT
         JMP     Q4.2
```

```
        LDA     1,C32
        STA     1,CONT
04.3    JSR     @SFPI43
        JSR     @SFP443
        JSR     @SDC61
        DSZ     CONT
        JMP     04.3

05.1:   JSR     @SFPI51
        JSR     @SFP451
        JSR     @SAC62

        LDA     1,C641
        STA     1,CONT
05.2:   JSR     @SFPI52
        JSR     @SFP452
        JSR     @SBA62
        DSZ     CONT
        JMP     05.2

        LDA     1,C64
        STA     1,CONT
05.3.   JSR     @SFPI53
        JSR     @SFP453
        JSR     @SEC62
        DSZ     CONT
        JMP     05.3

06.1.   JSR     @SFPI61
        JSR     @SFP461
        JSR     @SCC64

        LDA     1,C641
        STA     1,CONT
06.2:   JSR     @SFPI62
        JSR     @SFP462
        JSR     @SCC64
        DSZ     CONT
        JMP     06.2

        LDA     1,C64
        STA     1,CONT
06.3:   JSR     @SFPI63
        JSR     @SFP463
        JSR     @SDC64
        DSZ     CONT
        JMP     06.3

        LDA     1,C64
        S       1,CONT
06.4.   JSR     @SFPI64
        JSR     @SFP464
        JSR     @SEC64
        DSZ     CONT
        JMP     06.4

        LDA     1,C64
        STA     1,CONT
06.5    JSR     @SFPI65
        JSR     @SFP465
        JSR     @SFC64
        DSZ     CONT
        JMP     06.5

07.1.   JSR     @SFPI71
        JSR     @SFP471
        JSR     @SIC11

        LDA     1,C221
        STA     1,CONT
07.2.   JSR     @SFPI72
        JSR     @SFP472
        JSR     @SAC21
        DSZ     CONT
        JMP     07.2

07.3    JSR     @SFPI73
        JSR     @SFP473
        JSR     @SJC21

        LDA     1,C221
        STA     1,CONT
07.4    JSR     @SFPI74
        JSR     @SFP474
        JSR     @SEC21
        DSZ     CONT
        JMP     07.4

07.5.   JSR     @SFPI75
        JSR     @SFP475
        JSR     @SMC12

08.1.   JSR     @SFPI81
        JSR     @SFP481
        JSR     @SIC21

        LDA     1,C641
        STA     1,CONT
08.2    JSR     @SFPI82
        JSR     @SFP482
        JSR     @SAC21
        DSZ     CONT
        JMP     08.2

08.3    JSR     @SFPI83
        JSR     @SFP483
        JSR     @SJC21

        LDA     1,C641
        STA     1,CONT
08.4    JSR     @SFPI84
        JSR     @SFP484
        JSR     @SEC21
        DSZ     CONT
        JMP     08.4

08.5.   JSR     @SFPI85
        JSR     @SFP485
        JSR     @SMC21

        LDA     1,C641
        STA     1,CONT
08.6    JSR     @SFPI86
        JSR     @SFP486
        JSR     @SGC21
        DSZ     CONT
        JMP     08.6

08.7.   JSR     @SFPI87
        JSR     @SFP487
        JSR     @SLC21

        LDA     1,C641
        STA     1,CONT
08.8    JSR     @SFPI88
        JSR     @SFP488
        JSR     @SHC21
        DSZ     CONT
        JMP     08.8

09.1:   JSR     @SFPI91
        JSR     @SFP491
        JSR     @SIC64

        LDA     1,C641
        STA     1,CONT
09.2    JSR     @SFPI92
        JSR     @SFP492
        JSR     @SIC64
        DSZ     CONT
        JMP     09.2

09.3:   JSR     @SFPI93
        JSR     @SFP493
        JSR     @SMC64

        LDA     1,C641
        STA     1,CONT
09.4:   JSR     @SFPI94
        JSR     @SFP494
        JSR     @SDC64
        DSZ     CONT
        JMP     09.4

09.5:   JSR     @SFPI95
        JSR     @SFP495
        JSR     @SNC64

        LDA     1,C641
        STA     1,CONT
09.6.   JSR     @SFPI96
        JSR     @SFP496
        JSR     @SEC64
        DSZ     CONT
        JMP     09.6

09.7:   JSR     @SFPI97
        JSR     @SFP497
        JSR     @SOC64

        LDA     1,C641
        STA     1,CONT
09.8:   JSR     @SFPI98
        JSR     @SFP498
        JSR     @SFC64
        DSZ     CONT
        JMP     09.8

        JSR     @SFP75

        JMP     @RCON
END
```

A4.12 The Run Program

This program links the DFTTLU and the FFT programs so that an analysis of the differences between the two methods can be made. The Run program performs the following tasks.

1) Sets the number of bits representing a constant in the STV (see 4.4).

2) Runs the DFTTLU by calling MCN.

3) Runs the FFT routine.

4) Adjusts the scale of the FFT harmonics to that of the DFTTLU.

5) Analyses the differences between the two sets of harmonics.

6) Repeats the above steps a specified number of times.

A4.12.1 Setting the STV

The STV is programmed with the maximum number of bits representing a constant. The number of bits is reduced by shifting each constant one place to the right (ie. divide by 2). The number of times that constant is divided by 2 is set on the console switches of the computer and is loaded into location SR 5. The computer duplicates this value by writing it into RS and is printed at the end of each run.

Subroutine STLU performs the same task when overflow or underflow of the Accumulators occurs or when the resultant harmonics in PTG(A4.9) overflows. In these cases, control is passed to STLU which halves each constant in the STV and notes this in location RS by increasing its value by 1. Control is passed to Q1.1 in MCN and the time samples are reprocessed.

A4.12.2 Calling the DFTTLU

After the STV has been set the following instructions are executed.

```
        JSR   @CCAL           Call the CCL subprogram (A4.8).
        JSR   @MCON           Call the MCN subprogram (A4.11)
        LDA   1,RS  )         Print RS
                    )
        JSR   @BD   )
        HALT                  Stops minicomputer
        JMP   @S7             When the 'CONTINUE' switch on the
                              console is pressed this instruction
                              is executed.
                              S7 is the start of the FFT program.
```

A4.12.3 Calling the FFT

The following Assembler Code is used.

```
.S7:    LDA   2,N             These instructions load zero into
        STA   2,.N            the I1 input block of the FFT.
        LDA   2,I1
        STA   2,20
        LDA   1,Z
S4:     STA   1,@20
        DSZ   .N
        JMP   S4
```

```
        LDA   2,N            N is passed to the FFT
        STA   2,.NN          program
        JSR   @CFTST         Call the FFT program
A:      0                    The FFT program reads A.  If A is
                             zero a forward FFT is executed.
.NN:                         The FFT program reads N here.
.R1:    .BLK                 N location input block (Real values)
.I1:    .BLK                 N location input block (Imaginary
                             values).  This block has been set
                             to zero.
```

The .I1 block is also used as the Accumulator block. The total number of locations in all the Accumulator blocks is also N. The addresses of the blocks are listed as follows.

```
        R1:   .R1 - 1
        I1:   .I1 - 1
        SA1:  .SA1 - 1
        etc.
```

The -1 term is included to adjust for the auto-incrementing address. This first increments the address and then executes it.

A4.12.4 Adjusting the Scale of the FFT Harmonics

The real and imaginary harmonic components of the FFT are loaded into .R1 and .I1 respectively. Subroutine PTG, (similar to the subroutine PTG in A4.9) calculates the resultant harmonics. The scale of the harmonics is adjusted while the harmonic is in its squared form and the harmonic is shifted double the required number of times to the right. The following instructions are included in PTG.

```
     JSR   @DADD
AHS: HS
```

The resultant harmonic has its high order bits in reg 0 and its low order bits in reg.1.

```
     LDA   3,SRS.        Get the doubled number of shifts and
                         store them in
     STA   3,.SRS        .SRS.
S1:  MOVZR 0,0           These instructions perform the
     MOVR  1,1           double word shift.
     DSZ   .SRS
     JMP   S1
     JMP   @DSQR         Take the square root.
```

A4.12.5 Calculating the FFT Scale Factors

The FFT has its own scale factor SCNT which depends on the value of the partial sums in the FFT. To avoid overflow every partial sum is halved if the machine code is greater than 31463_8 and the number of times that this occurs is stored in SCNT. Assuming that no overflow occurred, the FFT harmonics would still have to be scaled down by a constant MAN. If overflow does occur the final scale factor is (MAN - SCNT). This value is doubled and stored in location SRS (A4.12.4). If subroutine STLU is called (A4.12.1), (MAN 'T) is increased by one to keep the scales of the DFTTLU and FFT equal.

```
ASCNT:SCNT                        ASCNT contains the address of SCNT.

      LDA   1,@ASCNT  )
                      )           Print the contents of SCNT
      JSR   @BD       )

      LDA   1,@ASCNT

      LDA   2,MAN

      SUBO  1,2                   Calculate MAN - SCNT

      MOVZL  2,2                  Double the result

      STA   2,SRS                 Store this value - the number of
                                  shifts to the right - in SRS

      MOV   2,1       )           Print SRS
                      )
      JSR   @BD       )
```

A4.12.6 Plotting the Differences between the DFTTLU and FFT Harmonics

The format of the plot is given in Fig.5.1 in chapter 5.3.1. A subroutine similar to STR in A4.9 is used to retrieve the FFT harmonics components from R1 and I1 and the resultant harmonic is calculated and scaled. The corresponding DFTTLU harmonic is retrieved from block RB and is subtracted from the FFT harmonic. The result is multiplied by four and plotted with its DFTTLU harmonic which has been halved. Thus the scale of the difference between the harmonics is eight times that of the DFTTLU harmonics.

```
      LDA   0,R1                  Set auto inc. address for
      STA   0,20                  real (R1) and imaginary (I1)
      LDA   0,I1                  FFT harmonic components
      STA   0,21
```

```
        LDA   0,RB              Set auto inc. address for
        STA   0,22              resultant DFTTLU harmonics.
        LDA   0,N       )       Set index for N/2 harmonics
                        )
        MOVZR 0,0       )
                        )
        STA   0,.NH     )
        JSR   @CPB              Position plotter pen at
                                (X,Y) = (-512, -512)10
STR:    LDA   1,@20             Get real component
        LDA   2,@21             Get imaginary component
        JSR   PTG               Calculate and scale resultant
        LDA   2,@22             Get corresponding DFTTLU
                                resultant harmonic.
        SUBO  2,1               Subtract DFTTLU from FFT harmonic.
        MOVZL 1,1       )
                        )       Multiply difference by 4.
        MOVZL 1,1       )
        JSR   @SRTC     )
                        )       Plot difference (A4.13).
        JSR   @DOR      )
        DSZ   .NH               Decrement index.
        JMP   STR               If index is greater than N/2 process
                                next harmonic.
        JSR   @CPE              All harmonics plotted.  Mark end of
                                plot with pen.
        HALT                    End of test.
```

```
                        .TITL   .B128
                        .ENT    STLU, RB
                        .ENT    R1, SA1, CA1, SA2, CA2, SA3, CA3
                        .EXTD   CPB, CPE, DOR, SPTB, SRTC
                        .EXTD   CFTST, MCON, CCAL, N
                        .EXTN   .DADD, .MPY, .DSOR, .BDEC, SCNT, .TLUO
                        .EXTN   Q1.1
                        .ZREL
                        .LOC    250
STLU:     .STLU
XR:       Q1.1
BD:       .BDEC
ASCNT:    SCNT
R1:       .R1-1
I1:       .SA1-1
.N:       0
RB:       .RB-1
S7:       .S7
                        .LOC    371
SA1:      .SA1-1
SA2:      .SA2-1
SA3:      .SA3-1
CA1:      .CA1-1
CA2:      .CA2-1
CA3:      .CA3-1

                        .NREL
          LDA           1, S5
          STA           1, .S5
B2:       LDA           1, ATLU
          STA           1, 20
          STA           1, 21
          LDA           1, NTLU
          STA           1, .NTLU
B3:       LDA           1, @20
          MOVZR         1, 1
          STA           1, @21
          DSZ           .NTLU
          JMP           B3
          LDA           1, RS
          INC           1, 1
          STA           1, RS
          DSZ           .S5
          JMP           B2
          JSR           @CCAL
          JSR           @MCON
          LDA           1, RS
          JSR           @BD
         .HALT
          JMP           @S7

.STLU:    LDA           1, ATLU
          STA           1, 20
          STA           1, 21
          LDA           1, NTLU
          STA           1, .NTLU
B1:       LDA           1, @20
          MOVZR         1, 1
          STA           1, @21
          DSZ           .NTLU
          JMP           B1
          LDA           1, RS
          INC           1, 1
          STA           1, RS
          JMP           @XR
```

```
ATLU:    .TLUD
NTLU:    764
.NTLU:   0
RS:      0
S5:      4
.S5:     0
Z:       0
.RB:     .BLK     100

.S7:     LDA      2,N
         STA      2,.N
         LDA      2,I1
         STA      2,20
         LDA      1,Z
S4.      STA      1,@20
         DSZ      .N
         JMP      S4

         LDA      2,N
         STA      2,.NN
         JSR      @CFTST
A:       0
.NN:     0
.R1:     .BLK     200
.SA1:    .BLK     40
.CA1:    .BLK     40
.SA2:    .BLK     20
.CA2:    .BLK     20
.SA3:    .BLK     20
.CA3:    .BLK     20

         LDA      1,@ASCNT
         JSR      @BD
         LDA      1,@ASCNT
         LDA      2,MAN
         SUBO     1,2
         MOVZL    2,2
         STA      2,SRS
         MOV      2,1
         JSR      @BD
         LDA      0,R1
         STA      0,20
         LDA      0,I1
         STA      0,21
         LDA      0,RB
         STA      0,22
         LDA      0,N
         MOVZR    0,0
         STA      0,.NH
         JSR      @CPB

STR:     LDA      1,@20
         LDA      2,@21
         JSR      PTG
         LDA      2,@22
         SUBO     2,1
         MOVZL    1,1
         MOVZL    1,1
         JSR      @SPTC
         JSR      @DOR
         DSZ      .NH
         JMP      STR
         JSR      @CPE

XX:      HALT
PTG:     STA      3,RPTG
         STA      2,TC

         MOV      1,2
         JSR      @MPY
         STA      0,HS
         STA      1,LS

         LDA      1,TC
         LDA      2,TC
         JSR      @MPY

         JSR      @DADD
AHS:     HS
         LDA      3,SRS
         STA      3,.SRS
S1:      MOVZR    0,0
         MOVR     1,1
         DSZ      .SRS
         JMP      S1

         JSR      @DSQR
         JMP      @RPTG

TC:      0
HS:      0
LS:      0
RPTG:    0
.NH:     0
MPY:     .MPY
DADD:    .DADD
SRS:     0
.SRS:    0
MAN:     4
DSQR:    .DSQR
.END
```

```
                        .TITL   .B256
                        .ENT    STLU,PB
                        .ENT    R1,SA1,CA1,SA2,CA2,SA3,CA3
                        .EXTD   CPB,CPE,DQR,SRTB,SRTC
                        .EXTD   CFTST,MCON,CCAL,N
                        .EXTN   .DADD,.MPY,.DSQR,.BDEC,SCNT,.TLUD
                        .EXTN   Q1.1
                        .ZREL
                        .LOC    250
STLU:     .STLU
XR:       Q1.1
BD:       .BDEC
ASCNT:    SCNT
R1:       .R1-1
I1:       .SA1-1
.N:       0
RB:       .RB-1
S7:       .S7
                        .LOC    371
SA1:      .SA1-1
SA2:      .SA2-1
SA3:      .SA3-1
CA1:      .CA1-1
CA2:      .CA2-1
CA3:      .CA3-1

                        .NREL
          LDA           1,S5
          STA           1,.S5
B2:       LDA           1,ATLU
          STA           1,20
          STA           1,21
          LDA           1,NTLU
          STA           1,.NTLU
B3:       LDA           1,@20
          MOVZR         1,1
          STA           1,@21
          DSZ           .NTLU
          JMP           B3
          LDA           1,RS
          INC           1,1
          STA           1,RS
          DSZ           .S5
          JMP           B2
          JSR           @CCAL
          JSR           @MCON
          LDA           1,RS
          JSR           @BD
          HALT
          JMP           @S7

.STLU:    LDA           1,ATLU
          STA           1,20
          STA           1,21
          LDA           1,NTLU
          STA           1,.NTLU
B1:       LDA           1,@20
          MOVZR         1,1
          STA           1,@21
          DSZ           .NTLU
          JMP           B1
          LDA           1,RS
          INC           1,1
          STA           1,RS
          JMP           @XR
```

```
ATLU:    .TLUD
NTLU:    764
.NTLU:   0
RS:      0
S5:      5
.S5:     0
.KW:     .BLK     200
Z:       0

.S7:     LDA      2,N
         STA      2,.N
         LDA      2,I1
         STA      2,20
         LDA      1,Z
S4:      STA      1,@20
         DSZ      .N
         JMP      S4

         LDA      2,N
         STA      2,.NN
         JSR      @CFTST
A:       0
.NN:     0
.R1:     .BLK     400
.SA1:    .BLK     100
.CA1:    .BLK     100
.SA2:    .BLK     40
.CA2:    .BLK     40
.SA3:    .BLK     40
.CA3:    .BLK     40

         LDA      1,@ASCNT
         JSR      @ED
         LDA      1,@ASCNT
         LDA      2,MAN
        .SUBO     1,2
         MOVZL    2,2
         STA      2,SRS
         MOV      2,1
         JSR      @BD
         LDA      0,R1
         STA      0,20
         LDA      0,I1
         STA      0,21
         LDA      0,FB
         STA      0,22
         LDA      0,N
         MOVZR    0,0
         STA      0,.NH
         JSR      @CPB

STR:     LDA      1,@20
         LDA      2,@21
         JSR      PTG
         LDA      2,@22
         SUBO     2,1
         MOVZL    1,1
         MOVZL    1,1
         JSR      @SRTC
         JSR      @DOR
         DSZ      .NH
         JMP      STR
         JSR      @CPE
XX:      HALT
PTG:     STA      3,RPTG
         STA      2,TC

         MOV      1,2
         JSR      @MPY
         STA      0,HS
         STA      1,LS

         LDA      1,TC
         LDA      2,TC
         JSR      @MPY

         JSR      @DADD
AHS:     HS
         LDA      3,SRS
         STA      3,.SRS
S1:      MOVZR    0,0
         MOVR     1,1
         DSZ      .SRS
         JMP      S1

         JSR      @DSQR
         JMP      @RPTG

TC:      0
HS:      0
LS:      0
RPTG     0
.NH:     0
MPY:     .MPY
DADD:    .DADD
SRS:     0
.SRS:    0
MAN:     5
DSQR:    .DSQR
.END
```

```
                         .TITL    B512
                         .ENT     STLU, RB
                         .ENT     R1, SA1, CA1, SA2, CA2, SA3, CA3
                         .EXTD    CPB, CPE, DOR, SRTB, SRTC
                         .EXTD    CFTST, MCON, CCAL, N
                         .EXTN    .DADD, .MPY, .DSQR, .BDEC, SCNT, .TLUD
                         .EXTN    @..1
                         .ZREL
                         .LOC     250
STLU:      .STLU
XR:        @1, 1
BD:        .BDEC
ASCNT:     SCNT
R1:        .R1-1
I1:        .SA1-1
.N:        0
RB:        .RB-1
S7:        .S7
                         .LOC     371
SA1:       .SA1-1
SA2:       .SA2-1
SA3:       .SA3-1
CA1:       .CA1-1
CA2:       .CA2-1
CA3:       .CA3-1

                         .NREL
           LDA          1, S5
           STA          1, .S5
B2:        LDA          1, ATLU
           STA          1, 20
           STA          1, 21
           LDA          1, NTLU
           STA          1, .NTLU
B3:        LDA          1, @20
           MOVZR        1, 1
           STA          1, @21
           DSZ          .NTLU
           JMP          B3
           LDA          1, RS
           INC          1, 1
           STA          1, RS
           DSZ          .S5
           JMP          B2
           JSR          @CCAL
           JSR          @MCON
           LDA          1, RS
           JSR          @BD
           JMP          @S7

.STLU:     LDA          1, ATLU
           STA          1, 20
           STA          1, 21
           LDA          1, NTLU
           STA          1, .NTLU
B1:        LDA          1, @20
           MOVZR        1, 1
           STA          1, @21
           DSZ          .NTLU
           JMP          B1
           LDA          1, RS
           INC          1, 1
           STA          1, RS
           JMP          @XR
```

```
ATLU    .TLUD
NTLU:   764
.NTLU:  0
RS:     0
S5:     6
.S5:    0
.RB:    .BLK    400
Z:      0

.S7:    LDA     2,N
        STA     2,.N
        LDA     2,I1
        STA     2,20
        LDA     1,Z
S4:     STA     1,@20
        DSZ     .N
        JMP     S4

        LDA     2,N
        STA     2,.NN
        JSR     @CFTST
A:      0
.NN:    0
.R1:    .BLK    1000
.SA1:   .BLK    200
.CA1:   .BLK    200
.SA2:   .BLK    100
.CA2:   .BLK    100
.SA3:   .BLK    100
.CA3:   .BLK    100

        LDA     1,@ASCNT
        JSR     @BD
        LDA     1,@ASCNT
        LDA     2,MAN
        SUBO    1,2
        MOVZL   2,2
        STA     2,SRS
        MOV     2,1
        JSR     @BD
        LDA     0,R1
        STA     0,20
        LDA     [illegible]
        STA     [illegible]
        LDA     [illegible]
        STA     0,_2
S6:     LDA     0,N
        MOVZR   0,0
        MOVZR   0,0
        MOVZR   0,0
        STA     0,.NH
        JSR     @CPB

STR:    LDA     1,@22
        STA     1,TR
        JSR     @SRTB
        JSR     @DOR
```

```
        LDA     1,@20
        LDA     2,@21
        JSR     PTG
        LDA     2,TR
        SUBO    2,1
        MOVZL   1,1
        MOVZL   1,1
        JSR     @SRTC
        JSR     @DOR
        DSZ     .NH
        JMP     STR
        JSR     @CPE

XX:     HALT
        DSZ     .NQ
        JMP     S6
        HALT
        HALT
PTG:    STA     3,RPTG
        STA     2,TC

        MOV     1,2
        JSR     @MPY
        STA     0,HS
        STA     1,LS

        LDA     1,TC
        LDA     2,TC
        JSR     @MPY

        JSR     @DADD
AHS:    HS
        LDA     3,SRS
        STA     3,.SRS
S1:     MOVZR   0,0
        MOVR    1,1
        DSZ     .SRS
        JMP     S1

        JSR     @DSQR
        JMP     @RPTG

TC:     0
HS:     0
LS:     0
RPTG:   0
.NH:    0
MPY:    .MPY
DADD:   .DADD
SRS:    0
TR:     0
.NQ:    4
.SRS:   0
MAN:    6
DSQR:   .D[illegible]
.END
```

```
                        .TITL    B1024
                        .ENT     STLU,RB
                        .ENT     R1,SA1,CA1,SA2,CA2,SA3,CA3
                        .EXTD    CPB,CPE,DOR,SPTB,SPTC
                        .EXTD    CFTST,MCON,CCAL,N
                        .EXTN    .DADD,.MPY,.DSQR,.BDEC,SCNT,.TLUD
                        .EXTN    Q1.1
                        .ZREL
                        .LOC     250
STLU:      .STLU
XR:        Q1.1
BD:        .BDEC
ASCNT:     SCNT
R1:        .R1-1
I1:        .SA1-1
.N:        0
RB:        .RB-1
S7:        .S7
                        .LOC     371
SA1:       .SA1-1
SA2:       .SA2-1
SA3:       .SA3-1
CA1:       .CA1-1
CA2:       .CA2-1
CA3:       .CA3-1

                        .NREL
           LDA          1,S5
           STA          1,.S5
B2:        LDA          1,ATLU
           STA          1,20
           STA          1,21
           LDA          1,NTLU
           STA          1,.NTLU
B3:        LDA          1,@20
           MOVZR        1,1
           STA          1,@21
           DSZ          .NTLU
           JMP          B3
           LDA          1,RS
           INC          1,1
           STA          1,RS
           DSZ          .S5
           JMP          B2
           JSR          @CCAL
           JSR          @MCON
           LDA          1,RS
           JSR          @BD
           JMP          @S7

.STLU:     LDA          1,ATLU
           STA          1,20
           STA          1,21
           LDA          1,NTLU
           STA          1,.NTLU
B1:        LDA          1,@20
           MOVZR        1,1
           STA          1,@21
           DSZ          .NTLU
           JMP          B1
           LDA          1,RS
           INC          1,1
           STA          1,RS
           JMP          @XR
```

```
ATLU:     .TLUD
NTLU:     764
.NTLU:    0
RS:       0
S5:       6
.S5:      0
.RB:      .BLK     1000
Z:        0

.S7:      LDA      2,N
          STA      2,.N
          LDA      2,I1
          STA      2,20
          LDA      1,Z
S4:       STA      1,@20
          DSZ      .N
          JMP      S4

          LDA      2,N
          STA      2,.NN
          JSR      @CFTST
A:        0
.NN:      0
.R1:      .BLK     2000
.SA1:     .BLK     400
.CA1:     .BLK     400
.SA2:     .BLK     200
.CA2:     .BLK     200
.SA3:     .BLK     200
.CA3:     .BLK     200

          LDA      1,@ASCNT
          JSR      @BD
          LDA      1,@ASCNT
          LDA      2,MAN
          SUBO     1,2
          MOVZL    2,2
          STA      2,SRS
          MOV      2,1
          JSR      @ED
          LDA      0,R1
          STA      0,20
          LDA      0,I1
          STA      0,21
          LDA      0,RB
          STA      0,22
S6:       LDA      0,N
          MOVZR    0,0
          MOVZR    0,0
          MOVZR    0,0
          STA      0,.NH
          JSR      @CPB

STR:      LDA      1,@22
          STA      1,TR
          JSR      @SRTD
          JSR      @DOR

          LDA      1,@20
          LDA      2,@21
          JSR      PTG
          LDA      2,TR
          SUBO     2,1
          MOVZL    1,1
          MOVZL    1,1
          JSR      @SRTC
          JSR      @DOR
          DSZ      .NH
          JMP      STR
          JSR      @CPE

XX:       HALT
          DSZ      .NQ
          JMP      S6
          HALT
          HALT
PTG:      STA      3,RPTG
          STA      2,TC

          MOV      1,2
          JSR      @MPY
          STA      0,HS
          STA      1,LS

          LDA      1,TC
          LDA      2,TC
          JSR      @MF '

          JSR      @DADD
AHS:      HS
          LDA      3,SRS
          STA      3,.SRS
S1:       MOVZR    0,0
          MOVR     1,1
          DSZ      .SRS
          JMP      S1

          JSR      @DSQR
          JMP      @RPTG

TC:       0
HS:       0
LS:       0
RPTG:     0
.NH:      0
MPY:      .MPY
DADD:     .DADD
SRS:      0
TR:       0
.NQ:      4
.SRS:     0
MAN:      6
DSQR:     .DSQR
.END
```

A4.12.7 Numerical Analysis of Errors - ESR Subroutine

The ESR subroutine implements the algorithm discussed in chapter 5.3 The routine sorts out which quantising level, H_p, accommodates F_k and then accumulates the magnitude of the error and the squared error in accumulators ET_p and ST_p respectively. The number of harmonics with magnitude within the limits of H_p is stored in location N_p. The limits of H_p in machine code are:

$$40000_8 \leq H_{14} < 77777_8$$

$$20000_8 \leq H_{13} < 37777_8.$$

$$\vdots$$

$$4_8 \leq H_2 < 7_8$$

$$2_8 \leq H_1 < 3_8$$

$$H_o = 1.$$

The most significant bit (MSB) of F_k is ANDed with 40000_8, 20000_8 10000_8 ..... 1_8 successively. eg.If the result of ANDing with 10000_8 is TRUE then F_k is in quantising level H_{12} and the magnitude of the error is added to ET_{12}, the squared error is added to ST_{12} and N_{12} is incremented by 1. ESR processes the remaining harmonics in a similar way.

```
LCK:  40000o
.ESR: STA   3,RESR        Save return to STR.
      LDA   2,TDF         Load Fk into TDF.
      LDA   3,LCK         Load 40000 into LCK
      AND   2,3,SZR       AND Fk and 40000 8.  If result is
                          FALSE skip next instruction.
```

```
        JMP    L15            Result is TRUE Go to L15

        MOVZR 3,3             Result is false.  Halve 40000₈.

        AND    2,2,SZR        AND F_k and 20000₈
```

These steps are repeated for all H_p.

```
        .
        .
        .

L15:  LDA    2,ET15           Get current error magnitude subtotal
                              from ET15.

        ADD 1,2               Add error

        STA 1,ET 15           Store new subtotal in ET15.

        MOV    1,2     )
                       )
        JSR    @AMPY   )
                       )
        LDA    2,ST15  )      Add error squared to ST15
                       )
        ADD    1,2     )

        STA    2,ST15

        ISZ    N15            Increment N_15, the number of harmonics
                              with magnitudes in level H_15.

        JMP    L0

L14:    .
        .
        .
        .
        .
        .
        .
L0:   JMP    @RESR            Return to STR subroutine.
```

The STR routine is slightly modified.

```
STR:  LDA   1,@20  )
                   )
      LDA   2,@21  )      Calculate FFT resultant harmonic
                   )
      JSR   PTG    )

      LDA   2,@22         Get DFTTLU resultant harmonic

      STA   2,TDF         Store in TDF for ESR subroutine

      SUBO  2,1           Calculate error.

      MOVZL 1,1,SZC)      If sign is negative, take the two's
                   )
      NEG   1,1    )      complement, ie. the magnitude of the

                          error is required

      JSR   @ESR          Call ESR subroutine

      DSZ   .NH           Decrement index

      JMP   STR           Index not zero.   Process next

                          harmonic

      DSZ   .NR           Decrement program index (Set to 32)

      JMP   @S2           Index not zero. Process next transform

      HALT                Index zero.   32 transforms processed.
```

On pressing the 'CONTINUE' switch on the console, the program writes ET_p, ST_p, N_p, $p = 0 \ldots\ldots 14$ from the accumulated results of 32 transforms and then returns to DDOS mode.

```
                        .TITL   G128
                        .ENT    STLU.R
                        .EXTN   .DIVI,PUTC
                        .ENT    R1,SA1,CA1,SA2,CA2,SA3,CA3
                        .EXTD   CFTST,MCON,CCAL,N
                        .EXTN   .DADD,.MPY,.DSGR,.BDEC,SCNT,.TLUD
                        .EXTN   Q1.1
                        .ZREL
                        .LOC    250
STLU:   .STLU
XR:     Q1.1
BD:     .BDEC
ASCNT:  SCNT
R1:     .R1-1
I1:     .SA1-1
.N:     0
RB:     .RB-1
S7:     .S7
                        .LOC    371
SA1:    .SA1-1
SA2:    .SA2-1
SA3:    .SA3-1
CA1:    .CA1-1
CA2:    .CA2-1
CA3:    .CA3-1

                .NREL
        LDA     1,S5
        STA     1,.S5
B2:     LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B3:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B3
        LDA     1,RS
        INC     1,1
        STA     1,RS
        DSZ     .S5
        JMP
        JSR       AL
.S2:    JSR        N
        LDA
        JSR
        JMP     @S7

.STLU:  LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B1:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B1
        LDA     1,RS
        INC     1,1
        STA     1,RS
        LDA     1,MAN
        INC     1,1
        STA     1,MAN
        JMP     @XR
```

```
ATLU:   .TLUD
NTLU:   764
.NTLU:  0
RS:     0
S5:     3
MAN:    3
.S5:    0
.RB:    .BLK    100
Z:      0

.S7:    LDA     2,N
        STA     2,.N
        LDA     2,I1
        STA     2,20
        LDA     1,Z
S4:     STA     1,@20
        DSZ     .N
        JMP     S4

        LDA     2,N
        STA     2,.NN
        JSR     @CFTST
A:      0
.NN:    0
.R1:    .BLK    200
.SA1:   .BLK    40
.CA1:   .BLK    40
.SA2:   .BLK    20
.CA2:   .BLK    20
.SA3:   .BLK    20
.CA3:   .BLK    20

        LDA     1,@ASCNT
        JSR     @BD
        LDA     1,@ASCNT
        LDA     2,@AMAN
        SUBO    1,2
        MOVZL   2,2
        STA     2,SRS
        MOV     2,1
        JSR     @BD
        LDA     0,R1
        STA     0,20
        LDA     0,I1
        STA     0,21
        LDA     0,RB
        STA     0,22
        LDA     0,N
        MOVZR   0,0
        STA     0,.NH

STR:    LDA     1,@20
        LDA     2,@21
        JSR     PTG
        LDA     2,@22
        STA     2,TDF
        SUBO    2,1
        MOVL#   1,1,SZC
        NEG     1,1
        JSR     @ESR
        DSZ     .NH
        JMP     STR
        DSZ     .NR
        JMP     @S2
        HALT
```

```
        LDA     1,AET15
        LDA     2,AN15
        JSR     S9
        LDA     1,AET10
        LDA     2,AN10
        JSR     S9
        LDA     1,AET5
        LDA     2,AN5
        JSR     S9
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AET15
        JSR     S6
        LDA     1,AET10
        JSR     S6
        LDA     1,AET5
        JSR     S6
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AST15
        JSR     S6
        LDA     1,AST10
        JSR     S6
        LDA     1,AST5
        JSR     S6
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AN15
        JSR     S6
        LDA     1,AN10
        JSR     S6
        LDA     1,AN5
        JSR     S6

        LDA     0,ASS
        JSR     @APUTC
        LDA     0,ASS
        JSR     @APUTC
        JMP     @RDOS

S9:     STA     3,SRET
        STA     1,20
        STA     2,21
        LDA     1,NN5
        STA     1,.NN5
S91:    LDA     1,@20
        LDA     2,@21
        JSR     @DIVI
        JSR     @BD
        DSZ     .NN5
        JMP     S91
        JMP     @SRET

S6:     STA     3,SRET
        STA     1,20
        LDA     1,NN5
        STA     1,.NN5
S61:    LDA     1,@20
        JSR     @BD
        DSZ     .NN5
        JMP     S61
        JMP     @SRET
```

```
PTG:    STA     3,RPTG
        STA     2,TC

        MOV     1,2
        JSR     @MPY
        STA     0,HS
        STA     1,LS

        LDA     1,TC
        LDA     2,TC
        JSR     @MPY

        JSR     @DADD
AHS:    HS
        LDA     3,SRS
        STA     3,.SRS
S1:     MOVZR   0,0
        MOVR    1,1
        DSZ     .SRS
        JMP     S1
        JSR     @DSQR
        JMP     @RPTG

AMAN:   MAN
SRS:    0
.SRS:   0
NN5:    5
.NN5:   0
.NH:    0
S2:     .S2
AET15:  ET15-1
AET10:  ET10-1
AET5:   ET5-1
AST15:  ST15-1
AST10:  ST10-1
AST5:   ST5-1
AN15:   N15-1
AN10:   N10-1
AN5:    N5-1
SRET:   0
APUTC:  PUTC
.NR:    40
ESR:    .ESR
TC:     0
HS:     0
LS:     0
RPTG:   0
MPY:    .MPY
DIVI:   .DIVI
DADD:   .DADD
DSQR:   .DSQR
TDF:    0
LCK:    40000
RDOS:   37600
ASS:    0
```

```
.ESR:   STA     3,RESR
        LDA     2,TDF
        LDA     3,LCK

        AND#    2,3,SZR
        JMP     L15
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L14
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L13
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L12
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L11
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L10
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L9
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L8
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L7
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L6
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L5
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L4
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L3
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L2
        JMP     @L1

L5:     .L5
L4:     .L4
L3:     .L3
L2:     .L2
L1:     .L1

L15:    LDA     2,ET15
        ADD     1,2
        STA     2,ET15
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST15
        ADD     1,2
        STA     2,ST15
        ISZ     N15
        JMP     L0
L14:    LDA     2,ET14
        ADD     1,2
        STA     2,ET14
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST14
        ADD     1,2
        STA     2,ST14
        ISZ     N14
        JMP     L0
L13:    LDA     2,ET13
        ADD     1,2
        STA     2,ET13
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST13
        ADD     1,2
        STA     2,ST13
        ISZ     N13
        JMP     L0
L12:    LDA     2,ET12
        ADD     1,2
        STA     2,ET12
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST12
        ADD     1,2
        STA     2,ST12
        ISZ     N12
        JMP     L0
L11:    LDA     2,ET11
        ADD     1,2
        STA     2,ET11
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST11
        ADD     1,2
        STA     2,ST11
        ISZ     N11
        JMP     L0
ET15:   0
ET14:   0
ET13:   0
ET12:   0
ET11:   0
ST15:   0
ST14:   0
ST13:   0
ST12:   0
ST11:   0
N15:    0
N14:    0
N13:    0
N12:    0
N11:    0
L0:     JMP     @RESR
RESR:   0
AMPY:   .MPY

L10:    LDA     2,ET10
        ADD     1,2
        STA     2,ET10
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST10
        ADD     1,2
        STA     2,ST10
        ISZ     N10
        JMP     L0
L9:     LDA     2,ET9
        ADD     1,2
        STA     2,ET9
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST9
        ADD     1,2
        STA     2,ST9
        ISZ     N9
        JMP     L0
L8:     LDA     2,ET8
        ADD     1,2
        STA     2,ET8
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST8
        ADD     1,2
        STA     2,ST8
        ISZ     N8
        JMP     L0
L7:     LDA     2,ET7
        ADD     1,2
        STA     2,ET7
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST7
        ADD     1,2
        STA     2,ST7
        ISZ     N7
        JMP     L0
L6:     LDA     2,ET6
        ADD     1,2
        STA     2,ET6
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST6
        ADD     1,2
        STA     2,ST6
        ISZ     N6
        JMP     L0
ET10:   0
ET9:    0
ET8:    0
ET7:    0
ET6:    0
ST10:   0
ST9:    0
ST8:    0
ST7:    0
ST6:    0
N10:    0
N9:     0
N8:     0
N7:     0
N6:     0
```

```
.L5:    LDA     2,ET5
        ADD     1,2
        STA     2,ET5
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST5
        ADD     1,2
        STA     2,ST5
        ISZ     N5
        JMP     L0
.L4:    LDA     2,ET4
        ADD     1,2
        STA     2,ET4
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST4
        ADD     1,2
        STA     2,ST4
        ISZ     N4
        JMP     L0
.L3:    LDA     2,ET3
        ADD     1,2
        STA     2,ET3
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST3
        ADD     1,2
        STA     2,ST3
        ISZ     N3
        JMP     L0
.L2:    LDA     2,ET2
        ADD     1,2
        STA     2,ET2
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST2
        ADD     1,2
        STA     2,ST2
        ISZ     N2
        JMP     L0
.L1:    LDA     2,ET1
        ADD     1,2
        STA     2,ET1
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST1
        ADD     1,2
        STA     2,ST1
        ISZ     N1
        JMP     L0
ET5:    0
ET4:    0
ET3:    0
ET2:    0
ET1:    0
ST5:    0
ST4:    0
ST3:    0
ST2:    0
ST1:    0
N5:     0
N4:     0
N3:     0
N2:     0
N1:     0
.END
---
---
```

```
                        .TITL   G256
                        .ENT    STLU,RB
                        .EXTN   .DIVI,PUTC
                        .ENT    R1,SA1,CA1,SA2,CA2,SA3,CA3
                        .EXTD   CFTST,MCON,CCAL,N
                        .EXTN   .DADD,.MPY,.DSQR,.BDEC,SCNT,.TLUD
                        .EXTN   Q1.1
                        .ZREL
                        .LOC    250
STLU:   .STLU
XR:     Q1.1
BD:     .BDEC
ASCNT:  SCNT
R1:     .R1-1
I1:     .SA1-1
.N:     0
RB:     .RB-1
S7:     .S7
                        .LOC    371
SA1:    .SA1-1
SA2:    .SA2-1
SA3:    .SA3-1
CA1:    .CA1-1
CA2:    .CA2-1
CA3:    .CA3-1

                .NREL
        LDA     1,S5
        STA     1,.S5
B2:     LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B3:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B3
        LDA     1,RS
        INC     1,1
        STA     1,RS
        DSZ     .S5
        JMP     B2
        JSR     @CCAL
.S2     JSR     @MCON
        LDA     1,RS
        JSR     @BD
        JMP     @S7

.STLU:  LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B1:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B1
        LDA     1,RS
        INC     1,1
        STA     1,RS
        LDA     1,MAN
        INC     1,1
        STA     1,MAN
        JMP     @XR
```

```
ATLU:    .TLUD
NTLU:    764
.NTLU:   0
RS:      0
S5:      4
MAN:     4
.S5:     0
.RB:     .BLK     200
Z:       0

.S7:     LDA      2,N
         STA      2,.N
         LDA      2,I1
         STA      2,20
         LDA      1,Z
S4:      STA      1,@20
         DSZ      .N
         JMP      S4

         LDA      2,N
         STA      2,.NN
         JSR      @CFTST
A:       0
.NN:     0
.R1:     .BLK     400
.SA1:    .BLK     100
.CA1:    .BLK     100
.SA2:    .BLK     40
.CA2:    .BLK     40
.SA3:    .BLK     40
.CA3:    .BLK     40

         LDA      1,@ASCNT
         JSR      @BD
         LDA      1,@ASCNT
         LDA      2,@AMAN
         SUBO     1,2
         MOVZL    2,2
         STA      2,SRS
         MOV      2,1
         JSR      @BD
         LDA      0,R1
         STA      0,20
         LDA      0,I1
         STA      0,21
         LDA      0,RB
         STA      0,22
         LDA      0,N
         MOVZR    0,0
         STA      0,.NH
STR:     LDA      1,@20
         LDA      2,@21
         JSR      PTG
         LDA      2,@22
         STA      2,TDF
         SUBO     2,1
         MOVL#    1,1,SZC
         NEG      1,1
         JSR      @ESR
         DSZ      .NH
         JMP      STR
         DSZ      .NR
         JMP      @S2
         HALT
```

```
         LDA      1,AET15
         LDA      2,AN15
         JSR      S9
         LDA      1,AET10
         LDA      2,AN10
         JSR      S9
         LDA      1,AET5
         LDA      2,AN5
         JSR      S9
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AET15
         JSR      S6
         LDA      1,AET10
         JSR      S6
         LDA      1,AET5
         JSR      S6
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AST15
         JSR      S6
         LDA      1,AST10
         JSR      S6
         LDA      1,AST5
         JSR      S6
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AN15
         JSR      S6
         LDA      1,AN10
         JSR      S6
         LDA      1,AN5
         JSR      S6

         LDA      0,ASS
         JSR      @APUTC
         LDA      0,ASS
         JSR      @APUTC
         JMP      @RDOS
S9:      STA      3,SRET
         STA      1,20
         STA      2,21
         LDA      1,NN5
         STA      1,.NN5
S91:     LDA      1,@20
         LDA      2,@21
         JSR      @DIVI
         JSR      @BD
         DSZ      .NN5
         JMP      S91
         JMP      @SRET

S6:      STA      3,SRET
         STA      1,20
         LDA      1,NN5
         STA      1,.NN5
S61:     LDA      1,@20
         JSR      @BD
         DSZ      .NN5
         JMP      S61
         JMP      @SRET
```

```
PTG:     STA      3,RPTG
         STA      2,TC

         MOV      1,2
         JSR      @MPY
         STA      0,HS
         STA      1,LS

         LDA      1,TC
         LDA      2,TC
         JSR      @MPY

         JSR      @DADD
AHS:     HS
         LDA      3,SRS
         STA      3,.SRS
S1:      MOVZR    0,0
         MOVR     1,1
         DSZ      .SRS
         JMP      S1
         JSR      @DSQR
         JMP      @RPTG

AMAN:    MAN
SRS:     0
.SRS:    0
NN5:     5
.NN5:    0
.NH:     0
S2:      .S2
AET15:   ET15-1
AET10:   ET10-1
AET5:    ET5-1
AST15:   ST15-1
AST10:   ST10-1
AST5:    ST5-1
AN15:    N15-1
AN10:    N10-1
AN5:     N5-1
SRET:    0
APUTC:   PUTC
.NR:     40
ESR:     .ESR
TC:      0
HS:      0
LS:      0
RPTG:    0
MPY:     .MPY
DIVI:    .DIVI
DADD:    .DADD
DSQR:    .DSQR
TDF:     0
LCK:     40000
RDOS:    37600
ASS:     0
```

```
.ESR:   STA     3,RESR
        LDA     2,TDF
        LDA     3,LCK

        AND#    2,3,SZR
        JMP     L15
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L14
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L13
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L12
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L11
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L10
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L9
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L8
        MOVZR   3 3
        AND#    2,3,SZR
        JMP     L7
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L6
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L5
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L4
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L3
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L2
        JMP     @L1

L3:     .L5
L4:     .L4
L3:     .L3
L2:     .L2
L1:     .L1

L15:    LDA     2,ET15
        ADD     1,2
        STA     2,ET15
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST15
        ADD     1,2
        STA     2,ST15
        ISZ     N15
        JMP     L0
L14:    LDA     2,ET14
        ADD     1,2
        STA     2,ET14
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST14
        ADD     1,2
        STA     2,ST14
        ISZ     N14
        JMP     L0
L13:    LDA     2,ET13
        ADD     1,2
        STA     2,ET13
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST13
        ADD     1,2
        STA     2,ST13
        ISZ     N13
        JMP     L0
L12:    LDA     2,ET12
        ADD     1,2
        STA     2,ET12
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST12
        ADD     1,2
        STA     2,ST12
        ISZ     N12
        JMP     L0
L11:    LDA     2,ET11
        ADD     1,2
        STA     2,ET11
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST11
        ADD     1,2
        STA     2,ST11
        ISZ     N11
        JMP     L0
ET15:   0
ET14:   0
ET13:   0
ET12:   0
ET11:   0
ST15:   0
ST14:   0
ST13:   0
ST12:   0
ST11:   0
N15:    0
N14:    0
N13:    0
N12:    0
N11:    0
L0:     JMP     @RESR
RESR:   0
AMPY:   .MPY

L10:    LDA     2,ET10
        ADD     1,2
        STA     2,ET10
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST10
        ADD     1,2
        STA     2,ST10
        ISZ     N10
        JMP     L0
L9:     LDA     2,ET9
        ADD     1,2
        STA     2,ET9
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST9
        ADD     1,2
        STA     2,ST9
        ISZ     N9
        JMP     L0
L8:     LDA     2,ET8
        ADD     1,2
        STA     2,ET8
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST8
        ADD     1,2
        STA     2,ST8
        ISZ     N8
        JMP     L0
L7:     LDA     2,ET7
        ADD     1,2
        STA     2,ET7
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST7
        ADD     1,2
        STA     2,ST7
        ISZ     N7
        JMP     L0
L6:     LDA     2,ET6
        ADD     1,2
        STA     2,ET6
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST6
        ADD     1,2
        STA     2,ST6
        ISZ     N6
        JMP     L0
ET10:   0
ET9:    0
ET8:    0
ET7:    0
ET6:    0
ST10:   0
ST9:    0
ST8:    0
ST7:    0
ST6:    0
N10:    0
N9:     0
N8:     0
N7:     0
N6:     0
```

```
.L5:    LDA     2,ET5
        ADD     1,2
        STA     2,ET5
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST5
        ADD     1,2
        STA     2,ST5
        ISZ     N5
        JMP     L0
.L4:    LDA     2,ET4
        ADD     1,2
        STA     2,ET4
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST4
        ADD     1,2
        STA     2,ST4
        ISZ     N4
        JMP     L0
 L3:    LDA     2,ET3
        ADD     1,2
        STA     2,ET3
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST3
        ADD     1,2
        STA     2,ST3
        ISZ     N3
        JMP     L0
.L2:    LDA     2,ET2
        ADD     1,2
        STA     2,ET2
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST2
        ADD     1,2
        STA     2,ST2
        ISZ     N2
        JMP     L0
.L1:    LDA     2,ET1
        ADD     1,2
        STA     2,ET1
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST1
        ADD     1,2
        STA     2,ST1
        ISZ     N1
        JMP     L0
ET5:    0
ET4:    0
ET3:    0
ET2:    0
ET1:    0
ST5:    0
ST4:    0
ST3:    0
ST2:    0
ST1:    0
N5:     0
N4:     0
N3:     0
N2:     0
N1:     0
.END
```

```
                        .TITL   G512
                        .ENT    STLU,RB
                        .EXTN   .DIVI,PUTC
                        .ENT    R1,SA1,CA1,SA2,CA2,SA3,CA3
                        .EXTD   CFTST,MCON,CCAL,N
                        .EXTN   .DADD,.MPY,.DSQR,.BDEC,SCNT, TLUD
                        .EXTN   Q1.1
                        .ZREL
                        .LOC    25
STLU:   .STLU
XR:     Q1.1
BD:     .BDEC
ASCNT:  SCNT
R1:     .R1-1
I1:     .SA1-1
.N:     0
RB:     .RB-1
S7:     .S7
                        .LOC    371
SA1:    .SA1-1
SA2:    .SA2-1
SA3:    .SA3-1
CA1:    .CA1-1
CA2:    .CA2-1
CA3:    .CA3-1

                .NREL
        LDA     1,S5
        STA     1,.S5
B2:     LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B3:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B3
        LDA     1,RS
        INC     1,1
        STA     1,RS
        DSZ     .S5
        JMP     B2
        JSR     @CCAL
.S2:    JSR     @MCON
        LDA     1,RS
        JSR     @BD
        JMP     @S7

.STLU:  LDA     1,ATLU
        STA     1,20
        STA     1,21
        LDA     1,NTLU
        STA     1,.NTLU
B1:     LDA     1,@20
        MOVZR   1,1
        STA     1,@21
        DSZ     .NTLU
        JMP     B1
        LDA     1,RS
        INC     1,1
        STA     1,RS
        LDA     1,MAN
        INC     1,1
        STA     1,MAN
        JMP     @XR
```

```
ATLU:   .TLUD
NTLU:   764
.NTLU:  0
RS:     0
S5:     5
MAN:    5
.S5:    0
.RB:    .BLK    400
Z:      0

.S7:    LDA     2,N
        STA     2,.N
        LDA     2,I1
        STA     2,20
        LDA     1,Z
S4:     STA     1,@20
        DSZ     .N
        JMP     S4

        LDA     2,N
        STA     2,.NN
        JSR     @CFTST
A:      0
.NN:    0
.R1:    .BLK    1000
.SA1:   .BLK    200
.CA1:   .BLK    200
.SA2:   .BLK    100
.CA2:   .BLK    100
.SA3:   .BLK    100
.CA3:   .BLK    100

        LDA     1,@ASCNT
        JSR     @BD
        LDA     1,@ASCNT
        LDA     2,@AMAN
        SUBO    1,2
        MOVZL   2,2
        STA     2,SRS
        MOV     2,1
        JSR     @BD
        LDA     0,R1
        STA     0,20
        LDA     0,I1
        STA     0,21
        LDA     0,RB
        STA     0,22
        LDA     0,N
        MOVZR   0,0
        STA     0,.NH

STR:    LDA     1,@20
        LDA     2,@21
        JSR     PTG
        LDA     2,@22
        STA     2,TDF
        SUBO    2,1
        MOVL#   1,1,SZC
        NEG     1,1
        JSR     @ESR
        DSZ     .NH
        JMP     STR
        DSZ     .NR
        JMP     @S2
        HALT
```

```
        LDA     1,AET15
        LDA     2,AN15
        JSR     S9
        LDA     1,AET10
        LDA     2,AN10
        JSR     S9
        LDA     1,AET5
        LDA     2,AN5
        JSR     S9
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AET15
        JSR     S6
        LDA     1,AET10
        JSR     S6
        LDA     1,AET5
        JSR     S6
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AST15
        JSR     S6
        LDA     1,AST10
        JSR     S6
        LDA     1,AST5
        JSR     S6
        LDA     0,ASS
        JSR     @APUTC

        LDA     1,AN15
        JSR     S6
        LDA     1,AN10
        JSR     S6
        LDA     1,AN5
        JSR     S6

        LDA     0,ASS
        JSR     @APUTC
        LDA     0,ASS
        JSR     @APUTC
        JMP     @RDOS

S9:     STA     3,SRET
        STA     1,20
        STA     2,21
        LDA     1,NN5
        STA     1,.NN5
S91:    LDA     1,@20
        LDA     2,@21
        JSR     @DIVI
        JSR     @BD
        DSZ     .NN5
        JMP     S91
        JMP     @SRET

S6:     STA     3,SRET
        STA     1,20
        LDA     1,NN5
        STA     1,.NN5
S61:    LDA     1,@20
        JSR     @BD
        DSZ     .NN5
        JMP     S61
        JMP     @SRET
```

```
PTG:    STA     3,RPTG
        STA     2,TC

        MOV     1,2
        JSR     @MPY
        STA     0,HS
        STA     1,LS

        LDA     1,TC
        LDA     2,TC
        JSR     @MPY

        JSR     @DADD
AHS:    HS
        LDA     3,SRS
        STA     3,.SRS
S1:     MOVZR   0,0
        MOVR    1,1
        DSZ     .SRS
        JMP     S1
        JSR     @DSQR
        JMP     @RPTG

AMAN:   MAN
SRS:    0
.SRS:   0
NN5:    5
.NN5:   0
.NH:    0
S2:     .S2
AET15:  ET15-1
AET10:  ET10-1
AET5:   ET5-1
AST15:  ST15-1
AST10:  ST10-1
AST5:   ST5-1
AN15:   N15-1
AN10:   N10-1
AN5:    N5-1
SRET:   0
APUTC.  PUTC
.NR:    40
ESR:    .ESR
TC:     0
HS:     0
LS:     0
RPTG:   0
MPY:    .MPY
DIVI:   .DIVI
DADD:   .DADD
DSQR:   .DSQR
TDF:    0
LCK:    40000
RDOS:   37600
ASS:    0
```

```
.ESR:   STA     3,RESR
        LDA     2,TDF
        LDA     3,LCK

        AND#    2,3,SZR
        JMP     L15
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L14
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L13
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L12
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L11
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L10
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L9
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L8
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L7
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L6
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L5
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L4
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L3
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L2
        JMP     @L1

L5:     .L5
L4:     .L4
L3:     .L3
L2:     .L2
L1:     .L1
```

```
L15:    LDA     2,ET15
        ADD     1,2
        STA     2,ET15
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST15
        ADD     1,2
        STA     2,ST15
        ISZ     N15
        JMP     L0
L14:    LDA     2,ET14
        ADD     1,2
        STA     2,ET14
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST14
        ADD     1,2
        STA     2,ST14
        ISZ     N14
        JMP     L0
L13:    LDA     2,ET13
        ADD     1,2
        STA     2,ET13
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST13
        ADD     1,2
        STA     2,ST13
        ISZ     N13
        JMP     L0
L12:    LDA     2,ET12
        ADD     1,2
        STA     2,ET12
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST12
        ADD     1,2
        STA     2,ST12
        ISZ     N12
        JMP     L0
L11:    LDA     2,ET11
        ADD     1,2
        STA     2,ET11
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST11
        ADD     1,2
        STA     2,ST11
        ISZ     N11
        JMP     L0
ET15:   0
ET14:   0
ET13:   0
ET12:   0
ET11:   0
ST15:   0
ST14:   0
ST13:   0
ST12:   0
ST11:   0
N15:    0
N14:    0
N13:    0
N12:    0
N11:    0
L0:     JMP     @RESR
RESR:   0
AMPY:   .MPY
```

```
L10:    LDA     2,ET10
        ADD     1,2
        STA     2,ET10
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST10
        ADD     1,2
        STA     2,ST10
        ISZ     N10
        JMP     L0
L9:     LDA     2,ET9
        ADD     1,2
        STA     2,ET9
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST9
        ADD     1,2
        STA     2,ST9.
        ISZ     N9
        JMP     L0
L8:     LDA     2,ET8
        ADD     1,2
        STA     2,ET8
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST8
        ADD     1,2
        STA     2,ST8
        ISZ     N8
        JMP     L0
L7:     LDA     2,ET7
        ADD     1,2
        STA     2,ET7
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST7
        ADD     1,2
        STA     2,ST7
        ISZ     N7
        JMP     L0
L6:     LDA     2,ET6
        ADD     1,2
        STA     2,ET6
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST6
        ADD     1,2
        STA     2,ST6
        ISZ     N6
        JMP     L0
ET10:   0
ET9:    0
ET8:    0
ET7:    0
ET6:    0
ST10:   0
ST9:    0
ST8:    0
ST7:    0
ST6:    0
N10:    0
N9:     0
N8:     0
N7:     0
N6:     0
```

```
.L5:    LDA     2,ET5
        ADD     1,2
        STA     2,ET5
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST5
        ADD     1,2
        STA     2,ST5
        ISZ     N5
        JMP     L0
.L4:    LDA     2,ET4
        ADD     1,2
        STA     2,ET4
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST4
        ADD     1,2
        STA     2,ST4
        ISZ     N4
        JMP     L0
.L3:    LDA     2,ET3
        ADD     1,2
        STA     2,ET3
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST3
        ADD     1,2
        STA     2,ST3
        ISZ     N3
        JMP     L0
.L2:    LDA     2,ET2
        ADD     1,2
        STA     2,ET2
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST2
        ADD     1,2
        STA     2,ST2
        ISZ     N2
        JMP     L0
.L1:    LDA     2,ET1
        ADD     1,2
        STA     2,ET1
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST1
        ADD     1,2
        STA     2,ST1
        ISZ     N1
        JMP     L0
ET5:    0
ET4:    0
ET3:    0
ET2:    0
ET1:    0
ST5:    0
ST4:    0
ST3:    0
ST2:    0
ST1:    0
N5:     0
N4:     0
N3:     0
N2:     0
N1:     0
.END
---
---
```

```
---
                    .TITL   G1024
                    .ENT    STLU,RB
                    .EXTN   .DIVI,PUTC
                    .ENT    R1,SA1,CA1,SA2,CA2,SA3,CA3
                    .EXTD   CFTST,MCON,CCAL,N
                    .EXTN   .DADD,.MPY,.DSQR,.BDEC,SCNT,.TLUD
                    .EXTN   Q1.1
                    .ZREL
                    .LOC    250
STLU:    .STLU
XR:      Q1.1
BD:      .BDEC
ASCNT:   SCNT
R1:      .R1-1
I1:      .SA1-1
.N:      0
RB:      .RB-1
S7:      .S7
                    .LOC    371
SA1:     .SA1-1
SA2:     .SA2-1
SA3:     .SA3-1
CA1:     .CA1-1
CA2:     .CA2-1
CA3:     .CA3-1

                    .NREL
         LDA        1,S5
         STA        1,.S5
B2:      LDA        1,ATLU
         STA        1,20
         STA        1,21
         LDA        1,NTLU
         STA        1,.NTLU
B3:      LDA        1,@20
         MOVZR      1,1
         STA        1,@21
         DSZ        .NTLU
         JMP        B3
         LDA        1,RS
         INC        1,1
         STA        1,RS
         DSZ        .S5
         JMP        B2
         JSR        @CCAL
.S2:     JSR        @MCON
         LDA        1,RS
         JSR        @BD
         JMP        @S7

.STLU:   LDA        1,ATLU
         STA        1,20
         STA        1,21
         LDA        1,NTLU
         STA        1,.NTLU
B1:      LDA        1,@20
         MOVZR      1,1
         STA        1,@21
         DSZ        .NTLU
         JMP        B1
         LDA        1,RS
         INC        1,1
         STA        1,RS
         LDA        1,MAN
         INC        1,1
         STA        1,MAN
         JMP        @XP
```

```
ATLU:    .TLUD
NTLU:    764
.NTLU:   0
RS:      0
S5:      6
MAN:     6
.S5:     0
.RB:     .BLK     1000
Z:       0

.S7:     LDA      2,N
         STA      2,.N
         LDA      2,I1
         STA      2,20
         LDA      1,Z
S4:      STA      1,@20
         DSZ      .N
         JMP      S4

         LDA      2,N
         STA      2,.NN
         JSR      @CFTST
A:       0
.NN:     0
.R1:     .BLK     2000
.SA1:    .BLK     400
.CA1:    .BLK     400
.SA2:    .BLK     200
.CA2:    .BLK     200
.SA3:    .BLK     200
.CA3:    .BLK     200

         LDA      1,@ASCNT
         JSR      @BD
         LDA      1,@ASCNT
         LDA      2,@AMAN
         SUBO     1,2
         MOVZL    2,2
         STA      2,SRS
         MOV      2,1
         JSR      @BD
         LDA      0,R1
         STA      0,20
         LDA      0,I1
         STA      0,21
         LDA      0,RB
         STA      0,22
         LDA      0,N
         MOVZR    0,0
         STA      0,.N

STR:     LDA      1,@20
         LDA      2,@21
         JSR      PTG
         LDA      2,@22
         STA      2,TDF
         SUBO     2,1
         MOVL#    1,1,SZC
         NEG      1,1
         JSR      @ESR
         DSZ      .NH
         JMP      STR
         DSZ      .NR
         JMP      @S2
         HALT

         LDA      1,AET15
         LDA      2,AN15
         JSR      S9
         LDA      1,AET10
         LDA      2,AN10
         JSR      S9
         LDA      1,AET5
         LDA      2,AN5
         JSR      S9
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AET15
         JSR      S6
         LDA      1,AET10
         JSR      S6
         LDA      1,AET5
         JSR      S6
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AST15
         JSR      S6
         LDA      1,AST10
         JSR      S6
         LDA      1,AST5
         JSR      S6
         LDA      0,ASS
         JSR      @APUTC

         LDA      1,AN15
         JSR      S6
         LDA      1,AN10
         JSR      S6
         LDA      1,AN5
         JSR      S6

         LDA      0,ASS
         JSR      @APUTC
         LDA      0,ASS
         JSR      @APUTC
         JMP      @RDOS

S9:      STA      3,SRET
         STA      1,20
         STA      2,21
         LDA      1,NN5
         STA      1,.NN5
S91:     LDA      1,@20
         LDA      2,@21
         JSR      @DIVI
         JSR      @BD
         DSZ      .NN5
         JMP      S91
         JMP      @SRET

S6:      STA      3,SRET
         STA      1,20
         LDA      1,NN5
         STA      1,.NN5
S61:     LDA      1,@20
         JSR      @BD
         DSZ      .NN5
         JMP      S61
         JMP      @SRET

PTG:     STA      3,RPTG
         STA      2,TC
         MOV      1,2
         JSR      @MPY
         STA      0,HS
         STA      1,LS

         LDA      1,TC
         LDA      2,TC
         JSR      @MPY

         JSR      @DADD
AHS:     HS
         LDA      3,SRS
         STA      3,.SRS
S1:      MOVZR    0,0
         MOVR     1,1
         DSZ      .SRS
         JMP      S1
         JSR      @DSQR
         JMP      @RPTG

AMAN:    MAN
SRS:     0
.SRS:    0
NN5:     5
.NN5:    0
.NH:     0
S2:      .S2
AET15:   ET15-1
AET10:   ET10-1
AET5:    ET5-1
AST15:   ST15-1
AST10:   ST10-1
AST5:    ST5-1
AN15:    N15-1
AN10:    N10-1
AN5:     N5-1
SRET:    0
APUTC:   PUTC
.NR:     40
ESR:     .ESR
TC:      0
HS:      0
LS:      0
RPTG:    0
MPY:     .MPY
DIVI:    .DIVI
DADD:    .DADD
DSQR:    .DSQR
TDF:     0
LCK:     40000
RDOS:    37600
ASS:     0
```

```
.ESR:   STA     3,RESR
        LDA     2,TDF
        LDA     3,LCK

        AND#    2,3,SZR
        JMP     L15
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L14
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L13
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L12
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L11
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L10
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L9
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L8
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L7
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     L6
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L5
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L4
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L3
        MOVZR   3,3
        AND#    2,3,SZR
        JMP     @L2
        JMP     @L1

L5:     .L5
L4:     .L4
L3:     .L3
L2:     .L2
L1:     .L1

L15:    LDA     2,ET15
        ADD     1,2
        STA     2,ET15
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST15
        ADD     1,2
        STA     2,ST15
        ISZ     N15
        JMP     L0
L14:    LDA     2,ET14
        ADD     1,2
        STA     2,ET14
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST14
        ADD     1,2
        STA     2,ST14
        ISZ     N14
        JMP     L0
L13:    LDA     2,ET13
        ADD     1,2
        STA     2,ET13
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST13
        ADD     1,2
        STA     2,ST13
        ISZ     N13
        JMP     L0
L12:    LDA     2,ET12
        ADD     1,2
        STA     2,ET12
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST12
        ADD     1,2
        STA     2,ST12
        ISZ     N12
        JMP     L0
L11:    LDA     2,ET11
        ADD     1,2
        STA     2,ET11
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST11
        ADD     1,2
        STA     2,ST11
        ISZ     N11
        JMP     L0
ET15:   0
ET14:   0
ET13:   0
ET12:   0
ET11:   0
ST15:   0
ST14:   0
ST13:   0
ST12:   0
ST11:   0
N15:    0
N14:    0
N13:    0
N12:    0
N11:    0
L0:     JMP     @RESR
RESR:   0
AMPY:   .MPY

L10:    LDA     2,ET10
        ADD     1,2
        STA     2,ET10
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST10
        ADD     1,2
        STA     2,ST10
        ISZ     N10
        JMP     L0
L9:     LDA     2,ET9
        ADD     1,2
        STA     2,ET9
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST9
        ADD     1,2
        STA     2,ST9
        ISZ     N9
        JMP     L0
L8:     LDA     2,ET8
        ADD     1,2
        STA     2,ET8
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST8
        ADD     1,2
        STA     2,ST8
        ISZ     N8
        JMP     L0
L7:     LDA     2,ET7
        ADD     1,2
        STA     2,ET7
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST7
        ADD     1,2
        STA     2,ST7
        ISZ     N7
        JMP     L0
L6:     LDA     2,ET6
        ADD     1,2
        STA     2,ET6
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST6
        ADD     1,2
        STA     2,ST6
        ISZ     N6
        JMP     L0
ET10:   0
ET9:    0
ET8:    0
ET7:    0
ET6:    0
ST10:   0
ST9:    0
ST8:    0
ST7:    0
ST6:    0
N10:    0
N9:     0
N8:     0
N7:     0
N6:     0
```

```
.L5:    LDA     2,ET5
        ADD     1,2
        STA     2,ET5
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST5
        ADD     1,2
        STA     2,ST5
        ISZ     N5
        JMP     L0
.L4:    LDA     2,ET4
        ADD     1,2
        STA     2,ET4
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST4
        ADD     1,2
        STA     2,ST4
        ISZ     N4
        JMP     L0
.L3:    LDA     2,ET3
        ADD     1,2
        STA     2,ET3
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST3
        ADD     1,2
        STA     2,ST3
        ISZ     N3
        JMP     L0
.L2:    LDA     2,ET2
        ADD     1,2
        STA     2,ET2
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST2
        ADD     1,2
        STA     2,ST2
        ISZ     N2
        JMP     L0
.L1:    LDA     2,ET1
        ADD     1,2
        STA     2,ET1
        MOV     1,2
        JSR     @AMPY
        LDA     2,ST1
        ADD     1,2
        STA     2,ST1
        ISZ     N1
        JMP     L0
ET5:    0
ET4:    0
ET3:    0
ET2:    0
ET1:    0
ST5:    0
ST4:    0
ST3:    0
ST2:    0
ST1:    0
N5:     0
N4:     0
N3:     0
N2:     0
N1:     0
.END
```

A4.13 X - Y Plotter Control Subroutine

The plotter, device code 23, has 3 control lines

1 A - X direction

2 B - Y direction

3. C - Pen lift.

Subroutines CPB and CPE mark the start and the end of the frequency axis by drawing a small mark in the Y direction of $X = -1000_8$ and $X = 0_8$.

Subroutine SRTB moves the pen in the X direction by the value in TBI and then draws a vertical line from Y equal to the harmonic magnitude to Y = 0. Subroutine SRTC is used to plot the error $G_k - F_k$. It also moves the pen in the X direction by TBI and draws a vertical line from Y equal to the error value to $Y = -400_8$. The format of the plot is given in Fig.5.4 in chapter 5.2.1.

Subroutine DOR contains the Assembler instructions that control the plotter. The following convention is adopted for this subroutine.

Register 0 controls line A - X direction

Register 1 controls line B - Y direction

Register 2 controls line C - Pen lift.

The plotter has a built in delay of about 300 mS because of the inertia of the pen carriage. Subroutine DEL increases this delay to about 1 second.

Subroutine SRTB.

```
SRTB: STA   3,RET        Save return
      MOVZR 1,1          Halve harmonic
      LDA   0,.TB        Get current X coordinate
      LDA   2,TB1        Get X shift
      ADD   2,0          Calculate next X coordinate
      STA   0,.TB        Store X coordinate
      LDA   2,PU         Raise pen
      JSR   .DOR         Output the contents of registers 0,1
                         and 2 to the plotter control subroutine
      LDA   2,PD         Lower pen
      JSR   .DOR         Output reg. 0,1,2 to plotter control
                         subroutine
      LDA   1,PU         Load zero into reg. 1.
      JSR   .DOR         Output reg. 0,1,2 to plotter control
                         subroutine.  The plotter draws a
                         line proportional to the original
                         contents of reg 1.
      LDA   2,PU         Raise pen
      JSR   .DOR         Output reg. 0,1,2 to plotter control
                         subroutine
      JMP   RET          Return
```

```
        .TITL   PLPJ
        .ENT    SRTC
        .ENT    CPB,CPE,SRTB,DOP,DEL
        .EXTD   TBI
        .ZREL
        .LOC    340
CPB.    .CPB
CPE     .CPE
SRTA    .SRTB
SRTC.   SRTC
DOP:    DOP
DEL     .DEL
        .NREL
CPB     STA     3,RET
        LDA     0,FTB
        LDA     1,M
        LDA     2,PU
        JSR     .DOR
        JSR     .DEL
        LDA     2,PD
        JSR     DOP
        LDA     1,FTB
        JSR     DOP
        LDA     E TB
        STA     0, TB
        JMP     @RET

CPE.    STA     3,RET
        LDA     0,LTE
        LDA     1,FTB
        LDA     2,PU
        JSR     DOR
        JSR     .DEL
        LDA     2,PD
        JSR     .DOP
        LDA     1,M
        JSR     .DOP
        LDA     2,PU
        JSR     .DOP
        JMP     @RET

.DOP:   STA     3,RDOP
        DOA     0,22
        DOB     1,22
        DOCP    2,22
        SKPDZ   22
        JMP     -1
        LDA     3,D
        STA     3,.D
S6      MOV     3,3
        DSZ     .D
        JMP     S6
        JMP     @RDOP

.SRTB   STA     3,RET
        MOVZR   1,1
        LDA     0,.TB
        LDA     2,TBI
        ADD     2,0
        STA     0,.TB
        LDA     2,PU
        JSR     DOP
        JSR     .DOP
        LDA     2,PD
        JSR     .DOP
        JSR     DOP
        LDA     1,PU
        JSR     DOR
        LDA     2,PU
        JMP     @RET

SRTC    STA     3,RET
        LDA     0,.TB
        LDA     2,TBI
        ADD     2,0
        STA     0,.TB
        LDA     2,ETB
        ADD     2,1
        LDA     2,PU
        JSR     .DOP
        JSR     .DOR
        LDA     2,PU
        JSR     DOR
        JSR     DOR
        LDA     1,ETB
        JSR     DOP
        LDA     2,PU
        JSR     @RET

DEL.    STA     3,RDEL
        LDA     3,ND
        STA     3,.ND
S7.     LDA     3,D
        STA     3,.D
S6      MOV     3,3
        MOV     3,3
        MOV     3,3
        DSZ     .D
        JMP     S6
        DSZ     .ND
        JMP     S7
        JMP     @RDEL
RET:    0
RDOR.   0
RDEL    0
FTB.    -1000
M.      -760
PD:     1
PU:     0
LTE     0
D:      77777
ETB     -400
.D:     0
ND.     15
.ND     0
TB      -1001
.TB     0
.END
```

```
                        .TITL    PLFG
                        .ENT     SRTC
                        .EXT     CFB,CFE,SRTB,DOR,DEL
                        .EXTD    TBI
                        .ZREL
                        .LOC     140
CFB.     .CFB
CFE.     .CFE
SRTB     .SRTB
SRTC.    .SRTC
DOR      .DOR
DEL      .DEL
                        .NREL
.CFB.    STA     3,RET
         LDA     0,GTB
         LDA     1,D
         LDA     2,PU
         JSR     .DOR
         JSR     .DEL
         LDA     2,PD
         JSR     .DOR
         LDA     1,ATB
         JSR     .DOR
         LDA     [illegible]
         STA     0,.TB
         JMP     @RET

.CFE.    STA     3,RET
         LDA     0,LTB
         LDA     1,ATB
         LDA     2,PU
         JSR     .DOR
         JSR     .DEL
         LDA     2,PD
         JSR     .DOR
         LDA     1,M
         JSR     .DOR
         LDA     2,PU
         JSR     .DOR
         JMP     @RET

.DOR.    STA     3,ADOR
         DOA     0,22
         DOB     1,22
         DOCP    2,22
         SKPBZ   22
         JMP     .-1
         LDA     3,D
         STA     3,.D
S2:      MOV     3,3
         DSZ     .D
         JMP     S2
         JMP     @ADOR

.SRTB:   STA     3,RET
         MOVZR   1,1
         LDA     0,.TB
         LDA     2,TBI
         ADD     2,0
         STA     0,.TB
         LDA     2,PU
         JSR     .DOR
         JSR     .DOR
         LDA     2,PD
         JSR     .DOR
         JSR     .DOR
         LDA     2,PU
         JSR     .DOR
         LDA     2,PU
         JMP     @RET

SRTC.    STA     3,RET
         LDA     0,.TB
         LDA     2,ETB
         ADD     2,1
         LDA     2,PU
         JSR     .DOR
         JSR     .DOR
         LDA     2,PD
         JSR     .DOR
         JSR     .DOR
         LDA     1,ETB
         JSR     .DOR
         LDA     2,PU
         JSR     @RET

.DEL:    STA     3,ADEL
         LDA     3,ND
         STA     3,.ND
S7:      LDA     3,D
         STA     3,.D
S6.      MOV     3,3
         MOV     3,3
         MOV     3,3
         DSZ     .D
         JMP     S6
         DSZ     .ND
         JMP     S7
         JMP     @ADEL
RET      0
ADOR.    0
ADEL     0
ATB:     -1000
M.       -700
PD:      1
PU:      0
LTB:     0
D:       57777
ETB:     -400
.D.      0
ND.      12
GTB:     -200
.ND:     0
TB.      -201
.TB:     0
.END
```

A4.14 The LINK Software

The LINK software[1] links all the subprograms into one program which is loaded into the computer core and run when the name of the program is keyed in. The programs are denoted as follows.

1) The Plotted Harmonics and Errors DP(X)I

2) The Numerical Error Analysis DV(X)I

(X) = 128,256,512,1024

The LINK software lists the subprograms names and core addresses.

```
:
: LINK(DP128I,$VDU,O=DL,S=1000,BSR128:B,CCL128:B..,MCN52:B,DP21:B,DP25:B,DP3X:B,
D04X:B,D05X:B,DP61:B,DP7551:B,PLP5:B,RVO:B,DP125:B,DP143:B,ARTH.B

LINK(DP128I,$VDU,O=DL,S=1000,BSR128:B,CCL128:B,MCN52:B,DP21:B,DP25:B,D
P3X:B,D04X:B,D05X:B,DP61:B,DP7551:B,PLP5:B,RVO:B,DP125:B,DP143:B,ARTH.
B
```

```
---

DDOS 1.6 RELOCATING LINKER VER 1.0 - 05/13/76   DP128I

TITLE   NREL     ZREL     START

O=DL
S=1000
.B128   001000   000050
.CCL5   001667   000050
.MCN4   002170   000050
.P21    002476   000050
.P25    002546   000050
.P31    002576   000050
P32     002662   000050
.P33    003016   000050
.P34    003102   000050
.P35    003166   000050
.P36    003322   000050
.P37    003406   000050
.P38    003472   000050
.P39    003556   000050
.Q41    003642   000050
.Q42    003675   000050
.P43    003730   000050
.Q44    003763   000050
.Q45    004016   000050
.Q46    004051   000050
.Q47    004104   000050
.Q48    004137   000050
.Q49    004177   000050
.A61    004232   000050
.B61    004365   000050
.P62    004524   000050
.B62    004657   000050
.C64    005016   000050
.D64    005247   000050
.E64    005505   000050
.F64    005736   000050
.A63    006173   000050
.B63    006320   000050
.G63    006465   000050
.H63    006620   000050
.I63    006757   000050
.J63    006777   000050
.K63    007176   000050
.L63    007173   000050
.I611   007332   000050
.J61    007352   000050
K612    007511   000050
.I64    007546   000050
.M64    007566   000050
.N64    007655   000050
.O64    007717   000050
.P61    01000[illegible]   000050
P7[illegible]   010[illegible]   000050
```

.PLP5	010475	000050
.RVM	010640	000050
.P125	011776	000050
.P143	012764	000050
.MPY	015001	000050
.DADD	015012	000050
.DMUR	015047	000050
.SMBD	015104	000050
.BDBC	015160	000050
.TTIO	015232	000050
	015272	000050
	024400	

See reference 4.

START ADDR 001000

SYMBOL TABLE

C16	000147	C161	000150	C32	000151	C321	000152
C64	000153	C641	000154	CA1	000274	CA2	000375
CA3	000276	CCAL	000042	CFTST	000273	CPB	000140
CPF	000341	CSR	000113	CT1	000144	CT2	000145
C	000146	DEL	000045	DOP	000[illegible]	GETC	015222
IH1	000050	IH2	000053	IH3	000103	IH4	000054
IH5	000062	IH6	000104	IH7	000055	IH8	000063
IH9	000105	II1	000061	II2	000066	II3	000067
II4	000070	II5	000073	II6	000072	II7	000071
II8	000074	II9	000075	IISH	000043	MB	000044
MB0	000116	MB1	000117	MB10	000130	MB11	000131
MB12	000132	MB13	000133	MB14	000134	MB15	000135
MB16	000136	MB17	000137	MB18	000140	MB19	000141
MB2	000120	MB20	000142	MB21	000143	MB3	000121
MB4	000122	MB5	000123	MB6	000124	MB7	000125
MB8	000126	MB9	000127	MBI	000115	MCON	000160
MSK1	000111	MSK2	000112	N	000114	NH	000155
PUTC	015251	R1 1	002122	R1	000254	RB	000257
SA1	000271	SA2	000272	SA3	000273	SA61	000300
SA62	000302	SA63	000310	SB61	000301	SB62	000303
SB63	000311	SC64	000304	SCNT	011125	SD64	000305
SE64	000306	SF64	000307	SG63	000312	SH1	000045
SH2	000046	SH3	000076	SH4	000047	SH5	000052
SH6	000077	SH63	000313	SH7	000100	SH8	000101
SH9	000102	SI1	000051	SI2	000056	SI3	000057
SI4	000060	SI5	000064	SI6	000065	SI611	000320
SI63	000314	SI64	000[illegible]	SI7	000106	SI8	000107
SI9	000110	SJ61	000321	SJ63	000315	SK612	000322
SK63	000316	SL63	000317	SM64	000[illegible]	SN64	000[illegible]
[illegible]	[illegible]	SP21	000164	SP25	000166	SP211	000200
SP212	[illegible]	SP[illegible]	[illegible]	SP[illegible]	[illegible]	SP[illegible]	[illegible]
SP[illegible]	000205	SP[illegible]	000206	SP[illegible]	000207	SP351	000210
SP352	[illegible]	SP[illegible]	[illegible]	SP[illegible]	[illegible]	SP371	[illegible]
SP372	000215	[illegible]	000216	SP[illegible]	000217	SP291	000220
SP392	[illegible]	SP411	[illegible]	SP41[illegible]	[illegible]	SP421	[illegible]
SP4[illegible]	000225	[illegible]	000226	[illegible]	000227	SP441	000230
SP442	[illegible]	SP451	[illegible]	SP4[illegible]	[illegible]	SP461	[illegible]
SP462	[illegible]	[illegible]71	000226	SP472	[illegible]	SP481	000240
SP482	[illegible]	SP4[illegible]	[illegible]	SP49[illegible]	[illegible]	SP492	[illegible]
SP61	000250	[illegible]	000254	[illegible]	000255	SP62	000251
SP63	000252	SP64	000[illegible]	SP75	000[illegible]	SRTP	000[illegible]
SRTC	000343	STLU	000250	TRI	000123	.BDBC	015160
.CE	000172	.DADD	015012	.DIVI	015[illegible]	.DIVU	01[illegible]
.DMUR	015047	.HS	000170	.IS	000171	.MB	013500
.MPY	015001	.SB	012764	.SMBD	015104	.TLU	011777
.TLUD	011776						

: @P

```
: LINK(DP256I, #VDU, O=DL, S=1000, BSP256:B, CCL256:B, MCN52:B, DP21:B, DP25:B, DP3X:B, (
4X:B, DQSX:B, DP61:B, DP7551.B, PLP5.B, RVO B, DP125.B, DP143:B, ATH:BB.HTRA, ,ARTH:B

RIN[illegible]B, D
8
```

DDOS 1.6 RELOCATING LINKER VER 1.0 - 05/18/76   DP256I

TITLE	NREL	ZREL	START
O=DL			
S=1000			
.B256	001000	000050	
.CCLS	002367	000050	
. Q4	002613	[illegible]	
.P21	003176	[illegible]	
.P25	003246	000050	
.P31	003276	000050	
.P32	003362	000050	
.P33	003516	000050	
.P34	003602	[illegible]	
.P35	003665	000050	
.P36	004022	[illegible]	
.P37	004106	000050	
.P38	004172	000050	
.P39	004236	000050	
.Q41	004342	000050	
.Q42	004375	[illegible]	
.P43	004430	000050	
.Q44	004463	000050	
.Q45	004516	000050	
.Q46	004551	[illegible]	
.Q47	004604	000050	
.Q48	004637	000050	
.Q49	004677	000050	
.A61	004732	[illegible]	
.B61	005065	000050	
.A62	005224	[illegible]	
.B62	005257	000050	
.C64	005310	000050	
.D64	005747	000050	
.E64	006280	[illegible]	
.F64	006426	000050	
.A63	006671	[illegible]	
.B63	007026	[illegible]	
.G63	007165	[illegible]	
.H63	007320	000050	
.I63	007457	000050	
.J63	007477	000050	
.K63	007636	000050	
.L63	007673	000050	
.I611	010072	[illegible]	
.J61	010052	[illegible]	
.K612	010211	000050	
.I64	010240	[illegible]	
.M64	010266	000050	
. 4	010255	[illegible]	
.O64	010412	[illegible]	
.P61	010501	[illegible]	
.P755	011[illegible]	[illegible]	
PLP5	[illegible]	[illegible]	

```
.RVO     01134L  000000
.P125    012002  000000
.P143    013470  000000
.MPY     015005  000050
.DADD    015536  000000
.DSDR    015552  000050
.SMED    015612  000000
.ODEC    015664  000050
.TTIO    015726  000000
         015776  000000
         025107

START ADDR      001000


SYMBOL TABLE

C16   000147    C161  000150    C32   000151    C321  000152
C64   000153    C641  000154    CA1   000274    CA2   000375
CA3   000376    CCAL  000041    CFTST 000273    CP0   000340
CPE   000341    CSR   000113    CT0   000144    CT2   000145
T3   000146    DEL   000345    DO"   000344    GETC  01573V
IH1   0000EP    IH20  @P00US @  IH100 000003 @  IH4   P0005T
IH5   000062    IH6   000104    IH7   P00055    @H80  000061
IH9   0?7405 000IO    000061    II2   00006N    II3   P000N7
II4   000P70  00III   000072    @II6  000061  @ I7   000071
I8   000074    II9   000075    IISH  000043    MB    000044
MB0   000115    MB1   000117    MB10  000120    MB11  000121
MB12  000132    MB13  000133    MB14  000134    MB15  000135
MB16  000136    MB17  000137    MB18  000140    MB19  000141
BEP 0000DLZR @ MB20   000143    MB3 @ 000121
M4   000122    MB5   000123    MB6   000124    MB7   000125
MB21  000111/ 000M7   0P01270  @MI   00P1AI0   MCON 000F160
                  R @ 0000P14     NH000 000151
 UTC  015755    1 1  0026F2 00 "1 @  000F5T @  RB    000257
SAQ  0000371@   SAF   00037F000 53 @ 0POCOS 0000AA  0001000
                                                          '62  000309 00
000310    3B60 0P000000 @ '1J
                                 SB63 0000G10   @SCVL00000EFT @ @SCNT  @11630
4@ 00000U
SF640 0003?004   S?1   000?3?     #B6?  000?12     SH1    00004?
```

```
: ·

: ·
: LINK(DP5121,#VDU,0=DL,S=1000,BSRS12:B,CCLS12:B,MCN52:B,DP21:B,DP25:B,DP3X:B,DV
4X:B,DO5X:B,DP61:B,DP7552.B,PLF6.B,RVO:B,DP125.B,DP143:B,HRTH.B

LINK(DP5121,#VDU,0=DL,S=1000,BSRS12.B,CCLS12.B,MCN52:B,DP21.B,DP25:B,D
P3X.B,DO4X.B,DO5X.B,DP61.B,DP7552.B,PLF6.D,RVO:B,DP125.B,DP143:B,ARTH:
B

---

DDOS 1.6 RELOCATING LINKER VER 1.0 - 05/24/76   DP5121

TITLE   NREL     ZREL     START

0=DL
S=1000
BS12    001000   000050
.CCLS   002602   000050
.MCN4   004033   000050
.P31    004411   000050
.P25    004461   000050
.P21    004511   000050
.P32    004575   000050
.P33    004721   000050
.P34    005015   000050
.P35    005101   000050
.P36    005215   000050
.P37    005321   000050
.P38    005403   000050
.P39    005471   000050
.Q41    005555   000050
.Q42    005610   000050
.P43    005643   000050
.Q44    005676   000050
.Q45    005731   000050
.Q46    005764   [illegible]
.Q47    006017   000050
.Q48    006053   000050
.Q49    006112   000050
.A61    006145   000050
.B61    006200   000050
.A62    006427   000050
.B62    006572   000050
.C64    006721   000050
.D64    007162   000050
.E64    007420   000050
.F64    007651   [illegible]
.A63    010106   000050
.B63    010241   000050
.G63    010400   000050
.H63    010523   000050
.I63    010672   000050
.J63    010722   000050
.K63    011051   [illegible]
.L63    011106   000050
.I611   011245   000050
.J61    011265   000050
.K612   011424   [illegible]
.L64    011461   000050
.M64    011541   [illegible]
[illegible]
```

.064	011625	[illegible]
.P61	011714	[illegible]
P7502	0121[illegible]	[illegible]
.PLF6	0134[illegible]	[illegible]
.RVO	012[illegible]	[illegible]
.P125	013[illegible]	[illegible]
.P143	014[illegible]	[illegible]
.MPY	[illegible]	000050
.DADD	[illegible]	[illegible]
.DSUR	[illegible]	000050
.SMBD	017017	[illegible]
.BDEC	[illegible]	000050
.TTIO	017142	[illegible]
	017202	[illegible]
	026230	

START ADDR 0010[illegible]

SYMBOL TABLE

C16	000147	C161	[illegible]	C32	000151	C321	[illegible]
C64	000153	C641	000154	CA1	000374	CA2	000375
CA3	000376	CCAL	[illegible]	CFTST	[illegible]	CPB	[illegible]
CPE	000341	CSR	000113	CT1	000144	CT2	000145
CT3	000146	DEL	[illegible]	DOR	000344	GETC	017[illegible]
IH1	000050	IH2	000053	IH3	000103	IH4	000054
IH5	000062	IH6	[illegible]	IH7	000055	IH8	[illegible]
IH9	000105	II1	000061	II2	000066	II3	000067
II4	000070	II5	[illegible]	II6	000072	II7	[illegible]
II8	000074	II9	000075	IISH	000043	MB	000044
MB0	000116	MB1	[illegible]	MB10	000120	MB11	[illegible]
MB12	000132	MB13	000133	MB14	000134	MB15	000135
MB16	[illegible]	MB17	[illegible]	MB18	000140	MB19	[illegible]
MB2	[illegible]	MB20	000142	MB21	000143	MB3	000121
MB4	[illegible]	MB5	[illegible]	MB6	000124	MB7	[illegible]
MB8	000126	MB9	000127	MBI	000115	MCON	000160
MSK1	[illegible]	MSK2	[illegible]	N	000114	NH	[illegible]
PUTC	017162	[illegible]	[illegible]	R1	000254	RB	000257
SA1	[illegible]	SA2	[illegible]	SA3	[illegible]	SA61	[illegible]
SA62	[illegible]	SA63	000210	SB61	000201	SB62	000202
SB63	[illegible]	SC64	[illegible]	SCNT	012016	SD64	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	000212	SH1	000045
SH2	[illegible]	SH3	[illegible]	SH4	[illegible]	SH5	[illegible]
SH6	000077	SH62	[illegible]	SH7	000100	SH8	000101
SH9	[illegible]	SI1	[illegible]	SI2	[illegible]	SI3	[illegible]
SI4	[illegible]	SI5	[illegible]	SI6	[illegible]	SI612	000220
SI63	[illegible]	SI64	[illegible]	SI7	[illegible]	SI8	[illegible]
SI9	[illegible]	SJ61	[illegible]	SJ62	[illegible]	[illegible]	000222
SM63	[illegible]	[illegible]	[illegible]	SM64	[illegible]	SN64	[illegible]
SO64	[illegible]	SP21	[illegible]	SP25	000165	SP211	[illegible]
SP212	[illegible]	SP221	[illegible]	SP222	[illegible]	SP231	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
SP352	[illegible]	SP361	[illegible]	SP362	[illegible]	SP371	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	SP391	000220
SP392	[illegible]	SP411	[illegible]	SP412	[illegible]	SP421	[illegible]
[illegible]	000225	[illegible]	[illegible]	[illegible]	[illegible]	SP441	000230
SP442	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	SP461	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	SP62	000251
SP63	[illegible]	SP64	[illegible]	SP75	[illegible]	SRTB	[illegible]
SRTC	[illegible]	[illegible]	[illegible]	TBI	[illegible]	.BDEC	017071
.CS	[illegible]	.DADD	[illegible]	.DIV	[illegible]	.DIVU	[illegible]
[illegible]	[illegible]	.HS	[illegible]	.IS	000171	.MB	014721
.MPY	[illegible]	.SB	[illegible]	.SHDD	[illegible]	.TLU	[illegible]

```
:
: LINK(DP1024I,$VDU,O=DL,S=1000,BSR1024:B,CCL1024.B,MCN52:B,DP21:B,DP25:B,DP3X:B
,OPP04X:B,DOSX:B,DP61:B,DP7552:B,PLP6:B,RVO:B,DP125:B,DP143:B,ARTH:B

LINK(DP1024I,$VDU,O=DL,S=1000,BSR1024:B,CCL1024:B,MCN52:B,DP21:B,DP25:
B,DP3X:B,D04X:B,DOSX:B,DP61:B,DP7552:B,PLP6.B,RVO.B,DP125.B,DP143.B,AR
TH:B
```

```
DDOS 1.6 RELOCATING LINKER VER 1.0 - 05/22/76    DP1024I

TITLE    NREL      ZREL      START

O=DL
S=1000
B1024     001000    000050
.CCL5     006202    000050
.MCN4     006432    000050
.P21      007011    000050
.P25      007061    000050
.P01.     007111    000050
.P32      007175    000050
.P33      007331    000050
.P34      007415    000050
.P35      007501    000050
.P36      007625    000050
.P37      007721    000050
.P38      010005    000050
.P39      010071    000050
.041      010155    000050
.042      010210    000050
.P43      010243    000050
.044      010276    000050
.045      010331    000050
.046      010364    000050
.047      010417    000050
.048      010452    000050
.049      010512    000050
.A61      010545    000050
.B61      010700    000050
.A62      011027    000050
.B62      011172    000050
.C64      011321    000050
.D64      011562    000050
.E64      012070    000050
.F64      012251    000050
.A63      012506    000050
.B63      012641    000050
.G63      013000    000050
.H63      013133    000050
.I63      013272    000050
.J63      013312    000050
.K63      013451    000050
.L63      013506    000050
.I611     013645    000050
.J61      012665    000050
.K612     014004    000050
.I64      014061    000050
.M64      014101    000050
.N64      014170    000050
.064      014[illegible]    000050
.P61      014[illegible]    [illegible]
P[illegible]    014[illegible]    [illegible]
```

.PLP6	015[illegible]0	[illegible]
.RVO	015151	[illegible]
.P125	016[illegible]	[illegible]
.P143	017275	[illegible]
.MPY	021312	000050
.DADD	0213[illegible]	[illegible]
.DSUB	021360	000050
.SMBD	0214[illegible]	[illegible]
.BDEC	021471	000050
.TTIO	0215[illegible]	[illegible]
	021[illegible]	[illegible]
	020730	

START ADDR 001000

SYMBOL TABLE

C16	000147	C161	000150	C32	000151	C321	000153
C64	000152	C641	000154	CA1	000374	CA2	000375
CA3	000376	CCAL	0000[illegible]	CFTST	000[illegible]	CPB	000[illegible]
CPE	000341	CSR	000113	CT1	000144	CT2	000145
CT3	000146	DEL	000[illegible]	DOR	000[illegible]	GETC	0215[illegible]
IH1	000050	IH2	000053	IH3	000103	IH4	000054
I	000[illegible]	IH6	0001[illegible]	IH7	000[illegible]	IH8	000[illegible]
IH[illegible]	000105	II1	000061	II2	000066	II3	000067
II4	000[illegible]	II5	0000[illegible]	II6	000[illegible]	II7	000[illegible]
II8	000074	II9	000075	IISH	000043	MB	000[illegible]
MB0	000116	MB1	000117	MB10	000[illegible]	MB11	000[illegible]
MB12	000132	MB13	000133	MB14	000134	MB15	000135
MB16	000136	MB17	000137	MB18	000140	MB19	000[illegible]
MB2	000120	MB20	000142	MB21	000143	MB3	000121
MB4	000122	MB5	000123	MB6	000124	MB7	000125
MB8	000126	MB9	000127	MBI	000115	MCON	000160
MSK1	000111	MSK2	000112	N	000114	NH	000155
P[illegible]	021562	O1 1	006435	R1	000254	RB	000257
SA1	000371	SA2	000372	SA3	000373	SA61	000[illegible]
SA62	000302	SA63	000310	SB61	000301	[illegible]62	000303
SB63	000311	SC64	000[illegible]	SCNT	01[illegible]	SD64	000[illegible]
[illegible]64	000306	SF64	000307	SG63	000312	SH1	000045
SH2	000[illegible]	SH3	000[illegible]	SH4	000[illegible]	SH5	000[illegible]
SH6	000077	SH63	000[illegible]	SH7	000100	SH8	000101
SH9	000[illegible]	SI1	000[illegible]	SI2	000[illegible]	SI3	000[illegible]
S[illegible]	000060	SI5	000064	SI6	000065	SI611	000320
S[illegible]	000[illegible]	SI64	000[illegible]	SI7	000[illegible]	SI8	000[illegible]
SI9	000110	SJ61	000221	SJ63	000215	S[illegible]12	000222
SK63	000[illegible]	SL[illegible]	000[illegible]	SM64	000[illegible]	SN64	000[illegible]
SO64	000326	SP21	000164	SP25	000166	SP211	000200
SP212	000[illegible]	SP2[illegible]	000[illegible]	SP222	000[illegible]	SP2[illegible]1	000[illegible]
[illegible]	000205	[illegible]	000206	[illegible]	000207	SP151	000210
SP3[illegible]	000[illegible]	SP[illegible]	000[illegible]	SP[illegible]	000[illegible]	SP371	000[illegible]
[illegible]	000215	[illegible]	000216	[illegible]	000217	[illegible]	000220
SP3[illegible]	000[illegible]	SP411	000[illegible]	SP412	000[illegible]	SP421	000[illegible]
[illegible]	000225	[illegible]	000226	[illegible]432	000227	SP441	000230
SP442	000[illegible]	SP451	000[illegible]	SP4[illegible]	000[illegible]	SP461	000[illegible]
[illegible]	000235	[illegible]	000236	[illegible]	000237	[illegible]31	000240
SP482	000241	SP4[illegible]	000[illegible]	SP491	000[illegible]	SP492	000[illegible]
SP61	000250	[illegible]11	000254	[illegible]	000255	SP62	000251
SP63	000[illegible]	SP64	000[illegible]	SP[illegible]	000[illegible]	SRTD	000[illegible]
SRTC	000[illegible]	[illegible]	[illegible]	TBI	000172	.BDEC	021471
.CS	000[illegible]	.DADD	021[illegible]	.DIVI	021[illegible]	.DIVU	021[illegible]
.[illegible]	021360	.HS	000170	.IS	000171	.ME	017211
.MPY	021[illegible]	.SB	017275	.SMUD	021417	.TLU	016210
.TLUD	016[illegible]						

```
: LINK(DV128I,#VDU,O=DL,S=1000,GSR128:B,CCL128:B,DRXX:B

LINK(DV128I,#VDU,O=DL,S=1000,GSR128:B,CCL128:B,DRXX.B


---

DDOS 1.6 RELOCATING LINKER VER 1.0 - 06/22/76    DV128I

TITLE    NREL     ZREL     START

O=DL
S=1000

G128     001000   000050
.CCL5    002267   000050
.MCN4    002620   000050
.P21     003176   000050
.P25     003246   000050
 P31     003276   000050
.P32     003362   000050
.P33     003516   000050
.P34     003602   000050
.P35     003666   000050
.P36     004022   000050
.P37     004106   000050
.P38     004172   000050
.P39     004256   000050
.Q41     004342   000050
.Q42     004375   000050
.P43     004430   000050
.Q44     004463   000050
.Q45     004516   000050
.Q46     004551   000050
.Q47     004604   000050
.Q48     004637   000050
.Q49     004677   000050
 A61     004732   000050
 B61     005065   000050
.A62     005224   000050
.B62     005357   000050
.C64     005516   000050
.D64     005747   000050

.E64     006205   000050
.F64     006436   000050
.A63     006673   000050
.B63     007026   000050
.G63     007165   000050
.H63     007320   000050
.I63     007457   000050
.J63     007477   000050
.K63     007636   000050
.L63     007673   000050
.I611    010032   000050
.J61     010052   000050
.K612    010211   000050
.I64     010246   000050
.M64     010266   000050
.N64     010355   000050
.O64     010412   000050
 F61     010501   000050
```

```
P7552    011065   000050
.RVO     011171   000030
.P125    012327   000030
.P143    013315   000030
.MPY     015322   000050
.DADD    015363   000030
.DSOR    015400   000050
.SMBD    015437   000050
.BDTC    015511   000050
.TTIO    015563   000030
         015623   000050
         025207

START ADDR        001000
```

SYMBOL TABLE

C16	000147	C161	000150	C32	000151	C321	000152
C64	000153	C641	000154	CA1	000374	CA2	000375
CA3	000376	CCAL	000042	CFTST	000273	CSR	000113
CT1	000144	CT2	000145	CT3	000146	GETC	015565
IH1	000050	IH2	000053	IH3	000103	IH4	000054
IH5	000052	IH6	000104	IH7	000055	IH8	000063
IH9	000105	II1	000061	II2	000066	II3	000067
II4	000070	II5	000072	II6	000073	II7	000071
II8	000074	II9	000075	IISH	000043	MB	000044
MB0	000116	MB1	000117	MB10	000130	MB11	000131
MB12	000132	MB13	000133	MB14	000134	MB15	000135
MB16	000136	MB17	000137	MB18	000140	MB19	000141
MB2	000120	MB20	000142	MB21	000143	MB3	000121
MB4	000122	MB5	000123	MB6	000124	MB7	000125
MB8	000126	MB9	000127	MBI	000115	MCON	000160
MSK1	000111	MSK2	000112	N	000114	NH	000155
PUTC	015602	Q1.1	002622	R1	000254	RB	000257
SA1	000371	SA2	000372	SA3	000373	SA61	000300
SA62	000302	SA63	000310	SB61	000301	SB62	000303
SB63	000311	SC64	000304	SCNT	011456	SD64	000305
SE64	000306	SF64	000307	SG63	000312	SH1	000045
SH2	000046	SH3	000076	SH4	000047	SH5	000053
SH6	000077	SH63	000313	SH7	000100	SH8	000101
SH9	000102	SI1	000051	SI2	000056	SI3	000057
SI4	000060	SI5	000064	SI6	000065	SI611	000320
SI63	000314	SI64	000333	SI7	000106	SI8	000107
SI9	000110	SJ61	000321	SJ63	000315	SK612	000322
SK63	000316	SL63	000317	SM64	000324	SN64	000325
SO64	000326	SP21	000164	SP25	000166	SP311	000200
SP312	000201	SP321	000202	SP322	000203	SP331	000204
SP332	000205	SP341	000206	SP342	000207	SP351	000210
SP352	000211	SP361	000212	SP362	000213	SP371	000214
SP372	000215	SP381	000216	SP382	000217	SP391	000220
SP392	000221	SP411	000222	SP412	000223	SP421	000224
SP422	000225	SP431	000226	SP432	000227	SP441	000230
SP442	000231	SP451	000232	SP452	000233	SP461	000234
SP462	000235	SP471	000236	SP472	000237	SP481	000240
SP482	000241	SP483	000242	SP491	000243	SP492	000244
SP61	000350	SP611	000354	SP612	000355	SP62	000351
SP63	000352	SP64	000353	SP75	000165	STLU	000250
TBI	000173	.BDTC	015511	.C5	000172	.DADD	015363
.DIVI	015417	.DIVU	015414	.DSOR	015400	.H5	000170
.I5	000171	.MB	013331	.MPY	015322	.S6	013315
.SMBD	015437	TLU	012330	TLUD	012327		

A4.15. The Stored Table of Values STV

The format of the block is described in section A 3.10.2

The data is given in decimal form but the Assembler software converts this to octal code.

```
.TITL   .F123
.ENT     TLUG  TLU
.NFIL
.RDX    10
TLUG.   0
```

.TLU.	0	0	0	0	0	0
1	1	5	8	10	11	12
3	8	13	19	24	25	30
5	17	27	37	45	51	56
9	27	45	61	74	84	92
14	41	70	94	113	131	142
21	60	103	142	173	197	214
33	89	155	209	255	281	315
47	129	226	303	371	414	460
67	184	325	433	528	611	663
97	266	465	620	752	876	931
139	411	663	900	1058	1251	1337
197	586	952	1281	1561	1761	1922
281	833	1352	1820	2218	2531	2746
398	1182	1915	2582	3147	3571	3894
565	1675	2720	3661	4461	5090	5522
801	2373	3854	5186	6320	7210	7827
1134	3360	5452	7343	8948	10208	1107[illegible]
1603	4759	[illegible]	10391	12663	14448	156[illegible]
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
2	5	7	9	11	12	13
6	12	17	22	25	25	31
11	22	32	41	48	54	57
18	36	53	65	75	88	94
29	57	82	105	124	137	146
43	85	124	153	186	207	220
64	126	183	221	274	305	324
93	181	267	340	393	444	471
135	265	385	490	576	640	680
193	380	552	701	826	918	974
275	542	788	10[illegible]	1179	1310	1381
394	772	1121	14[illegible]	1675	1855	1991
558	10[illegible]	1594	2023	2365	2651	2814
794	1525	2263	2878	3359	3761	3977
1126	2208	3206	4071	4754	5322	5662
1593	3128	4542	5761	6738	7533	8017
2255	4428	6431	81[illegible]	9534	10654	1132[illegible]
3196	6251	9103	11505	13622	15136	1606[illegible]
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

```
0
11
21
53
90
142
211
319
478
630
969
1411
2010
2856
4032
5744
8176
11520
16183
0
0
0
0
0
0
0
0
0
0
0
0
0
0
17
32
55
95
148
224
329
480
681
981
1418
2013
2805
4071
5771
8173
11575
16254
0
0
END
```

A5 The Fast Fourier Transform (FFT)

The input and output specifications and operating instructions are given in the listing at the end of this chapter. A discussion on the errors is given in reference 18.

A5.1 The Scale Factors MAN and SCNT

The FFT is programmed to allow the harmonics to have the maximum machine code values but not to overflow. Thus the harmonics are represented at maximum resolution but the scale is not compatible with that of the DFTTLU. If the FFT analyses a maximum dc level (Fig. A5.1) ie the ADC output is 2047_{10}, the harmonic G_0 always has a machine code value equivalent to 32768_{10}. The value of SCNT is dependent on N and is given in Table A5.1. These are the maximum values of SCNT and are termed MAN.

Table A5.2 lists the value of MAN where the input signal is time varying and the maximum value of the harmonics, $\frac{N}{2}$ has the value equivalent to 16384_{10}, 8192_{10} etc.

TABLE A5.1

N	MAN (MAX. SCNT)
128	3
256	4
512	5
1024	6

Table A 5.2 MAN for different N and $(\frac{N}{2})$

MAN	128	256	512	1024
3	16384			
4	8192	16384		
5	4096	8192	16384	
6	2048	4096	8192	16384
7	1024	2048	4096	8192
8		1024	2048	4096
9			1024	2048

When a noiselike signal is analysed, the maximum value of the harmonic is set by the corresponding value of MAN. eg for $N = 1024$ and $(\frac{N}{2}) = 8192$, MAN = 7. SCNT can be less than or equal to MAN depending on the amplitude of the signal. If SCNT is less than MAN, the harmonics are NOT to the scale and the scale must be adjusted by the difference between MAN and SCNT ie. All harmonics must be shifted by MAN-SCNT times to the right.

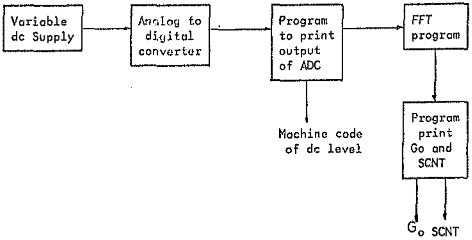


Fig.A5.1

A5.2 Discussion on the Accuracy of a Fixed Point FFT

In reference 18, Welch presents a fixed point FFT error analysis which compares the results of a fixed point FFT with those of a floating point FFT. The floating point FFT uses a word with a 27 bit mantissa and is considered to be the 'accurate' answer. The analysis results in an expression for the upper bound error which is confirmed with experimental results.

The expression for the upper bound error is

$$\frac{\text{rms (error)}}{\text{rms (result)}} \leqslant \frac{2^{(M+3)/2}\ 2^{-B}\ \ C}{\text{rms (initial sequence)}} \qquad \text{A5.1}$$

$M = \log_2 N$

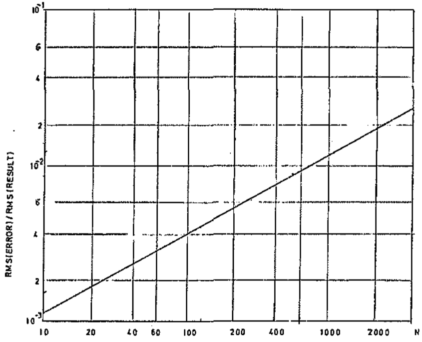
B = number of bits in the fixed point word excluding the sign bit.

= 15 (Nova 1200)

C = constant

= 0,9 (for 2's complement machine with truncation).

The signal used in the error analysis described has a ptp value of ± 1 and an rms value of 0,2. The variation of rms(error)/rms(result) with N is given in fig. A5.2.



B = 15
C = 0,9
Input Signal. Random Numbers between -1 and +1, rms value = 0,2

Fig. A.5.2 Theoretical Upper Bound of Error

NOVA FAST FOURIER TRANSFORM CALCULATION
****

DEPARTMENT OF ELECTRICAL ENGINEERING

UNIVERSITY OF THE WITWATERSRAND

APRIL 1972

THE DISCRETE TRANSFORM(DFT) OF A COMPLEX SERIES F(N*T) IS

G((K*Ω)) = SUM(((F((N*T))*EXP(-J*Ω*T*N*K))))

WHERE THE SUM IS FROM N=0 TO N=NUM-1, AND Ω=2*PI/(NUM*T)

THE INVERSE TRANSFORM (IDFT) IS

F((L*T)) = (1/NUM)*SUM(((G((K*Ω))*EXP(J*Ω*T*N*K))))

AND THE SUM IS OVER K FROM 0 TO NUM-1

THE FACTOR (1/NUM) IS NOT INCLUDED IN THIS PROGRAM.

DEFINITIONS:

NUM IS THE NUMBER OF SAMPLE POINTS. IT MUST BE AN INTEGER POWER OF 2.
M IS THE INTEGER POWER OF 2, THUS NUM=2**M.

N IS AN INDEX INDICATING THE ELEMENTS OF THE INPUT SERIES. THUS
X(0), X(1), ..., X(N), ..., X(NUM-1) FORM THE INPUT SERIES X.

I IS AN INDEX INDICATING THE PARTIAL SUMS AS THE DFT IS CALCULATED.
THUS I=0 INDICATES THE INPUT ARRAY, I=M INDICATES THE DFT WITH ITS
ELEMENTS IN BIT REVERSE ORDER.

CALLING SEQUENCE:

```
                  .LOC  005
JSR @5; LOC 5 CONTAINS STARTING ADDRESS OF DFT PROGRAM (7102)
A:  <0 OR 1>     0 FOR  FORWARD TRANSFORM,    1 FOR  REVERSE.
    <NUM>
    <X(0)>       REAL PART OF INPUT SERIES.
      .
      .          THIS TAKES NUM LOCATIONS
      .
    <X(NUM-1)>
    <Y(0)>       IMAGINARY PART OF INPUT SERIES
      .
      .          THIS ALSO TAKES NUM LOCATIONS
      .
    <Y(NUM-1)>
    <CONTINUE MAIN PROGRAM HERE>    THE ADDRESS OF THIS LOCATION
                                    IS A + 2*NUM + 2
```

THE TRANSFORMED DATA WILL HAVE REPLACED THE INPUT DATA WHEN CONTROL
RETURNS FROM THIS SUBROUTINE. SEE SCNT FOR SCALING OF OUTPUT DATA.

```
NOTES:

SCALING.  TO AVOID OVERFLOW EVERY ARRAY EXCEPT THE FIRST IS DIVIDED
          BY 2 IF IT HAS ANY ELEMENT LARGER THAN 31463 (OCTAL).
          THE CONDITION IS NOT NECESSARY ON A 16 BIT MACHINE.

CAUTION:  INPUT ARRAY DATA MUST BE LESS THAN 2**14 (DECIMAL) TO
          AVOID OVERFLOW ON THE I=1 ARRAY.

SCNT.     THE NUMBER OF TIMES THE DATA HAS BEEN SCALED IS STORED
IN LOC 7265.  THE OUTPUT SERIES MUST BE MULTIPLIED BY 2**<SCNT>
FOR THE FORWARD TRANSFORM AND 2**<SCNT -1> FOR THE INVERSE.

FIXED POINT. ALL SINES AND COSINES ARE MULTIPLIED BY 2**14 (DEC),
          ALL ARITHMETIC IS INTEGER MODE AND ALL DATA ARE ASSUMED
          INTEGER.

MAXIMUM LENGTH.  THE INPUT SERIES IS LIMITED TO 1024 (DECIMAL)
          COMPLEX VALUES. A REAL SERIES OF DOUBLE LENGTH CAN BE
          TRANSFORMED FOR BROAD BAND SPECTRAL PURPOSES BY TREATING
          IT AS A COMPLEX SERIES OF HALF LENGTH.

COSINE ARRAY.  LOCATIONS 6377 THROUGH 6777 CONTAIN ALL THE COSINES
          NECESSARY TO COMPUTE THE SINUSOIDAL FACTORS OF
          EXP(2*PI*N*K/NUM). THEY ARE IN A FORMAT WHEREBY 1.0 IS
          REPRESENTED BY 40000 (OCTAL)

REENTRY.  INITIAL VALUES FOR SUBROUTINE ARE ALL AUTOMATICALLY
          COMPUTED ON ENTRY. HENCE SUBROUTINE IS POTENTIALLY
          REENTRANT. IT DOES NOT NOW PROVIDE FOR SAVING OF AC'S
          AND C.


*U*U*U*U*
```

```
            000002 .LOC 02
--- 00002 000000 NUM: 0
    00003 000000 ALPHA: 0


    00004 000000 M: 0
    00005 007102 CFTST: FFTST

    00006 000000 FLAG: 0
    00007 000000 NUM4: 0

            007000 .LOC 7000
    07000 020424 FFTRA: LDA 0,SPF
    07001 101235 MOVZR# 0,0,SNR
    07002 002530 JMP @GBMP
    07003 030526 LDA 2,BETA
    07004 034417 LDA 3,MAX4
    07005 024007 LDA 1,NUM4
    07006 133000 ADD 1,2
    07007 151400 INC 2,2
    07010 024414 LDA 1,SPF
    07011 050030 STA 2,30
    07012 050031 STA 2,31
    07013 021400 LDA 0,0,3
    07014 032030 LDA 2,@30
    07015 042031 STA 0,@31
    07016 051400 STA 2,0,3
    07017 136400 SUB 1,3
    07020 014007 DSZ NUM4
    07021 000772 JMP .-6
    07022 002510 JMP @GBMP
    07023 006777 MAX4: 6777
    07024 000000 SPF: 0
    07025 007404 GFFIN: FFTNC

    07026 054407 GETDA: STA 3,BACK
    07027 133000 ADD 1,2
    07030 052020 STA 2,@20
    07031 021000 LDA 0,0,2
    07032 004443 JSR FFTSC
    07033 042021 STA 0,@21
    07034 002401 JMP @BACK


    07035 000000 BACK: 0


    07036 101132 TEST: MOVZL# 0,0,SZC
    07037 101241 MOVOR 0,0,SKP
    07040 101220 MOVZR 0,0
    07041 001400 JMP 0,3


    07042 034467 FFTCC: LDA 3,BETA
    07043 054020 STA 3,20
    07044 054021 STA 3,21
    07045 020426 LDA 0,MM
```

```
07046 024004 LDA 1,4
07047 123000 ADD 1,0
07050 126520 SUBZL 1,1
07051 101404 INC 0,0,SZR
07052 125121 MOVZL 1,1,SKP
07053 101001 MOV 0,0,SKP
07054 000775 JMP .-3
07055 044747 STA 1,SPF
07056 020002 LDA 0,NUM
07057 101220 MOVZR 0,0
07060 101220 MOVZR 0,0
07061 040007 STA 0,NUM4
07062 040753 STA 0,BACK

07063 137000 ADD 1,3
07064 021400 LDA 0,0,3
07065 032020 LDA 2,020.
07066 042021 STA 0,021
07067 051400 STA 2,0,3
07070 014745 DSZ BACK
07071 000772 JMP .-6
07072 002733 JMP @GFFIN

07073 177765 MM: -13
07074 063077 FACFI: HALT




07075 024562 FFTSG: LDA 1,ATF+1
07076 125014 MOV# 1,1,SZR
07077 000737 JMP TEST

07100 001400 JMP 0,3
07101 063077 HALT

07102 021400 FFTST: LDA 0,0,3
07103 040006 STA 0,FLAG
07104 021401 LDA 0,1,3
07105 040002 STA 0,NUM
07106 126440 SUBO 1,1
07107 101223 MOVZR 0,0,SNC
07110 125401 INC 1,1,SKP
07111 000402 JMP .+2
07112 000775 JMP .-3

07113 044004 STA 1,4
07114 175400 INC 3,3
07115 175400 INC 3,3
07116 054003 STA 3,ALPHA
07117 020002 LDA 0,NUM
07120 105120 MOVZL 0,1
07121 167000 ADD 3,1
```

```
07122 044410 STA 1,GTMP
07123 126400 SUB 1,1
07124 044532 STA 1,ATF
07125 044532 STA 1,ATF+1
07126 044537 STA 1,SCNT
07127 044675 STA 1,SPF
07130 000712 JMP FFICC

07131 006377 GETA: 6377
07132 000000 GTMP: 0


                                    ;PAGES 1&2


07133 054522 CALC: STA 3,RETN
07134 024576 LDA 1,N
07135 004531 JSR FFTAG
07136 030512 LDA 2,SAP
07137 050020 STA 2,20
07140 030520 LDA 2,TAM
07141 050021 STA 2,21
07142 030003 LDA 2,ALPHA
07143 024567 LDA 1,N
07144 004662 JSR GETDA
07145 024002 LDA 1,NUM
07146 004660 JSR GETDA
07147 024564 LDA 1,A
07150 004656 JSR GETDA
07151 024002 LDA 1,NUM
07152 124400 NEG 1,1
07153 004653 JSR GETDA
                              ;PAGE 2

07154 024472 LDA 1,COS
07155 004557 JSR FFINT
07156 131000 MOV 1,2
07157 024505 LDA 1,TEMP3
07160 034554 JSR FFINT
07161 004561 JSR MULT
07162 042472 STA 0,QSAV3
07163 024464 LDA 1,SIN
07164 004550 JSR FFINT
07165 131000 MOV 1,2
07166 024475 LDA 1,TEMP4
07167 004545 JSR FFINT
07170 004552 JSR MULT
07171 032463 LDA 2,QSAV3
07172 113000 ADD 0,2
07173 024466 LDA 1,TEMP1
07174 140400 NEG 2,0
07175 133000 ADD 1,2
07176 123000 ADD 1,0
07177 052452 STA 2,QSAV1
07200 042454 STA 0,QSAV3
07201 004565 JSR FFTAT
```

```
07202 010454 ISZ AIF
07203 101000 MOV 0,0
07204 111000 MOV 0,2
07205 004561 JSR FFIAT
07206 010450 ISZ AIF
07207 101000 MOV 0,0
07210 024436 LDA 1,COS
07211 004523 JSR FFINT
07212 131000 MOV 1,2
07213 024450 LDA 1,TEMP4
07214 004520 JSR FFINT
07215 004525 JSR MULT
07216 042435 STA 0,@SAV4
07217 024430 LDA 1,SIN
07220 124400 NEG 1,1
07221 004513 JSR FFINT
07222 131000 MOV 1,2
07223 024441 LDA 1,TEMP3
07224 004510 JSR FFINT
07225 004515 JSR MULT
07226 032425 LDA 2,@SAV4
07227 113000 ADD 0,2
07230 140400 NEG 2,0
07231 024431 LDA 1,TEMP2
07232 133000 ADD 1,2
07233 123000 ADD 1,0
07234 052416 STA 2,@SAV2
07235 042416 STA 0,@SAV4
07236 004530 JSR FFIAT
07237 010417 ISZ AIF
07240 101000 MOV 0,0
07241 111000 MOV 0,2
07242 004524 JSR FFIAT
07243 010413 ISZ AIF
07244 101000 MOV 0,0
07245 002416 JMP @RETN
```

```
07246 000000 COS : 0
07247 000000 SIN: 0
07250 007250 SAP: SAP
07251 000000 SAV1: 0
07252 000000 SAV2: 0
07253 000000 SAV4: 0
07254 000000 SAV3: 0
07255 000000 RETN: 0
07256 000000 AIF: 0
07257 000000 AIFC: 0
07260 007260 IAM: IAM
07261 000000 TEMP1: 0
07262 000000 TEMP2: 0
07263 000000 TEMP4: 0
07264 000000 TEMP3: 0

07265 000000 SCNT: 0
```

```
07266 054476 FFTAG: STA 3,MSAV
07267 020573 LDA 0,I
07270 030004 LDA 2,M
07271 142405 SUB 2,0,SNR
07272 000404 JMP MID
07273 125220 MOVZR 1,1
07274 101404 INC 0,0,SZR
07275 000776 JMP .-2
07276 004565 MID: JSR FFTAV

07277 020007 LDA 0,NUM4
07300 111000 MOV 0,2
07301 122422 SUBZ 1,0,SZC
07302 000407 JMP .+7
07303 143000 ADD 2,0
07304 030625 LDA 2,BETA
07305 113000 ADD 0,2
07306 021000 LDA 0,0,2
07307 100400 NEG 0,0
07310 000404 JMP .+4
07311 030620 LDA 2,BETA
07312 133000 ADD 1,2
07313 021000 LDA 0,0,2
07314 040732 STA 0,COS
07315 020007 LDA 0,NUM4
07316 122032 ADCZ# 1,0,SZC
07317 122401 SUB 1,0,SKP
07320 106401 SUB 0,1,SKP
07321 105000 MOV 0,1

07322 030607 LDA 2,BETA
07323 133000 ADD 1,2
07324 021000 LDA 0,0,2
07325 030006 LDA 2,FLAG
07326 151232 MOVZR# 2,2,SZC
07327 100400 NEG 0,0
07330 040717 STA 0,SIN
07331 002433 JMP @MSAV
07332 000000 N: 0
07333 000000 A: 0

07334 125133 FFINT: MOVZL# 1,1,SNC
07335 001400 JMP 0,3
07336 124400 NEG 1,1
07337 010402 ISZ ASF
07340 001400 JMP 0,3

07341 000000 ASF: 0
```

```
07342 102460 MULT: SUBC 0,0
07343 054421 STA 3,MSAV
07344 034421 LDA 3,M20
07345 125203 MLOOP: MOVR 1,1,SNC
07346 101201 MOVR 0,0,SKP
07347 143220 ADDZR 2,0
07350 175404 INC 3,3,SZR
07351 000774 JMP MLOOP
07352 101160 MOVCL 0,0
07353 125120 MOVZL 1,1
07354 101100 MOVL 0,0


07355 024764 LDA 1,ASF
07356 125233 MOVZR# 1,1,SNC
07357 002405 JMP @MSAV
07360 100400 NEG 0,0
07361 126400 SUB 1,1
07362 044757 STA 1,ASF
07363 002401 JMP @MSAV
07364 000000 MSAV: 0
07365 177760 M20: -20



07366 014670 FFTAT: DSZ ATF
07367 010667 ISZ ATF
07370 001400 JMP 0,3
                                ;THE EST
07371 024412 LDA 1,CUTO
07372 151132 MOVZL# 2,2,SZC
07373 000404 JMP NEGA
07374 132032 ADCZ# 1,2,SZC
07375 001400 JMP 0,3
07376 001401 JMP 1,3
07377 124400 NEGA: NEG 1,1
07400 132032 ADCZ# 1,2,SZC

07401 001401 JMP 1,3
07402 001400 JMP 0,3

07403 031463 CUTO: 31463


07404 024004 FFINC: LDA 1,M
07405 044452 STA 1,ACNT
07406 102520 SUBZL 0,0
07407 040453 STA 0,I
```

```
07410 126400 NUDS: SUB 1,1
07411 044721 STA 1,N


07412 020450 AGET: LDA 0,I
07413 030002 LDA 2,NUM
07414 100400 NEG 0,0
07415 151220 MOVZR 2,2
07416 101404 INC 0,0,SZR
07417 000776 JMP .-2
07420 050713 STA 2,A
07421 050437 STA 2,NCNT


07422 006437 ROAD: JSR @GCALC
07423 014435 DSZ NCNT
07424 000427 JMP  NIND
07425 010705 ISZ N
07426 030705 LDA 2,A
07427 050431 STA 2,NCNT
07430 024702 LDA 1,N
07431 147000 ADD 2,1
07432 044700 STA 1,N
07433 030002 LDA 2,NUM
07434 132404 SUB 1,2,SZR
07435 000765 JMP ROAD
07436 014421 DSZ ACNT
07437 000416 JMP AIND
07440 000434 JMP OUT


07441 020615 STSR: LDA 0,ATF
07442 101134 MOVZL# 0,0,SZR
07443 000403 JMP SCALE
07444 040613 STA 0,ATF+1
07445 000743 JMP NUDS


07446 010611 SCALE: ISZ ATF+1
07447 010616 ISZ SCNT
07450 102400 SUB 0,0
07451 040605 STA 0,ATF
07452 000736 JMP NUDS


07453 010657 NIND: ISZ N
07454 000746 JMP ROAD


07455 010405 AIND: ISZ I
07456 000763 JMP STSR
```

```
07457 000000 ACNT: 0
07460 000000 NCNT: 0
07461 007133 GCALC: CALC
07462 000000 I: 0
07463 030004 FFTRV: LDA 2,M
07464 121000 MOV 1,0
07465 126400 SUB 1,1
07466 150400 NEG 2,2
07467 101220 MOVZR 0,0
07470 125100 MOVL 1,1
07471 151404 INC 2,2,SZR
07472 000775 JMP .-3
07473 001400 JMP 0,3


07474 024407 OUT: LDA 1,PHASE
07475 044020 STA 1,20
07476 024003 LDA 1,ALPHA
07477 044433 STA 1,ALPHB
07500 101001 MOV 0,0,SKP
07501 002433 JMP @AFFRA
07502 126400 SUB 1,1
07503 044627 STA 1,N
07504 024002 LDA 1,NUM
07505 044753 STA 1,NCNT

07506 024624 CYCLE: LDA 1,N
07507 004754 JSR FFTRV

07510 030422 LDA 2,ALPHB

07511 034621 LDA 3,N'
07512 136432 SUBZ# 1,3,SZC
07513 000407 JMP EQUAL
07514 157000 ADD 2,3
07515 133000 ADD 1,2
07516 025000 LDA 1,0,2
07517 021400 LDA 0,0,3
07520 041000 STA 0,0,2
07521 045400 STA 1,0,3


07522 010610 EQUAL: ISZ N
07523 014735 DSZ NCNT
07524 000762 JMP CYCLE
07525 030405 LDA 2,ALPHB
07526 024002 LDA 1,NUM
07527 133000 ADD 1,2
07530 050402 STA 2,ALPHB
07531 002020 JMP @20
07532 000000 ALPHB: 0

07533 007477 PHASE: OUT+3
```

07534 007000 GFFKAI FFIKA


```
       006377 .LOC 6377
06377 040000 040000
06400 040000 040000
06401 037777 037777
06402 037775 037775
06403 037773 037773
06404 037770 037770
06405 037765 037765
06406 037761 037761
06407 037754 037754
06410 037747 037747
06411 037741 037741
06412 037733 037733
06413 037724 037724
06414 037714 037714
06415 037704 037704
06416 037673 037673
06417 037661 037661
06420 037647 037647
06421 037634 037634
06422 037621 037621
06423 037605 037605
06424 037570 037570
06425 037553 037553
06426 037535 037535
06427 037517 037517
06430 037500 037500
06431 037460 037460
06432 037440 037440
06433 037417 037417
06434 037375 037375
06435 037353 037353
06436 037330 037330
06437 037305 037305
06440 037261 037261
06441 037235 037235
06442 037210 037210
06443 037162 037162
06444 037134 037134
06445 037105 037105
06446 037055 037055
06447 037025 037025
06450 036774 036774
06451 036743 036743
06452 036711 036711
06453 036657 036657
06454 036623 036623
06455 036570 036570
06456 036533 036533
06457 036477 036477
06460 036441 036441
06461 036403 036403
06462 036344 036344
06463 036305 036305
06464 036245 036245
06465 036205 036205
```

06466	036144	036144
06467	036102	036102
06470	036040	036040
06471	035775	035775
06472	035732	035732
06473	035666	035666
06474	035622	035622
06475	035555	035555
06476	035507	035507
06477	035441	035441
06500	035372	035372
06501	035323	035323
06502	035253	035253
06503	035202	035202
06504	035131	035131
06505	035060	035060
06506	035006	035006
06507	034733	034733
06510	034660	034660
06511	034604	034604
06512	034530	034530
06513	034453	034453
06514	034375	034375
06515	034317	034317
06516	034241	034241
06517	034161	034161
06520	034102	034102
06521	034022	034022
06522	033741	033741
06523	033660	033660
06524	033576	033576
06525	033513	033513
06526	033430	033430
06527	033345	033345
06530	033261	033261
06531	033175	033175
06532	033110	033110
06533	033022	033022
06534	032734	032734
06535	032645	032645
06536	032556	032556
06537	032467	032467
06540	032377	032377
06541	032306	032306
06542	032215	032215
06543	032123	032123
06544	032031	032031
06545	031737	031737
06546	031643	031643
06547	031550	031550
06550	031454	031454
06551	031357	031357
06552	031262	031262
06553	031164	031164
06554	031066	031066
06555	030770	030770
06556	030671	030671
06557	030571	030571
06560	030471	030471
06561	030371	030371

06562	030270	030270
06563	030166	030166
06564	030064	030064
06565	027762	027762
06566	027657	027657
06567	027554	027554
06570	027450	027450
06571	027344	027344
06572	027237	027237
06573	027132	027132
06574	027025	027025
06575	026717	026717
06576	026610	026610
06577	026501	026501
06600	026372	026372
06601	026262	026262
06602	026152	026152
06603	026041	026041
06604	025730	025730
06605	025617	025617
06606	025505	025505
06607	025373	025373
06610	025260	025260
06611	025145	025145
06612	025032	025032
06613	024716	024716
06614	024601	024601
06615	024465	024465
06616	024347	024347
06617	024232	024232
06620	024114	024114
06621	023776	023776
06622	023657	023657
06623	023540	023540
06624	023421	023421
06625	023301	023301
06626	023161	023161
06627	023040	023040
06630	022717	022717
06631	022576	022576
06632	022454	022454
06633	022332	022332
06634	022210	022210
06635	022065	022065
06636	021742	021742
06637	021616	021616
06640	021473	021473
06641	021347	021347
06642	021222	021222
06643	021075	021075
06644	020750	020750
06645	020623	020623
06646	020475	020475
06647	020347	020347
06650	020221	020221
06651	020072	020072
06652	017743	017743
06653	017614	017614
06654	017464	017464
06655	017334	017334

06656	017204	017204
06657	017053	017053
06660	016723	016723
06661	016571	016571
06662	016440	016440
06663	016306	016306
06664	016154	016154
06665	016022	016022
06666	015670	015670
06667	015535	015535
06670	015402	015402
06671	015247	015247
06672	015113	015113
06673	014757	014757
06674	014623	014623
06675	014467	014467
06676	014333	014333
06677	014176	014176
06700	014041	014041
06701	013704	013704
06702	013546	013546
06703	013411	013411
06704	013253	013253
06705	013114	013114
06706	012756	012756
06707	012620	012620
06710	012461	012461
06711	012322	012322
06712	012163	012163
06713	012023	012023
06714	011664	011664
06715	011524	011524
06716	011364	011364
06717	011224	011224
06720	011064	011064
06721	010723	010723
06722	010563	010563
06723	010422	010422
06724	010261	010261
06725	010120	010120
06726	007756	007756
06727	007615	007615
06730	007453	007453
06731	007312	007312
06732	007150	007150
06733	007006	007006
06734	006644	006644
06735	006501	006501
06736	006337	006337
06737	006174	006174
06740	006032	006032
06741	005667	005667
06742	005524	005524
06743	005361	005361
06744	005216	005216
06745	005053	005053
06746	004707	004707
06747	004544	004544
06750	004401	004401
06751	004235	004235

```
06752 004071 004071
06753 003726 003726
06754 003562 003562
06755 003416 003416
06756 003252 003252
06757 003106 003106
06760 002742 002742
06761 002576 002576
06762 002432 002432
06763 002265 002265
06764 002121 002121
06765 001755 001755
06766 001610 001610
06767 001444 001444
06770 001300 001300
06771 001133 001133
06772 000767 000767
06773 000622 000622
06774 000456 000456
06775 000311 000311
06776 000145 000145
06777 000000 000000
                   .END
```

```
A        007333
ACNT     007457
AGET     007412
AIND     007455
ALPHA    000003
ALPHB    007532
ASF      007341
ATF      007256
ATFC     007257
BACK     007035
BETA     007131
CALC     007133
CFTST    000005
COS      007246
CUTO     007403
CYCLE    007506
EQUAL    007522
FACPT    007074
FFTAG    007266
FFTAT    007366
FFTCC    007042
FFTNC    007404
FFTNT    007334
FFTRA    007000
FFTRV    007463
FFTSC    007075
FFTST    007102
FLAG     000006
GBMP     007132
GCALC    007461
GETDA    007026
GFFRA    007534
GFFTN    007025
I        007462
M        000004
M20      007365
MAX4     007023
MID      007276
MLOOP    007345
MM       007073
MSAV     007364
MULT     007342
N        007332
NCNT     007460
NEGA     007377
NIND     007453
NUDS     007410
NUM      000002
NUM4     000007
OUT      007474
PHASE    007533
RETN     007255
ROAD     007422
SAP      007250
SAV1     007251
SAV2     007252
SAV3     007254
SAV4     007253
SCALE    007446
SCNT     007265
```

```
SIN     007247
SPF     007024
STSR    007441
TAM     007260
TEMP1   007261
TEMP2   007262
TEMP3   007264
TEMP4   007263
TEST    007036
```

A6 System t Perform Error Tests

Fig.A6.1 shows the error test system block diagram. The system can be divided into 2 parts.

1) The Hardware ie the noise generators, amplifiers etc.

2) The Software ie. the FFT and DFTTLU programs.

A6.1 The Hardware

The hardware consists of a noise generator, filter set, amplifier and attenuator . The noise generator has a frequency range of 20Hz to 20KHz at a voltage of 1 volt peak to peak. The noise, filtered by the filter set, is amplified to approximately 10V peak to peak and is fed to three inputs.

1) The ADC, used in conjunction with the FFT.

2) A true r.m.s. voltmeter.

3) An attenuator set to approximately 12dB which attenuates to signal to about 2.5V peak to peak. This signal is fed to the LADC and is processed by the DFTTLU.

4) An oscillator controlled pulse generator controls the sampling rate which is set at 20,48 KHz.

A6.2 Notes on the System

1) The filter set allows a variety of noise signals to be generated ie bandpass, band reject and low pass wide and narrow band signals.

2) The oscillator sets the sampling rate of the system. For a sampling rate of 20,48 KHz the following integral multiples of frequencies can be extracted.

N	freq (Hz)
128	160
256	80
512	40
1024	20

3) Noise generator is set so that rms value of noise at the 12 bit A/D is 2 volts. (Peak value of noise generator output is 5 times its rms value)

4) The LADC and ADC are peripherals of the Nova 1200 but the sampling rate in this case is controlled externally

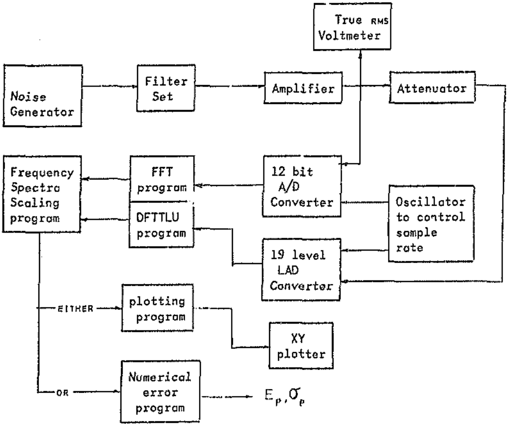


Fig.A6.1 Block Diagram of System to Measure Errors

A6.3 Scaling the Input Signals

The ADC has a peak to peak (ptp) voltage saving of approximately -10V to +10V. The binary output for the maximum positive level and maximum negative level is equivalent to 2047_{10} and -2048_{10} respectively. The LADC has a ptp voltage swing of approximately $\pm 2,5$V. Sign magnitude notation is used and the binary output is equivalent to $\pm 18_{10}$. The attenuator must be set so that a peak signal gives maximum output at both analog to digital convertors. The attenuator in Fig.A6.2 is set to a nominal value of 12dB and a short program is written to print the output of both analog to digital convertors. The dc supply is adjusted until the printer shows that the ADC output is equivalent to 2047_{10}. The attenuation is then adjusted until the printer shows that the LADC output is equivalent to 18_{10}. The value of the attenuation is 11,7dB.

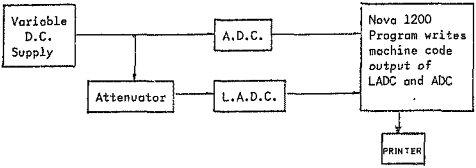


Fig.A6.2 System to calibrate the FFT and DFTTLU

A6.4 Specifications and Interconnection Diagram of Error Test System

Instrument				Store Number
1.	Noi enerator:		Rohde and Schwarz Type SVF BN 4150 30 Hz - 6MHz	
2.	Filter Set:	(1)	Bruel and Kjaer. Filter Set Type. 1614.	AL41
		(2)	Krohn - Hite Model 3550 2Hz - 200KHz	
3.	Amplifier:		Bruel and Kjaer Measuring Amplifier Type 2609 20Hz - 20 000 Hz	
4.	Attenuator:		Marconi Instruments MF Attenuator TF 2162 DC - 1MHz	AL15
5.	ADC:		Experimental Instrument	
6.	LADC:		Experimental Instrument	
7.	Plotter System:	(1)	DAC. Experimental Instrument	
		(2)	Hewlett-Packard 7005B X - Y Recorder	R-17
8.	Pulse Generator:		Hewlett Packard Type 80034	
9.	Oscillator:		General Radio Corp Type 1309-A 10Hz - 100KHz	102-30

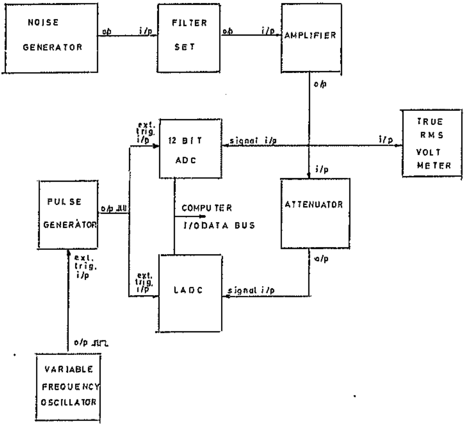


Fig. A 6.3 DETAILED INTERCONNECTIONS

A7 A Simple Worked Example of an 8 point DFTTLU

The example has the following specifications.

1) The X channel has 5 linearly spaced quantising levels.

2) The Y channel has 4 sinusoidally spaced abscissas

3) The tangent rule is used to solve the integration in both channels.

4) N = 16

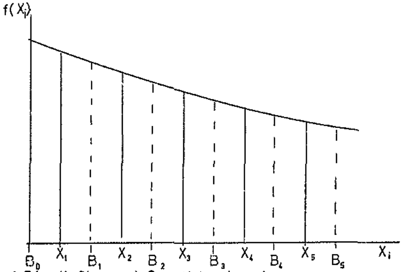


Fig.A 7.1 X Channel Quantising Levels

A7.1 The X Channel

The quantising thresholds are placed in the centre of each panel.

The areaof the ith panel is $(B_i - B_{i-1})\, f\,(X_i)$.

The total areaunder the function is

$$\int_0^1 f(x)dx \doteq \sum_1^5 a_i$$

$$\sum_{i=1}^{5} a_i = \sum_{i=1}^{5} (B_i - B_{i-1})\ f(X_i)$$

$$= h_x \sum_{i=1}^{5} w_i\ f(X_i)$$

Now $X_5 \equiv 1$ and the abscissas are linear

$$B_i - B_{i-1} = 0,2 = w_i$$

i	1	2	3	4	5
w_i	0,2	0,2	0,2	0,2	0,2

s	1	2	3	4	5
$\sum_{i=1}^{s} w_i$	0,2	0,4	0,6	0,8	1,0

$s = 1, M_x$

M_x is the number of quantising levels

$M_x = 5$

$w_i = 0,2$

$h_x = 1$

A7.2 TheY Channel

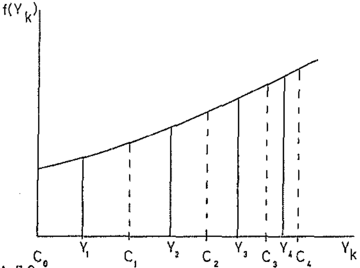


Fig. A 7.2

$$C_k = \sin\left(\frac{k\pi}{8}\right),\ k = 0,1,2,3,4 \qquad Y_k = \frac{C_k + C_{k-1}}{2}$$

The area, a_k, of the k th panel $= (C_k - C_{k-1})\ f\ (Y_k)$

The total area under the function is $\int_0^1 f(y)dy \doteq \sum_{k=1}^{4} a_k$

$$= \sum_{k=1}^{4} (C_k - C_{k-1})\ f\ (Y_k)$$

$$= h_y \sum_{k=1}^{4} u_k f(Y_k)$$

ie. $h_y = 1$

$$u_k = C_k - C_{k-1}$$

k	1	2	3	4
u_k	0,38	0,32	0,22	0,08

r	1	2	3	4
$\sum_{k=1}^{r} u_k$	0,38	0,70	0,92	1,00

My = number of quantising levels

$r = 1 \ldots\ldots M_y$

$M_y = 4$

Equivalent signed address generator output	Y channel weight contribution to STV
$0 \leqslant \theta < \frac{\pi}{8}$	0,38
$\frac{\pi}{8} \leqslant \theta < \frac{2\pi}{8}$	0,70
$\frac{2\pi}{8} \leqslant \theta < \frac{3\pi}{8}$	0,92
$\frac{3\pi}{8} \leqslant \theta \leqslant \frac{4\pi}{8}$ *	1,00

* The limits of this level are defined as in A1.2.3.

A7.4 The Derivation of the STV

$$\underset{\text{row}}{S}(s, \underset{\text{column}}{r}) = \sum_{i=1}^{s} \sum_{k=1}^{r} w_i u_k \qquad \left.\begin{array}{l} s = 1 \ldots\ldots M_x \\ r = 1 \ldots\ldots M_y \end{array}\right\}$$

The two dimensional array is derived as follows

s \ r	1	2	3	4
1	(0,38)(0,20)	(0,70)(0,20)	(0,92)(0,20)	(1,00)(0,20)
2	(0,38)(0,40)	(0,70)(0,40)	(0,92)(0,40)	(1,00)(0,40)
3	(0,38)(0,60)	(0,70)(0,60)	(0,92)(0,60)	(1,00)(0,60)
4	(0,38)(0,80)	(0,70)(0,80)	(0,92)(0,80)	(1,00)(0,80)
5	(0,38)(1,00)	(0,70)(1,00)	(0,92)(1,00)	(1,00)(1,00)

eg. S(2,3) · (0,92)(0,40)

Fig.A 7.3 STV Array

D_x and D_y are scaling factors. If the STV is set up in a minicomputer, the machine code representation is usually a different set of values. D_x and D_y are the scaling factors of these values.

In this case $D_x = 1$ $D_y = 1$

A7.5 Simple Fourier Analysis

The signal to be analysed is a 16 point sampled sine wave.

Consider the system in Fig.2.3. 'Clock 1' is initially set to the same frequency as 'Clock 2' ie $k = 1$. 'Clock 1' is then set to double the frequency of 'Clock 2' ie. $k = 2$.

Equation 2.32 is used.

k refers to frequency index in this section

$$\text{ie} \quad \overline{xy} = \frac{h_x h_y}{D_x D_y N} \sum_{n=0}^{N-1} \operatorname{sgn}(x(t).\ y(t))\ S(I_n K_{nk})$$

$$\overline{xy} = \frac{1}{16} \sum_{n=1}^{16} \operatorname{sgn}(x(t)\ .\ y(t))\ S(I_n K_{nk})$$

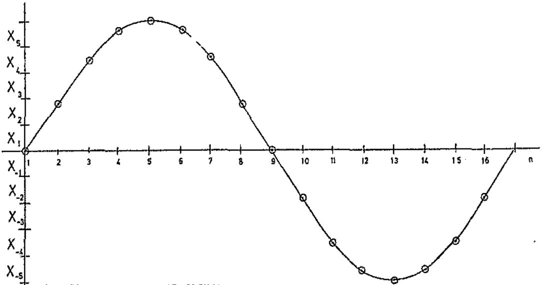


Fig. A 7.4 SAMPLED INPUT SIGNAL

TABLE A 7.1

time n	I_n	K_{n1}	K_{n2}	$S(I_nK_{n1})$	$S(I_nK_{n2})$
1	X_1	Y_1	Y_1	(0,38)(0,20)	(0,38)(0,20)
2	X_2	Y_2	Y_3	(0,70)(0,40)	(0,92)(0,40)
3	X_4	Y_3	Y_4	(0,92)(0,80)	(1,00)(0,80)
4	X_5	Y_4	Y_3	(1,00)(1,00)	(0,92)(1,00)
5	X_5	Y_4	$-Y_1$	(1,00)(1,00)	(0,38)(1,00)(-1)
6	X_5	Y_4	$-Y_3$	(1,00)(1,00)	(0,92)(1,00)(-1)
7	X_4	Y_3	$-Y_4$	(0,92)(0,80)	(1,00)(0,80)(-1)
8	X_2	Y_2	$-Y_3$	(0,70)(0,40)	(0,92)(0,40)(-1)
9	$-X_1$	$-Y_1$	Y_1	(0,38)(0,20)	(0,38)(0,20)(-1)
10	$-X_2$	$-Y_2$	Y_3	(0,70)(0,40)	(0,92)(0,40)(-1)
11	$-X_4$	$-Y_3$	Y_4	(0,92)(0,80)	(1,00)(0,80)(-1)
12	$-X_5$	$-Y_4$	Y_3	(1,00)(1,00)	(0,92)(1,00)(-1)
13	$-X_5$	$-Y_4$	$-Y_1$	(1,00)(1,00)	(0,38)(1,00)
14	$-X_5$	$-Y_4$	$-Y_3$	(1,00)(1,00)	(0,92)(1,00)
15	$-X_4$	$-Y_3$	$-Y_4$	(0,92)(0,80)	(1,00)(0,80)
16	$-X_2$	$-Y_2$	$-Y_3$	(0,70)(0,40)	(0,92)(0,40)
		$Ck_2=Ck_1$	$Ck_2=2Ck_1$	$\sum=10,22$	$\sum=0,00$
				$Ck_2=Ck_1$	$Ck_2=2Ck_1$

$S(I_n K_{nk})$ is the array in A7.3

Table A7.1 gives the quantised values of I_n of the signal as well as the values of K_{n1} and K_{n2}, that with I_n, form the address pairs of the STV. (I_n, K_{n1}) and (I_n, K_{n2}) are listed.

There is clearly [illegible] harmonic at $k = 2$ as the signal is a pure sine wave of frequency $k = 1$.

ie.

$$\overline{xy}/_{k=1} = \frac{1}{16}(10.22)$$

$$\overline{xy}/_{k=2} = 0$$