

# The Cyrix Corporation Fourier Transform Users Manual

**John H. Letcher**

## TABLE OF CONTENTS

Preface .....	2
Introduction .....	2
Systems and the Analysis of Signals .....	3
Definitions of Commonly Used Terms .....	4
The Laplace Transform .....	7
The Hankel Transform .....	10
The Fourier Transform .....	11
Linear Systems .....	16
Inverse Systems .....	20
The Hartley Transform .....	22
Inner Product spaces .....	25
Fourier Series .....	26
Basis Functions .....	27
Waveform Sampling .....	28
Discrete Transforms .....	30
The Fourier Transform .....	32
The Wavelet Transform .....	34
The FLIBSCI Fourier and Wavelet Transform Software .....	35

## Preface

Discrete integral transform techniques have been shown to be useful in many ways in the analysis of time series. A time series is defined as a *set* of data where each member of the set is a number. Each number represents the sampling of a continuously variable quantity which is the subject of our analyses. Usually, the time interval between each successive sample is chosen so that each interval duration is the same as any other. Furthermore, the set of data that is to be analyzed is ordered so that the first element in the set is the number that was obtained first and the last element in the set is the one that was obtained last. With the use of discrete integral transform techniques, information can be obtained from these data about this time series that cannot be obtained in other ways.

If one examines a text book on digital image processing (Castleman) it is found that over half of the book is devoted to expositions on integral transforms in general and discrete integral transforms in particular. The same thing can be said with regard to books on digital signal processing. So what are these things? Why are they useful? How can these be applied? Finally, how do I, a user, implement these ideas easily on a computer? This document is meant to function as a road map to explain to a computer user some of the salient points with regard to integral transforms. Then, the user will be introduced to the fact that accompanying this document is a floppy disk containing sets of files and these files are those that allow the user to carry out these operations called from his own Fortran or C program. The questions with regard to what these files are and how do I use them will be answered. Finally, what will be presented is a sequence of Fortran and C test programs that enable the user to try these routines and to gain a feeling for the action taken in the analysis of time series by both discrete fourier and discrete wavelet transformations.

## Introduction

This document is to serve to introduce a serious scientist or engineer to a new and exciting analytical tool, the wavelet transformation. The wavelet transformation is a member of a set of integral transforms, others of which, notably the Fourier transform, are already well established as a must in any well designed Scientist/Engineer tool kit. No prior knowledge of integral transforms is assumed in this document in order to read and understand the material. It is intended to introduce these topics in a way that develops a functional vocabulary for the Scientist/Engineer and to present some of the uses of these in subjects such as digital signal processing and statistical time series analysis.

Topics such as convolution and correlation will be presented and discussed; yet, it will be found that these techniques as well as most others of interest are performed by execution of only a small set of techniques. Three of these are fundamental: 1) the fourier transform, 2) the (forward) wavelet transform, and 3) the inverse wavelet transform. We mean by this that other techniques such as convolution and correlation are performed by enlisting services of one of the basic three tools (that is, after a modest preparation of data). How the basic three are performed (the algorithms) is interesting yet is not essential to the understanding of how to use any of the entire set of tools.

The basic three computer routines (to perform these tasks on user supplied data) are normally written in assembly language. These three are specifically designed to take full advantage of the architecture and organization of the arithmetic processor on which the calculations will be performed. But, the rest of the tools are written in high level languages

(Fortran-77 and C). These are easily read and understood by those skilled in the art of science or engineering and, if desired, the action of the three assembly language routines (which consume over 95% of the computer time) can be taken as an article of faith.

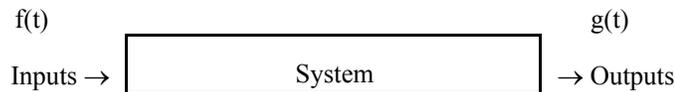
A statement should be made with regard to notation used in this document. A (non bold face) Roman letter will be used to designate a continuous function and the symbol shown in parenthesis will be the independent variable. That is,  $f(t)$  will be used to describe a function of the variable  $t$  and  $f(t_i)$  will denote the *value* of the function at  $t = t_i$ , that which has been measured or calculated. Bold face Roman characters will denote sets of values of functions taken over regular intervals; that is,  $\mathbf{f}$  will stand for a set of array values  $\{f(t_i)\}$ ,  $i = 1, \dots, N$ . Therefore,  $(\mathbf{f})_i = f(t_i)$ . For purposes of these discussions the values of  $f(t)$  are unknown for all values of  $t$  less than  $t_1$  and are unknown (or unmeasurable) for  $t$  greater than  $t_N$ . The regularity of the interval states that  $t_{i+1} - t_i = T$ , a fixed known value. As a result, the total sample time is  $NT \equiv T_o$ .

### Systems and The Analysis of Signals

The subject of our analyses, the signal, is of one or two types of sequences of numbers which are sets of numbers which are of defined lengths. These numbers represent the *sampling* of a continuous function, normally of a single independent variable, at regularly spaced intervals. The purpose of the analyses is to determine by calculation, information about the *continuous* function that is not readily discernable by inspection of the signal itself.

The value of the numbers of the set are either real or complex; that is, the signal that is sampled is single valued at any instant of time or it has two values. In the latter case it is convenient to represent the two numbers,  $x_i$  and  $y_i$ , as components of a complex function  $f$  such that  $f(t_i) = x(t_i) + i y(t_i)$  where  $i = \sqrt{-1}$ .  $x$  is called the real part and  $y$  is called the imaginary part. The reader is cautioned that the term imaginary is used in the *mathematical* sense; it does not refer in any way to the *reality* of the values or signal component. The analyses will be carried out entirely on *sampled data*, those all which values are determined at fixed intervals of time. In this process of sampling, information is lost with regard to the continuous function; yet, theorems exist that allow us to put bounds on any errors that may occur in our analyses using the sampled data. This is of utmost importance because our analyses are to be performed on digital computers with which we have difficulty defining continuous functions. These machines prefer sets of numbers to operate upon.

We wish to study signals for many reasons. One of the best of these is that it offers techniques to study "systems". A cybernetic system is any configuration (math, physical, biological, social) that admits inputs and produces outputs.



By applying known input signals to a system and measuring the outputs and performing analyses on these input and output signals, it is possible to *characterize* a system. This then makes it possible to calculate the outputs of a system under any set of inputs; i.e., we have a *model*.

The modelling capabilities and the predictive abilities of the models make these techniques desirable to use anytime the values of attributes of an entity can be measured as a sequence of number values. Our analyses give us the ability to learn more about the entities which are under study than can be directly measured.

Our goal is to acquire information about signals and systems using sampled data, only, and then to infer knowledge about the systems that in fact admit *continuous inputs* and produce *continuous outputs*.

Our analyses will be couched in such terms that we know nothing about the signal,  $f$ , before the time at which the first point is sampled. Furthermore, we will acquire a finite number of points and know nothing about the signal after the time at which the last point is sampled, yet we wish to glean information about the continuous function,  $f$ , from only these values, perhaps even to extrapolate what the sampled value will be at a future time. This is quite a challenge.

We must start with a rather theoretical discussion of mathematical properties of single valued functions in the continuous domain. After establishing relationships in the continuous domain, a magic wand will be waved over all of the continuous function relationships and equivalent ones for discrete (sampled) data will be presented. This step is far from trivial and these mappings will be given without any rigorous proofs as excellent references are given to the reader which justify these leaps. The point is that *relationships are easier to establish and prove in the continuous domain* yet are all but impossible to address by calculation in a digital computer owing to its inherent discrete nature of its data objects. As the theorems and relationships stated for continuous functions have direct corresponding counterparts in the discrete representation, we will use each in its place as it suits our purposes.

Our goal is to study continuous signals and systems producing continuous outputs. We will use the tools of sampled data sets. We will then be able to state the bounds on the errors, inaccuracies or misinterpretations due to the fact that we sampled the signals. This seemingly unreachable goal is made possible because of the *Sampling Theorem* which in essence states that it is possible to reconstruct a continuous function *exactly* from sampled data as long as the signal has certain properties (e.g. the signal does not contain frequencies greater than half of the sampling frequency).

### Definitions of Commonly Used Terms

We intend to introduce a list of properties of continuous functions of one independent variable. The reader is cautioned that this is not a rehash of the material learned in college mathematics courses. The integrals are not Riemannian. Many of the functions under study have properties that would not have been allowed in undergraduate courses: imagine a function that is zero everywhere except one point, and is undefined at that one point. Yet, this function exhibits properties so interesting that its description deserves chapters in books (this is the impulse function  $\delta(t)$  described below).

This topic described herein is based on the fine work of Oliver Heavyside (1850-1925) whose *Operational Calculus* established much of the foundation of this material. This work was continued and extended by others, eg., Schwartz (1950) in his *Theory of Distributions*.

The definitions listed below extends what was known and what was presented in standard calculus classes.

The object of study is a *signal*,  $f$ , which is a function of  $t$  (which is usually time, but need not be). In general,  $f$  is complex; that is

$$f(t) \equiv f_{\text{real}}(t) + i f_{\text{imag}}(t) \text{ where } i = \sqrt{-1}.$$

The magnitude operation  $|f(t)|$  is defined as

$$|f(t)| \equiv \sqrt{f_{\text{real}}^2 + f_{\text{imag}}^2}.$$

A time function  $f(t)$  is called  *$L^1$ -stable* if

$$\int_{-\infty}^{\infty} |f(t)| dt < \infty.$$

A time function is called  *$L^2$ -stable* iff

$$\int_{-\infty}^{\infty} |f(t)|^2 dt < \infty .$$

A *convolution* of two time functions  $f(t)$  and  $g(t)$  is given by

$$h(t) = \int_{-\infty}^{\infty} f(t-\tau) g(\tau) d\tau \equiv f * g .$$

Note that if  $f$  and  $g$  are  $L^1$ -stable, so is  $f * g$ .

The convolution is *commutative*, i.e.,  $f * g = g * f$  and, the convolution is *associative*, i.e.,  $(f * g) * h = f * (g * h)$  and the convolution is *distributive* with respect to addition, i.e.,  $f * (g + h) = f * g + f * h$ .

The Unit Step (Heavyside) Function is defined so that  $u(t) \equiv 0, t < 0$  and  $u(t) \equiv 1, t \geq 0$ . Familiar function forms may be used to approximate the Heavyside function. For example,

$$u(t) = \frac{1}{2} \left( 1 + \frac{t}{\sqrt{t^2}} \right) \quad \text{or}$$

$$u(t) = \lim_{\varepsilon \rightarrow 0^+} \left[ \frac{1}{2} + \frac{1}{\pi} \tan^{-1} \frac{t}{\varepsilon} \right], \quad \varepsilon > 0 \quad \text{where} \quad -\frac{\pi}{2} \leq \tan^{-1} \frac{t}{\varepsilon} \leq \frac{\pi}{2} .$$

The *Impulse Function* is defined so that

$$\delta(t) = 0 \quad t \neq 0 \quad \text{and} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1 .$$

A basic relationship is that

$$\int_0^{\infty} f(t) \delta(t) dt = f(0) .$$

The function  $\delta(t)$  exhibits an interesting *scaling property* which is

$$\delta(at) = \frac{1}{|a|} \delta(t) .$$

We may use the relationships that

$$\delta(t) = \lim_{\varepsilon \rightarrow 0^+} \frac{1}{\pi} \frac{\varepsilon}{t^2 + \varepsilon^2}, \quad \varepsilon > 0, \quad \text{and} \quad \delta(t) = \frac{d}{dt} u(t) .$$

It should be noted that  $\delta(t)$  is *even*, that is,  $\delta(-t) = \delta(t)$ .

Also,  $\delta(t)$  Plays the role of Unity in convolution, that is,  $f(t) * \delta(t) = f(t)$ .

The *Dirac Theorem* states that

$$f(t) * \delta^{(n)}(t) = \left(\frac{d}{dt}\right)^n f(t)$$

A function  $f(t)$  is *one-sided* if  $f(t) = 0$   $t < 0$ . Hence  $f(t) u(t) = f(t)$ .

## The Laplace Transform

The *one-sided*  $L^2$  Laplace transform is defined by two relationships. The first is the forward Laplace transform  $F(p)$  of  $f(t)$  is defined by the relationship

$$F(p) = \int_0^{\infty} f(t) e^{-pt} dt$$

The second relationship expresses the inverse transform. It can be shown to be given by

$$f(t) = \frac{1}{2\pi i} \int_{\sigma-i\infty}^{\sigma+i\infty} F(p) e^{pt} dp, \quad p_1 \leq \sigma \leq p_2.$$

We now introduce the symbol  $\leftrightarrow$  to represent the Laplace transform and its inverse. Now we say:

$$f(t) \leftrightarrow F(p)$$

Note that

$$cf(t) \leftrightarrow cF(p)$$

$$g(t) + f(t) \leftrightarrow G(p) + F(p)$$

and

$$f(t+c) \leftrightarrow e^{cp} F(p) \quad \text{where } c \text{ is complex.}$$

The last relation is obtained by using the substitution  $s = t + c$  and  $ds = dt$ , then

$$f(t+c) \leftrightarrow \int_0^{\infty} f(t+c) e^{-pt} dt = \int_0^{\infty} e^{cp} f(s) e^{-ps} ds$$

An important theorem states that the Laplace Transform of  $\delta(t) = 1$  for all  $p$

$$F(p) = \int_0^{\infty} \delta(t) e^{-pt} dt = e^{-p0} = 1$$

Another theorem states that if  $f(t)$  has Laplace transform  $F(p)$ , then

$$\frac{d}{dt} f(t) \equiv f'(t) \leftrightarrow pF(p)$$

since

$$\frac{f(t+\varepsilon) - f(t)}{\varepsilon} \leftrightarrow \left(\frac{e^{\varepsilon p} - 1}{\varepsilon}\right) F(p) = pF(p).$$

in the limit as

$$\varepsilon \rightarrow 0.$$

The *Convolution Rule* states that if  $F(p)$  and  $G(p)$  are the Laplace transforms of  $f(t)$  and  $g(t)$ , then the product  $F(p)G(p)$  is the Laplace transform of the convolution,  $f(t) * g(t)$ , since

$$f(t-\tau) \leftrightarrow e^{-p\tau} F(p)$$

and

$$f(t-\tau)g(\tau) d\tau \leftrightarrow e^{-p\tau} F(p)g(\tau) d\tau$$

hence,

$$\int f(t-\tau)g(\tau) d\tau \leftrightarrow \int e^{-p\tau} F(p)g(\tau) d\tau$$

$$f(t)*g(t) \leftrightarrow F(p)\int e^{-p\tau} g(\tau) d\tau$$

$$f(t)*g(t) \leftrightarrow F(p)G(p)$$

If  $f(t)$  has the Laplace transform  $F(p)$ , then

$$\frac{d^n}{dt^n} f(t) \equiv f^{(n)}(t) \leftrightarrow p^n F(p)$$

which is obtained by repeated application of the fact that

$$\frac{d}{dt} f(t) \leftrightarrow p F(p).$$

This allows us to state that the Laplace transform of the  $n$ th derivative  $\delta^{(n)}(t)$  of the impulse function  $= p^n$  ( $-\infty < \text{Re}(p) < \infty$ ), since  $\delta(t) \leftrightarrow 1$ .

Simply restated, the *Dirac theorem* states that if  $\delta^{(n)}(t)$  is the  $n$ th derivative of the impulse function, then

$$\begin{aligned} f(t) * \delta^{(n)}(t) &= \int f(t-s) \delta^{(n)}(s) ds \\ &= \int f(t) \delta^{(n)}(t-s) ds = f^{(n)}(t) = \frac{d^n}{dt^n} f(t) \end{aligned}$$

A one-sided  $L^2$ -stable time function  $w(t)$  has been called a *wavelet* although a more restrictive definition will be used later. Its Laplace transform

$$W(p) = \int_0^{\infty} w(t) e^{-pt} dt$$

is called the *transfer function* of the wavelet. Its total energy is

$$\int_0^{\infty} |w(t)|^2 dt = \int_{-\infty}^{\infty} |W(f)|^2 df$$

and the partial energy up to time  $\tau$  is defined to be

$$\int_0^{\tau} |w(t)|^2 dt$$

## The Hankel Transform

The Hankel Transform is defined

$$F_H(s) \equiv H_m f(t) \equiv H[f(t); s] = \int_0^{\infty} f(t) t J_m(st) dt$$

where  $f(t)$  is a real function and  $J_m(z)$  is the  $m^{\text{th}}$  order Bessel function.

We require that

$$\int_0^{\infty} |f(t)| dt < \infty$$

and  $f(t)$  is of bounded variation in the neighborhood of  $t = 0$ .

$$f_1(t) = \int_0^{\infty} f(s) J_m(st) ds$$

$$J_m(z) = \left(\frac{z}{2}\right)^m \sum_{k=0}^{\infty} \frac{(-1)^k}{k! \Gamma(m+k+1)} \left(\frac{z}{2}\right)^{2k}$$

## The Fourier Transform

The  $L^2$  Fourier transform may be defined by

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

and

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega$$

*Bessels Equality* states that

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |F(\omega)|^2 d\omega.$$

The Fourier Transform

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

is called the *gain-and-phase spectrum* where  $|F(\omega)|$  is the *gain*,  $\log |P(\omega)|$  is the *attenuation* and  $P(\omega)$  is the *phase-shift*, where  $F(\omega) = |F(\omega)| e^{iP(\omega)}$ .

The Fourier Transform may also be expressed as

$$H(f) = \int_{-\infty}^{\infty} h(t) e^{-2\pi i f t} dt$$

which may be decomposed as follows:

$$H(f) = R(f) + iI(f) = |H(f)| e^{i\theta(f)}$$

$|H(f)|$  is the amplitude or *Fourier Spectrum*  $= \sqrt{R^2(f) + I^2(f)}$ , and  $\theta(f)$  is called the phase  $= \tan^{-1} I(f) / R(f)$

The Inverse Fourier Transform is given by

$$h(t) = \int_{-\infty}^{\infty} H(f) e^{2\pi i f t} df$$

It should be noted that the functions  $h(t)$  and  $H(f)$  may be real but are in general complex, that is there is a real and an imaginary part to each.

The Fourier Transform operation may be represented by the notation

$$h(t) \Leftrightarrow H(f), \quad h(t) = \text{FT}^{-1}(H(f)), \quad \text{and } H(f) = \text{FT}(h(t)).$$

It is sufficient (but not necessary) that  $\int_{-\infty}^{\infty} |h(t)| dt < \infty$ .

Examples of Fourier transform pairs are the following

$$h(t) = A, \quad |t| < T_0 \Leftrightarrow 2 A T_0 \frac{\sin(2\pi T_0 f)}{2\pi T_0 f}$$

$$h(t) = K \Leftrightarrow K \delta(f).$$

$$H(f) = A, \quad |f| < f_0 \Leftrightarrow 2 A f_0 \frac{(\sin 2\pi f_0 t)}{2\pi f_0 t}$$

It is straight forward to show that

$$A \cos(2\pi f_0 t) \Leftrightarrow \frac{A}{2} \delta(f - f_0) + \frac{A}{2} \delta(f + f_0)$$

and

$$A \sin(2\pi f_0 t) \Leftrightarrow i \frac{A}{2} \delta(f + f_0) - i \frac{A}{2} \delta(f - f_0)$$

Also, another Fourier transform pair is a sequence of equally spaced impulses with another sequence of equally spaced impulses.

$$h(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \Leftrightarrow H(f) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta(f - \frac{n}{T})$$

Parseval's Theorem states that

$$\int_{-\infty}^{\infty} |h(t)|^2 dt = \int_{-\infty}^{\infty} |H(f)|^2 df$$

Fourier transform properties are given by the following

linearity:

$$x(t) + y(t) \Leftrightarrow X(f) + Y(f)$$

symmetry:

$$\text{FT}(\text{FT}(h(t))) = h(-t)$$

time scaling:

$$h(kt) \Leftrightarrow \frac{1}{|k|} H\left(\frac{f}{k}\right)$$

frequency scaling:

$$\frac{1}{|k|} h\left(\frac{t}{k}\right) \Leftrightarrow H(kf)$$

time shifting:

$$h(t - t_0) \Leftrightarrow H(f) e^{2\pi i f t_0}$$

frequency shifting:

$$h(t) e^{2\pi i f_0 t} \Leftrightarrow H(f - f_0)$$

For  $h_e$  even:

$$h_e(t) \Leftrightarrow R_e(f) = \int_{-\infty}^{\infty} h_e(t) \cos(2\pi f t) dt$$

For  $h_o$  odd:

$$h_o(t) \Leftrightarrow i I_o(f) = \int_{-\infty}^{\infty} -i h_o(t) \sin(2\pi f t) dt$$

$$h(t) = h_e(t) + h_o(t) = \left[ \frac{h(t)}{2} + \frac{h(-t)}{2} \right] + \left[ \frac{h(t)}{2} - \frac{h(-t)}{2} \right]$$

$$H(f) = R(f) + iI(f) = H_e(f) + H_o(f)$$

Complex Time Functions can be represented by the following

$$h(t) = h_r(t) + i h_i(t)$$

$$H(f) = R(f) + i I(f)$$

The table on page 46 in the Brigham reference should be examined for more information.

*Convolution* is again defined

$$y(t) = \int_{-\infty}^{\infty} x(\tau) h(t-\tau) d\tau \equiv x(t) * h(t) = h(t) * x(t)$$

The *Convolution theorem* states that

$$h(t) * x(t) \Leftrightarrow H(f) X(f)$$

Also,

$$h(t) x(t) \Leftrightarrow H(f) * X(f)$$

*Correlation* is defined by the relationship

$$z(t) = \int_{-\infty}^{\infty} x(\tau) h(t+\tau) d\tau$$

$$\int_{-\infty}^{\infty} h(\tau) x(t+\tau) dt \Leftrightarrow H(f) X^*(f)$$

here  $X^*(f)$  is the complex conjugate of the function  $X(f)$ . When the functions  $x$  and  $h$  are the same the term *auto-correlation* is used. Otherwise, *cross-correlation* is applied.

Important properties of convolution are that:

commutative:

$$h * x = x * h$$

associative:

$$h * [g * x] = [h(t) * g(t)] * x(t)$$

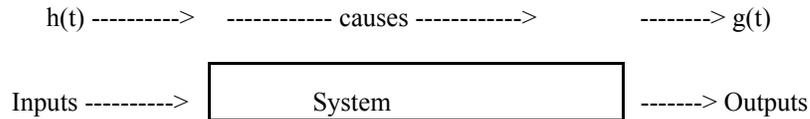
distributive with respect to addition:

$$h(t) * [g(t) + x(t)] = h(t) * g(t) + h(t) * x(t)$$

$$\frac{d}{dt}(f(t)*g(t)) = f' * g + f * g'$$

## Linear Systems

A *system* is any configuration (mathematical, physical, biological, social) which over time admits inputs and yields outputs. An input  $h(t)$  is said to *cause*  $g(t)$ , the output or *effect* or *normal response* of this system. This is to say that there is no part of  $g(t)$  which is not *caused* by  $h(t)$ , i.e., there are no additional internal factors generating a contribution to  $g(t)$ .



We assume that the system has been at rest up to  $t = t_0$ , i.e.,  $h(t) = 0, t < t_0$ . Therefore,  $h(t)$  is one-sided.

At this point a few definitions are in order:

1. The system is called *realizable* if  $g(t) = 0$  for  $t < t_0$
  2. The system is called *time or shift invariant* if  $h(t + s)$  causes  $g(t + s)$
- and
3. The system is called *linear* if  $[c_1 h_1(t) + c_2 h_2(t)]$  causes  $[c_1 g_1(t) + c_2 g_2(t)]$

Henceforth, the term "system" is used for a cause and effect, time invariant, linear system. We are concerned only with the relationship of the output to the input. What is inside the system is of no concern to us in the present discussion.

Our definition of system is confined to the subset of the general class of systems so that the relationships of linearity and shift invariance hold. Furthermore, so that we may be able to use the mathematical machinery that we have developed, we require that the input to the systems be one-sided.

$$h_1(t) = \exp(i\omega t) = e^{i\omega t} = e^{2\pi i f t} = \cos(\omega t) + i \sin(\omega t)$$

Consider now an input function which is a complex valued signal of the form where  $i^2 = -1$  and  $\omega = 2\pi f$ . This is called a *harmonic signal*.

The output of a shift invariant linear system subjected to this input may be defined to be

$$\begin{aligned}
 g_1(t) &= B(\omega, t) h_1(t) \\
 &= B(\omega, t) \exp(i\omega t)
 \end{aligned}$$

We may supply a second input signal  $h_2(t)$  to this system which is obtained by time shifting  $h_1(t)$ .

$$\begin{aligned}
h_2(t) &= \exp(i\omega[t-T]) \\
&= \exp(-i\omega T)\exp(i\omega t) \\
&= \exp(-i\omega T)h_1(t)
\end{aligned}$$

The output of this system to the input  $h_2(t)$  is given by

$$\begin{aligned}
g_2 &= B(\omega, t - T)\exp(i\omega[t - T]) \\
&= B(\omega, t - T)\exp(-i\omega T)\exp(i\omega t)
\end{aligned}$$

Because the system is linear, an input signal of  $f_1(t)$  multiplied by the complex constant  $\exp(-i\omega T)$  produces an output

$$g_2(t) = \exp(-i\omega T)B(\omega, t)\exp(i\omega t)$$

Therefore,

$$B(\omega, t) = B(\omega, t - T) = B(\omega)$$

We may now say that the response of a shift invariant linear system to an harmonic input is simply that input multiplied by a frequency dependent complex constant. Also, we have shown that an harmonic input always produces an harmonic signal output at the *same frequency*.

Shift invariant linear systems also preserve *realness*

$$x(t) \text{ causes } y(t) \text{ implies that } \operatorname{Re}(x(t)) \text{ causes } \operatorname{Re}(y(t))$$

That is, the real and imaginary parts of an harmonic input go through a system independently.

In general, the input signal  $h(t)$  will consist of many frequencies. The Fourier transform of  $h(t)$  is  $H(f)$  which may be nonzero for many values of  $f$ . Also, we say that the Fourier transform of the output signal  $g(t)$  is  $G(f)$ . At each of the frequencies the relationship

$$\begin{aligned}
G(f) &= B(\omega)H(f) \\
&= B(f)H(f)
\end{aligned}$$

holds. For a given linear system the value of  $B(f)$  is defined although, clearly, the value at one frequency need not be the same as it is at any other frequency. We call the function  $B(f)$  the *transfer function* of the

system. The importance of this function is that this function, alone, *characterizes* the system; that is, once the transfer function  $B(f)$  is known, then we can calculate the output signal of a system for an arbitrary (suitably well-behaved) input signal. The above states that for an arbitrary input function  $h(t)$  with Fourier transform  $H(f)$ , the Fourier transform of the output signal  $G(f)$  is the product of  $B(f)$  and  $H(f)$ . So, the output signal  $g(t)$  may be calculated by taking the inverse Fourier transform of  $G(f)$ .

The convolution theorem states that the product of two function in frequency space (the transform domain) is given by the convolution of two functions in the time domain.

So if  $H(f) = FT(h(t))$  and  $G(f) = FT(g(t))$ , then the relationship

$$G(f) = B(f)H(f)$$

implies that

$$g(t) = b(t) * h(t)$$

where

$$B(f) = FT(b(t))$$

If this system is given an input signal which is the impulse  $\delta(t)$  then the output of the system is

$$g(t) = b(t) * \delta(t) = b(t)$$

The response of the system to an impulse is the function  $b(t)$  which is the inverse Fourier transform function  $B(f)$ . We call  $b(t)$  the *impulse response* of the system. The knowledge of this function (and this function alone) is enough to be able to calculate any output signal from any input signal.

Before we leave this discussion, some additional terms should be defined:

The system is *dissipative* if its impulse function  $b(t)$  is  $L^2$  stable.

It is *realizable* and *dissipative* if and only if  $b(t)$  is a *wavelet*, an  $L^2$  stable one-sided function. Note that the term wavelet will be used in a more restrictive sense later.

The system is called a *pure delay* system if for  $f(t)$  as input,  $g(t) = f(t - \alpha)$   $\alpha \geq 0$ ,  $\alpha \equiv$  the time delay, hence the transfer function of a pure delay is

$$e^{-p\alpha}, \alpha \geq 0.$$

Consider:

$$b(t) = u(t) e^{\alpha t}$$

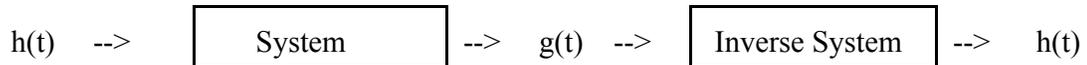
where  $u(t)$  is the Heavyside function ( $u(t) = 1, t \neq 0, 0$ , otherwise). The transfer function of this  $b(t)$  is  $1 / (p - \alpha)$ .

The following table relates the terms realizable and dissipative to possible forms of  $b(t)$ .

$u(t) e^{\alpha t}$	realizable and non-dissipative	$\text{Re } \alpha < 0$
$-u(-t) e^{\alpha t}$	non-realizable and non-dissipative	$\text{Re } p < \text{Re } \alpha$
$u(t) e^{\beta t}$	realizable and non-dissipative	$\text{Re } p > \text{Re } \beta > 0$
$-u(-t) e^{\beta t}$	non-realizable and dissipative	$\text{Re } p < \text{Re } \beta > 0$

## Inverse Systems

Consider a system with impulse response  $b(t)$ . It could be that the action of this system produces an undesirable result. We may wish to "undo" the undesirable action of the system by adding an additional system in series with the other so that the two systems, taken together, act as if they were perfect; i.e. all frequencies are passed through the combined system without attenuation (change in amplitude) or change in phase. That is, the transfer function of the combined system is such that  $B_T(f) \equiv 1$ . We call the second system the *inverse system*. Diagrammatically, we have



We have already established that

$$g(t) = b(t) * h(t)$$

or

$$G(f) = B(f)H(f)$$

Let us now assert that if it exists, the impulse response of the inverse system shall be defined to be  $i(t)$  with Fourier transform  $I(f)$ . For the inverse system to produce  $h(t)$  as its output from an input  $g(t)$ , then it must follow that

$$H(f) = I(f)G(f)$$

and

$$h(t) = i(t) * g(t)$$

If we view the system and the inverse system, taken together, we see that

$$h(t) = i(t) * g(t) = i(t) * b(t) * h(t)$$

and

$$H(f) = I(f)B(f)H(f)$$

Again if it exists, then it follows that

$$1 = I(f)B(f)$$

or

$$I(f) \equiv B^{-1}(f) \quad , \text{ the reciprocal of } B(f)$$

and

$$i(t)*b(t) = \delta(t) \quad , \text{ an impulse.}$$

The problems of existence come when  $B(f) = 0$  for some values of  $f$ . If, however, we assert that  $I(f) \equiv 0$  and under these circumstances,  $0/0 \equiv 1$ , then our problems are defined away! This apparently outrageous assertion will prove to be quite workable in most computational circumstances.

### The Hartley Transform

The *Hartley transform* is a special case for the Fourier transform which is defined on a real function  $V(t)$ . Application of this transform produces a real function  $H(f)$ . The transform pair is defined:

$$H(f) = \int_{-\infty}^{\infty} V(t) \text{cas}(2\pi f t) dt$$

$$V(t) = \int_{-\infty}^{\infty} H(f) \text{cas}(2\pi f t) df$$

where  $\text{cas}(t) \equiv \cos(t) + \sin(t)$

For comparison, the Fourier transform written according to the same convention is given by:

$$F(f) = \int_{-\infty}^{\infty} V(t) e^{-j2\pi f t} dt$$

and

$$V(t) = \int_{-\infty}^{\infty} F(f) e^{j2\pi f t} df$$

The relationship between the Fourier and Hartley transforms is derived based upon symmetry considerations. To do this we split  $H(f)$  into its even and odd parts,  $E(f)$  and  $O(f)$ . To connect the transform  $H(f)$  with the Fourier transform  $F(f)$  of  $V(t)$ , we adopt the following definition: Let  $H(f) = E(f) + O(f)$ , where  $E(f)$  and  $O(f)$  are the even and odd parts of  $H(f)$ , respectively. Then

$$E(f) = (H(f) + H(-f)) / 2 = \int_{-\infty}^{\infty} V(t) \cos(2\pi f t) dt$$

$$O(f) = (H(f) - H(-f)) / 2 = \int_{-\infty}^{\infty} V(t) \sin(2\pi f t) dt$$

The Fourier transform  $F(f)$  has a real part which is the same as  $E(f)$  and an imaginary part whose negative is  $O(f)$ . That is,  $F_{\text{real}}(f) = \text{Re } F(f) = E(f)$  and  $F_{\text{imag}}(f) = \text{Im } F(f) = -O(f)$ .

Conversely, given the Fourier transform  $F(f)$ , we may obtain  $H(f)$  by noting that  $H(f) = F_{\text{real}}(f) - F_{\text{imag}}(f)$ ; that is, from  $F(f)$  one finds  $H(f)$  as the sum of the real part and sign-reversed imaginary part of the Fourier transform.

To summarize: The Fourier transform is the even part of the Hartley transform minus  $i$  times the odd part; conversely, the Hartley transform is the real part of the Fourier transform minus the imaginary part.

The *Discrete Hartley transform* of a real function  $f(t)$  is, with its inverse, is given by:

$$H(f) = \sum_{t=0}^{N-1} f(t) \text{cas}(2\pi f t / N)$$

$$f(t) = (1/N) \sum_{f=0}^{N-1} H(f) \text{cas}(2\pi f t / N)$$

Using this notation, the Discrete Fourier transform and its inverse have the standard form:

$$F(f) = \sum_{t=0}^{N-1} f(t) \exp(-2\pi i f t / N)$$

$$f(t) = (1/N) \sum_{f=0}^{N-1} F(f) \exp(2\pi i f t / N)$$

We can say that  $f/N$  is the same as frequency measured in Hz, over the range  $-N/2 < f < N/2$ . We also arrive at equations for the even and odd parts that are consistent with what has gone before. Thus:

$$E(f) = (H(f) + H(N-f)) / 2$$

and

$$O(f) = (H(f) - H(N-f))/2$$

From the definition of the discrete Fourier transform it is apparent that  $F(f)$  can be formed from the discrete Hartley transform's even and odd parts by using:

$$F(f) = E(f) - iO(f)$$

Conversely, to form  $H(f)$  when  $F(f)$  is available, one may use

$$H(f) = \text{Re } F(f) - \text{Im } F(f).$$

## Inner Product Spaces

Let us now consider  $C[0,1]$ , the space of all continuous real valued functions on the closed interval  $[0,1]$  and  $L^2 [0,1]$ , the space of all square integrable functions on the interval  $[0,1]$ . Basis functions for these spaces will be introduced and it will be shown how discrete analysis is applied to these spaces through the use of the discrete fourier transform (which deals with  $L^2[0,1]$ ) and the wavelet transformation (which deals with  $C[0,1]$ ).

In each of these spaces, a scalar product is defined: if  $x(t)$  and  $y(t)$  are two functions defined in the

$$\langle x(t), y(t) \rangle = \int_0^1 x^*(t) y(t) dt$$

space then the scalar (inner) product of these function is given by where  $x^*(t)$  is the complex conjugate of  $x(t)$ .

Also, a *norm* is defined by

$$|x(t)| = \left[ \int |x(t)|^2 dt \right]^{\frac{1}{2}} = \left[ \int x^*(t) x(t) dt \right]^{\frac{1}{2}}$$

### Fourier Series

A periodic real valued function  $y(t)$  with period 1 is one that exhibits the property that  $y(t) = y(t + n)$  where  $n = 0, \pm 1, \pm 2$ , etc. This function,  $y(t)$ , may be expressed

$$y(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(2\pi n t) + b_n \sin(2\pi n t)]$$

The magnitude or coefficients of the sinusoids are given by the integrals

$$a_n = 2 \int_0^1 y(t) \cos(2\pi n t) dt$$

and

$$b_n = 2 \int_0^1 y(t) \sin(2\pi n t) dt.$$

By applying the identities

$$\cos(2\pi n t) = \frac{1}{2} [e^{2\pi i n t} + e^{-2\pi i n t}]$$

and

$$\sin(2\pi n t) = \frac{1}{2} [e^{2\pi i n t} - e^{-2\pi i n t}]$$

then

$$y(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n - ib_n) e^{2\pi i n t} + \sum_{n=1}^{\infty} (a_n + ib_n) e^{-2\pi i n t}$$

It is not hard to show that if we were to introduce negative values of  $n$  (to simplify the expressions above) then  $a_{-n} = a_n$  and  $b_{-n} = -b_n$ . Note that since  $b_{-n} = -b_n$ , then  $b_0 = 0$ . We now have

$$y(t) = \frac{a_0}{2} + \frac{1}{2} \sum_{n=-\infty, n \neq 0}^{\infty} (a_n - ib_n) e^{2\pi i n t} = \sum_{n=-\infty}^{\infty} a_n e^{2\pi i n t}$$

Here,  $a_n = 1/2 (a_n - ib_n)$   $n = 0, \pm 1, \pm 2$ , etc.

## Basis Functions

Consider a set of functions  $\{\Phi_n(t)\}$ ,  $n = 0, \dots, N-1$  such that

$$\int_0^1 \phi_n^*(t) \phi_m(t) dt = \delta_{nm} .$$

This is the delta function, which has the value of 1 if  $n=m$ , and 0 otherwise.

We call the set of function an orthonormal basis of a space. In this inner product space,  $S$ , any *member function*,  $f$ , can be exactly represented as a linear combination of the basis functions. That is,

$$f(t) = \sum_{n=0}^{N-1} \beta_n \phi_n(t)$$

$$\beta_n = \int_0^1 f^*(t) \phi_n(t) dt$$

The expansion coefficients  $\beta_n$  are calculated by the relationships

Carrying out these integrals to calculate the expansion coefficients is precisely what the discrete fourier and wavelet transformations perform!

For the discrete complex fourier transform the basis functions are

$$\phi_0 = 1 \quad \phi_n = [ \cos ( 2 \pi n t ) + i \sin ( 2 \pi n t ) ] = \exp ( 2 \pi i n t )$$

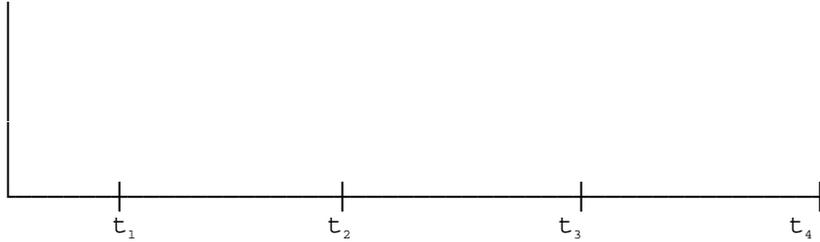
A multiple of the single function  $\phi_0$  represents the dc offset in the function  $f(t)$ .

The wavelet transform uses a single  $\phi_0$ , called a scaling function. The rest of the basis functions are dilations (a function made taller and thinner by multiplication by a constant) or translations (a function moved intact laterally along the independent variable axis) of a single function, called a wavelet, that has very special properties.

The act of performing an integral transform on a function answers the question: "How much of the function is contributed (in building the linear combination) by each of the basis functions?".

## Waveform Sampling

Consider a real (single valued) function  $f(t)$ . This is a continuous function, but we have chosen to employ techniques to measure the value of the function at any defined value of  $t$  very accurately. If this signal representing the function  $f(t)$  is the voltage at a point in a circuit, then with analog to digital circuitry it is possible to measure and record the binary value of the voltage precisely upon the command (a pulse) of a computer. It is possible to measure this value in a regular sequence of  $t$  values. What is produced is a set of numbers each of which represent a value of the signal at a precise instant of time. Consider Fig (1) which shows a family of functions that all have the same value at the sampling times.



Clearly, the three functions labeled A, B and C are not the same; yet measuring the values of each at  $t_1, t_2, \dots$  produce identical values. Our challenge is to be able to use the sampled values in our analyses ignoring (for the moment) the differences between functions A, B and C.

This sampling process is equivalent to convolving the continuous signal with a sequence of impulse functions  $\delta(t_1), \delta(t_2), \dots$ . We will use the properties described in the previous section to enable us to see the consequences of what we have done.

Let  $\Delta_0(t)$  and  $\Delta_1(t)$  be defined

$$\Delta_0(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT)$$

And

$$\Delta_1 = \sum_{k=-\infty}^{\infty} \delta(t - kT_0)$$

It will also be useful to define a function

$$x(t) \equiv 1 \quad (-T/2) \leq t \leq (T_0 - T/2), \text{ and } 0 \text{ otherwise.}$$

In terms of the above we may define

$$\Delta_0(f) = FT(\Delta_0(t))$$

and

$$\Delta_1(f) = FT(\Delta_1(t))$$

That is

$$\Delta_0(f) \Leftrightarrow \Delta_0(t)$$

and

$$\Delta_1(f) \Leftrightarrow \Delta_1(t)$$

To relate the fourier transform  $H(f)$  of  $h(t)$  which are continuous functions to that we would obtain by using sampled data, we use the relationship

$$[ H(f) * \Delta_0(f) * X(f) ] \Delta_1(f) = \text{FT} ( [ h(t) \Delta_0(t) x(t) ] * \Delta_1(t) )$$

The reader is urged to examine the enclosed figure reproduced from the Brigham reference. The understanding of the material presented in this figure will justify the assertion made in the previous equation.

## Discrete Transforms

The discrete complex fourier transform (DFT) of a function  $g(kT)$  is  $G(n/NT)$  where these are related by the following

$$G\left(\frac{n}{NT}\right) = \sum g(kT) \exp(-2\pi i n k / N)$$

where  $n = 0, 1, \dots, N-1$ .

The sampling interval is  $T$  and the total sampling period is  $T_0$ . There are  $N$  samples each of the functions  $g$  and  $G$  so that if the time is normalized to the unit interval  $[0,1]$ , i.e.,  $T_0 \equiv 1$ , then  $T = T_0/N = 1/N$ .

The discrete inverse fourier transform is given by

$$g(kT) = \frac{1}{N} \left[ \sum_{n=0}^{N-1} G\left(\frac{n}{NT}\right) \exp(2\pi i n k / N) \right]$$

This may also be written

$$g(kT) = \frac{1}{N} \left[ \sum_{n=0}^{N-1} G^* \left( \frac{n}{NT} \right) \exp(-2\pi i n k / N) \right]^* = \frac{1}{N} [DFT(G^*)]^*$$

For the above relationships to hold (that the discrete transform and the continuous transform are the same) requires:

1. The time function  $g$  is periodic with period  $T_0$ .
2.  $g$  must be band width limited by having values of  $G(n/NT) = 0$  for  $|n| \geq NT/2$

or, stated in another way. The sampling rate must be two times the largest frequency component of  $G$ .

*Discrete convolution* is defined by the equation

$$y(kT) = \sum_{j=0}^{N-1} x(jT)h[(k-j)T]$$

where both  $x(kT)$  and  $h(kT)$  are periodic functions with period  $T_0$ .

The *discrete convolution theorem* is expressed as

$$FT\left(\sum_{j=0}^{N-1} x(jT)h[(k-j)T]\right) = X\left(\frac{n}{NT}\right)H\left(\frac{n}{NT}\right)$$

Or

$$y(kT) = FT^{-1}\left(X\left(\frac{n}{NT}\right)H\left(\frac{n}{NT}\right)\right)$$

*Discrete correlation* is defined as

$$z(kT) = \sum_{j=0}^{N-1} x(jT)h[(k+j)T]$$

z, h and x are each periodic functions, thus:

$$z(kT) = z[(k+rN)T], r = 0, \pm 1, \pm 2, \text{ etc.}$$

The *discrete correlation theorem* is given by

$$x(kT) = x[(k+rN)T]$$

$$z(kT) = \sum_{j=0}^{N-1} x(jT)h[(k+j)T]$$

$$h(kT) = h[(k+rN)T]$$

Therefore,

$$z(kT) = FT^{-1}\left(X^*\left(\frac{n}{NT}\right)H\left(\frac{n}{NT}\right)\right)$$

## The Fourier Transform Software

Object modules to perform wavelet and fourier transforms are to be found in the library flibsci.lib. This library is compatible with the Microsoft Visual C/C++ compiler and linker.

Object modules have been prepared to execute a discrete complex fourier transform. Also supplied are routines to calculate inverse fourier transforms, convolution, and correlation.

The fast Fourier transform routine has the entry point named FFT. There are no subroutine arguments and no function value is returned.

A preparatory subroutine FFTPR is called **first, and only once**, that fills arrays with the appropriate sine and cosine values needed by the FFT routine.

The inverse fourier transform, the auto correlation, cross correlation and convolution of 8, 16, 32, 64, 128, 256, 512, AND 1024, 2048 and 4096 point complex data sets all call the routine FFT. The routine FFTPR and the other auxilliary routines are to be found in the source modules FFTETC.FOR (and FFTETC.C, if desired).

The **inverse fourier transform**

$$z(t) = FT^{-1}(Z(f)) = \frac{1}{N} [FT(Z^*(f))]^*$$

is calculated by performing the fourier transform of the complex conjugate of the function Z(f) and then taking the complex conjugate of the result (by changing the sign of the imaginary part) and then dividing by the number of sample points, N. This process is carried out by the routine IFFT which is supplied in source module form in Fortran and C in the files FFTETC.FOR and .C, respectively.

**Convolution** is defined by the following:

$$z(t) = h(t) * x(t) = \int_{-\infty}^{\infty} x(\tau) f(t - \tau) d\tau$$

where

$$H(f) = FT(h(t)) = \text{the Fourier transform of } h(t)$$

$$X(f) = FT(x(t)), \text{ the Fourier transform of } X(t)$$

and

$$H(f) X(f) = FT(h(t) * x(t)) = FT(z(t))$$

Therefore,

$$z(t) = FT^{-1}[H(f) X(f)]$$

**Correlation** is similarly defined, thus

$$y(t) = \int_{-\infty}^{\infty} x(\tau) h(t + \tau) d\tau$$

$$Y(f) = FT(y(t))$$

$$y(t) = FT^{-1}[H(f) X^*(f)]$$

The routines given in FFTETC.FOR to calculate convolution (CONV), autocorrelation (AUTOCORR) and cross correlation (CROSS) are all written in Fortran (or C if FFTETC.C is used). Each of these calls IFFT and FFT to do the work. The routine IFFT calculates the inverse fourier transform and it in turn calls FFT, which is the only routine written in assembly language.

Remember that the discrete transform yields correct results for continuous functions only as long as two important restrictions are obeyed. These are:

1. The function to be transformed must be periodic with a period equal to the sampling interval.
2. The sampling rate must be more than twice the maximum frequency content of the function to be transformed.

Otherwise, the user is destined to learn more about aliasing, the Gibbs phenomenon and other things than he probably ever wanted to know.

The wavelet transformation has no such restrictions!

## The Forward and Inverse Wavelet Transformation Software

Software routines have been implemented to perform the forward and inverse transformations for the Haar ( $M \equiv 2$ ) and the  $D_4$  wavelets ( $M \equiv 4$ ). These have been written to handle 2, 4, 8, 16, 32, 64, 128, 256, 512 and 1024 data points.

A way to cause a forward wavelet transformation to be performed is to use

```
CALL WAVE(M,JT)
```

where  $N \equiv 2^{JT}$ ; i.e., for  $JT = 3$  then  $N = 8$ , for  $JT = 4$  then  $N = 16$ , etc. The array, F, must have been filled with the data to be transformed before the call is made. The transform is performed and the A and B arrays are filled with the "answers".

The inverse transformation is performed by using

```
CALL IWAVE(M,JT)
```

The B array must have been filled with data to be back transformed. The arrays F and A are calculated by IWAVE. Again, M is the number of wavelet coefficients and  $N \equiv 2^{JT}$  is the number of sampled data points.

## The FLIBSCLIB Fourier and Wavelet Transformation Software

The routine SUBROUTINE FFTPR(NU), included with the FFT software in the library FLIBSCLIB, must be called before any of the Fourier transform routines, that are described below, may be used. Furthermore, it must be called again whenever the value of the variable NU is changed. Unless the value of NU is changed in a program, this routine need only be called once; however, it does no harm to call this routine any number of times.

The number of points utilized by each of the routines is given by specifying the value of NU which is the LOG to the base 2 of the number of sampled data points, NFFT. That is,  $NFFT=2^{*}NU$ .

Each of the routines uses certain data structures, irrespective of the number of data points included in the transform calculation. Clearly, the number of data points is a power of 2. The maximum number of points is 4096; that is, NU is less than or equal to 12.

The data structures (COMMON blocks) that are used in each of the routines is as follows:

```
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
COMPLEX*16 XC
COMMON/FFTT/XC(4096)
COMMON/NFFT/NFFT
```

The values of the sampled data points are to be inserted into and read from the array XC.

The routine SUBROUTINE FFT calculates the (forward) fourier transform on the array of values given in the complex array XC. The results are returned in the array XC, therefore the initial values in the array XC are destroyed in this process.

The routine SUBROUTINE IFFT calculates the (inverse) fourier transform on the array of values given in the complex array XC. The results are returned in the array XC, therefore the initial values in the array XC are destroyed in this process.

The routine SUBROUTINE CONV(G) calculates the convolution of two complex functions, XC and G. The result is placed in XC. G is an array of COMPLEX\*16 values.

The routine SUBROUTINE AUTOCOR calculates the auto-correlation function of the complex function in XC. The result is placed in XC.

The routine SUBROUTINE CROSS(G) calculates the cross-correlation function of the complex functions in XC and G. The result is placed XC. G is an array of COMPLEX\*16 values.

The routine SUBROUTINE HARTLEY calculates the Hartley transform of the real function given in the array XC. The result is placed in the array XC. The array XC is destroyed in this process.

The routine SUBROUTINE IHARTLEY calculates the inverse Hartley transform of the real function given in the array XC. The result is placed in the array XC. The original values of the array XC is destroyed in this process.

The forward and inverse wavelet transform routines use the data structures given here:

```
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
COMMON/WAVELET/F(4096),A(8192),B(4096),C(0:6)
```

These data structures (COMMON blocks) are used irrespective of the number of data points ,N, that are employed in performing wavelet transformations, the type of wavelet used, and the number of data points employed.

The value of N, the number of points used, is calculated by the wavelet transform routines from the value of JX that is given as one of the subroutine arguments. The value of JX is the LOG to the base 2 of the number of points. That is,  $N=2^{**}JX$ . This forces the number of points to be a power of 2.

The value of JX must be less than or equal to 12.

The value of MX must be either 2, for Haar wavelets, or 4 for D4 wavelets.

A COMMON block. /WAVEETC/, is defined within the wavelet transform routines. It is for the private use of the FLIBSCI routines. The values may be read but not modified by anything outside the wavelet transform routines. Its structure is given as follows, for reference only:

```
COMMON/WAVEETC/N,N2,NT,JT,M
```

The routine SUBROUTINE WAVE(MX,JX) carries out the (forward) wavelet transformation on a real function given in the array F held in the COMMON block /WAVELET/. The results are placed in the arrays A and B.

The routine SUBROUTINE IWAVE(MX,JX) carries out the inverse wavelet transformation on a real function given in the array B held in the COMMON block /WAVELET/. The results are placed in the arrays A and F.