Author:
Blackwell, Alison

Title:
The behavioural effects of formamidine pesticides in lepidoptera.

An Investigation into the Fatigue Behaviour of Wood Laminates for Wind Energy Converter Blade Design.

Submitted by Kuo Tsing Tsai
for the degree of Ph.D.
of the University of Bath 1987

....

(Kuo Tsing Tsai)

CONTENTS

Summary

With the needs of the designers of Wind Energy Converter blades in mind, this work sets out to improve the understanding of the fatigue properties of wood and fatigue failure mechanisms. Three important areas were identifiedfor research: (a) the effect of R ratio on fatigue life, (b) the effect of moisture on fatigue life, and (c) the development of cumulative damage laws. In order to understand fatigue damage mechanisms, a computer control and monitoring system called SArGen was developed.

The computer system is designed to accurately control a fatigue machine and simultaneously monitor load, deflections and/or strains for every cycle in a fatigue test. It is also capable of block loading a specimen with a load history consisting of up to 200 changes in load level. The systems includes a load level correction routine to compansate for drift in load levels and can be used for static tests, monitoring load and deflection or strain at high rates of loading.

Constant load flexural fatigue tests were conducted. Most of the tests were on 4-ply laminates of 4mm thick sliced Khaya ivorensis veneers glued with epoxy resin. For comparison, fatigue tests were also performed on rotary cut Khaya ivorensis laminates, solid Sitka spruce and unidirectional and 0/90 compressed Beech laminates.

The tests on the effect of R ratio showed clearly the severity of reversed fatigue stress application at negative R ratios. At an R ratio of 0.5, the mean fatigue strength for 10^7 cycles was nearly 90% of static flexural strength but with fully reversed flexural fatigue (R = -1), the mean fatigue strength was only about 35% of static strength. The constant life diagram for 10^7 cycles showed that the mean stress is best related to the alternating stress by a parabolic function or Gerber line. The results indicate that two distinct fatigue mechanisms are operating for positive and negative R ratios.

The effect of moisture on fatigue life was investigated by fatigueing sliced Khaya specimens at 5%, 11% and 35% Moisture contents. Increased moisture reduced the static strength and increased the slopes of the fatigue curves.

The fatigue properties of sliced and rotary cut Khaya and Sitka spruce at the same moisture content were found to be largely similar. When S-N lines were normalized by static strengths, they were found to be almost identical. Only when the wood structure has been greatly modified, as for the unidirectional and 0/90 compressed Beech laminates, is there a greater slope to the S-N curve.

The minimum fatigue deflection peak and modulus for sliced sliced Khaya and Sitka spruce were found to change in three stages with fatigue cycles which may be classified as primary, secondary and tertiary stages. For tests at R=0.1, when the changes in deflection and modulus was plotted against different load levels, a transition was found to occur at between 65% and 75% of static strength. It is suggested that this transition corresponds to the movement of the neutral axis following progressive compressive damage on the compression side of the specimen. An increase in modulus was also found when the fatigue stress was below 65% of static strength. Various mechanisms are proposed to explain this behavior.

With an R ratio of -1, no increase in modulus was observed, neither was there a transition to the change in modulus. The maximum and minimum deflection peaks changed in opposite directions and the modulus decreased consistently throughout the range of load levels tested. Damage from compressive stresses affects tensile strengths when the stress is reversed even at very low cyclic stresses.

Micro-cellular damage accumulation was observed in Sitka spruce stressed at 70% of the static strength is repeated loading at R=0.1. Compression kinks were observed after only 500 fatigue load cycles and they developed progressively in bands until visible compressive creases were formed. Macroscopically, tensile fractures was observed from beyond 50% of thefatigue life of the specimen.

Some block loading experiments have been performed at R=0.1 and R=0.5. The results suggests that sequence effects may have a major influence on the fatigue life of WEC blades which experience complex loads in the field. The subject of block and complex loading is one which should now receive detailed attention to more accurately satisfy the needs of WEC blade designers using a probabilistic design method.

Acknowledgements

This work was financed by the SERC on a CASE award with James Howden (Glasgow) Ltd., and the Khaya veneers were supplied by Gifford Technology (Southampton) . Their support is gratefully acknowledged.

I wish to express my thanks to my supervisor Dr. Martin P. Ansell for giving me this opportunity to undertake this research. His enthusiasm and guidance in this work has been invaluable. I would like to thank Mr. Robin Jones for his invaluable help and technical skill in specimen preperation, equipment maintanence and jig making. Also thanks are due the technicians in the School of Materials Science for their assistance. For special thanks is Mr. Peter Wakeford, Chief Technician, for always being available when help was needed. I would like to commend Mr. S. Bowman for his genius in electronics without whom SArGen would still be simply a dream.

Grateful thanks is also due to Dr. John Dinwoodie of the Building Research Establishment for suggesting the microtoming technique and preparing the slides seen in figures (10.9) to (10.13). Mr. Duncan Lawrie and Mark Hancox of Gifford Technology must also be thanked for contributing so greatly with some most helpful discussions on the WEC blade design.

Chapter 1

Introduction

In contrast to many engineering materials, wood has a very long history serving man. With the world production of timber at 10^9 tonnes annually (Dinwoodie, 1981), it is still a very important structural material. Despite this high production figure, information on the fatigue behaviour of wood is limited compared to metals or even the new composite materials. This is probably because wood has been used mainly in civil engineering where creep or duration of loading is the dominant design factor rather than fatigue. Even when wood was used in the aircraft industry, fatigue was considered insignificant or at any rate covered by safety factors and factors accounting for creep. It was not until just before the Second World War that fatigue in wood was recognized with its increased use in aircraft such as the Mosquito. At any rate, with the development of light weight metals, the use of wood in aircraft declined after the war.

In the past few decades, the interest in fatigue of wood has been sporadic. When planes were made from ply-wood during the Second World War, the US Forest Products Laboratory performed some tests to determine the fatigue life of various wood species. That work destroyed the myth that fatigue was unimportant. However, interest declined and publications since then have been scattered with most of the work done in Germany and more recently, in Japan.

The current interest in fatigue is due directly to the advent of Wind Energy Converters (WEC). Wood is potentially the most suitable material for WEC blades which for medium and large machines are over 20 meters in diameter. A few medium size WECs with wooden blades have been built with much success (Lark, 1983; Jamieson and McLeish, 1983). The notable ones in the U.K. are the HWP-300, built in 1984 and installed in the Orkney Islands by James Howden Ltd., and the Wind Energy Group's MS-2, which has been operating since 1985. These were commercial prototypes of which over 50 machines have since been exported. The latest machine constructed, the HWP750 has a blade diameter of 45 metres from tip to tip and the possibility of blade diameters of about 100 metres are being studied. In the USA, interest is also strong in wooden blades and a number have also been designed and manufactured (Zuteck, 1981).

Wood has many advantages over other materials for WEC blade applications. Wyatt et al (1983) have made a careful study of candidate materials. To the supporters of wood, its advantages are clear. It is light which reduces the weight of the blades and consequently the fatigue stresses due to gravity loads. Manufacture is simple using vacuum bagging with room temperature cure resins. Where stresses are very high, wood can be conveniently combined with GRP in the blade structure. This is a much cheaper option compared with the exclusive use of expensive composite materials such as GRP or CFRP. However despite these excellent qualities, the use of wood has been tentative due to the lack of design expertise with wood in fatigue. Fatigue data, particularly in the high cycle region is scarce. This research sets out to reduce this lack of data and to provide a sound basis for fatigue design with wood.

Chapter 2

Introduction to Wood as a Structural Material

2.1 Composition and Structure of Wood

2.1.1 Chemical Composition

In any study of a material, a basic understanding of its chemistry and structure is essential. This is especially true of wood because its chemistry and structure is very complex due to its natural origins. Many papers and books have been written on this subject (Panshin and Zeeuw, 1970; Dinwoodie, 1981; Bodig and Jayne, 1982) and only the relevant essentials are mentioned here. Table (2.1) summarizes its basic chemical constituents, the proportions of which vary according to the wood type and species.

The first class of constituents are the polysacharides, the most important in terms of the mechanics of wood. These are subdivided into two distinct groups, cellulose and hemicellulose. Cellulose $(C_6H_{12}O_5)_n$ is a linear polymer built up of glucose units $(C_6H_{12}O_6)$ with oxygen linkages between the 1 and 4 atoms of adjacent units. It is only a moderately large molecule with a degree of polymerization of around 5000 and 10,000. Cellulose molecules in wood are not totally crystalline. The degree of crystallinity is quite high, up to 90%, with regions of complete crystallinity and totally amorphous regions. X-ray analysis suggests that these crystalline regions, called crystallites, are about 60nm in length, 5nm in width and 3nm in thickness. This is much

Table (2.1) Chemical composition of timber. *(Dinwoodie, 1981)*

	% Weight	Polymeric State	Molecular derivatives	Function
Cellulose	40-50	Crystalline highly oriented large molecule	Glucose	'fibre'
Hemicellulose	20-25	Semi-crystalline smaller molecule	Galactose Mannose Xylose	'matrix' " "
Lignin	25-30	Amorphous large 3-D Molecule	Phenyl Propane	'matrix'
Extractives	0-10	Some polymeric; others nonpolymeric	e.g. Terpenes Polyphenols	extraneous

less than the length of the cellulose molecule. Cellulose molecules therefore passes through many crystallites and interlocks to form long slender strands called microfibrils. It is in fact surrounded by hemicellulose and lignin making the microfibrils about 10nm to 30nm in breadth. Within the crystalline and noncrystalline regions, the cellulose molecules also form strong cross-links with each other through its numerous hydroxyl (OH) groups along its length making the microfibril stiff and strong. With such a structure, it is estimated that the microfibrils have a modulus (in the axis of the molecule) of 132 GPa. Also, because of the hydroxyl groups, cellulose is basically hydrophilic.

While cellulose is analogous to fibres in composites, hemicellulose and lignin is analogous to the matrix. Hemicellulose is derived from various sugars (mannose, sucrose, lactose, etc.) and has a highly branched molecular structure. The molecular weights of hemicelluloses are much less then cellulose and this is confirmed by their solubility and ease of removal from wood. There are also many different side groups in the molecular chain and it is amorphous. Like cellulose, it is hydrophilic and in fact, because of its less rigid and amorphous state, it holds much of the moisture in wood.

Lignin is the most chemically complex of the main constituents and it is also very difficult to extract. It is a phenolic polymer having a three-dimensional structure with apparently no ordered molecular arrangement. Lignin results from free radical polymerization of various phenolic substances. Unlike polysacharides, it is hydrophobic. The elastic modulus of lignin has been estimated to be of the order of 2GPa.

Extractives are various oils and other chemicals which can control the durability, colour, odour and taste of wood. They have little or no direct effect on the mechanical properties of wood, only raising its density.

2.1.2 Cell Wall Structure

The arrangement of the constituents of wood may be described as a composite of cellulose fibrils in a matrix of hemicellulose and lignin. The arrangement of these components are however not entirely certain and models have been proposed to define the structure. What is agreed however, is that the cellulose is arranged in a core with hemicellulose and lignin. The core of cellulose is highly crystalline and it is enclosed by an outer layer of semicrystalline hemicellulose and then amorphous lignin (Dinwoodie, 1981). This gradual transition of crystallinity from fibre to matrix results in high interlaminar shear strength which contributes considerably to the high tensile strength and toughness of wood.

The microfibrils form the basic unit in the cell walls. The cell wall itself however may be described as a laminate of 4 plys with different orientations of the microfibrils and a layer for bonding between the cells. Figure (2.1) shows a model of wood structure. The whole structure may be described as made up of five concentric

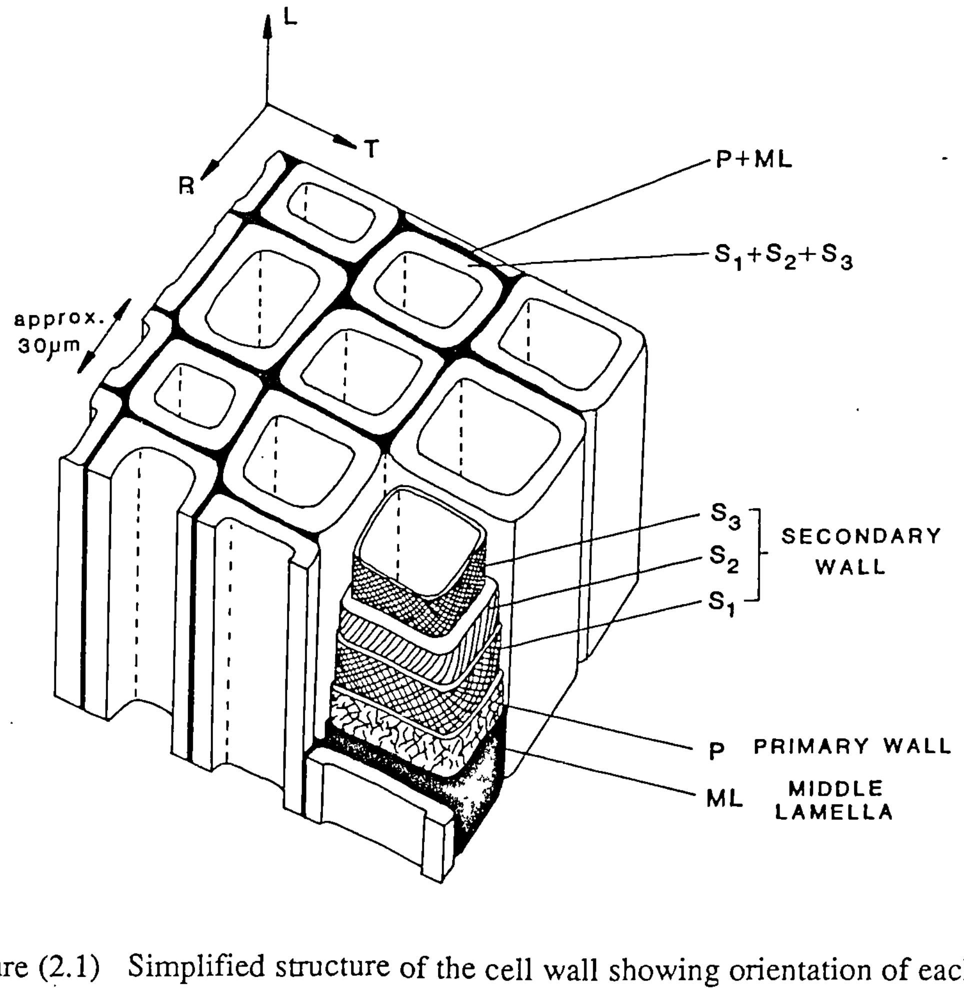


Figure (2.1) Simplified structure of the cell wall showing orientation of each major wall layers. *(Ansell and Tsai, 1984)* cylindrical rings. The directional arrangement of the microfibrils of each cell wall also vary amongst the layers as indicated in the figure. The middle lamella is the bonding medium between the cells being made up of a lignin-pectin complex and is devoid of cellulosic microfibrils. The primary or outer layer is very thin, usually no greater than 0.1μm, has a random arrangement of the microfibrils and usually its contribution to strength is minimal. The other three layers are usually referred to as the secondary wall and form the main structural part of the cell wall. The S2 layer dominates the secondary wall forming over 85% of its thickness and has a microfibrillar orientation typically of between 10° and 30°. It therefore contributes most to the behaviour of wood and properties such as shrinkage, tensile strength and failure morphology can be related to the microfibrillar angle.

2.1.3 Structure at the Macroscopic and Microscopic Level

A tree trunk has to fulfill the functions of support, conduction of mineral solutions and storage of food. Figure (2.2) shows the various parts of the cross-section

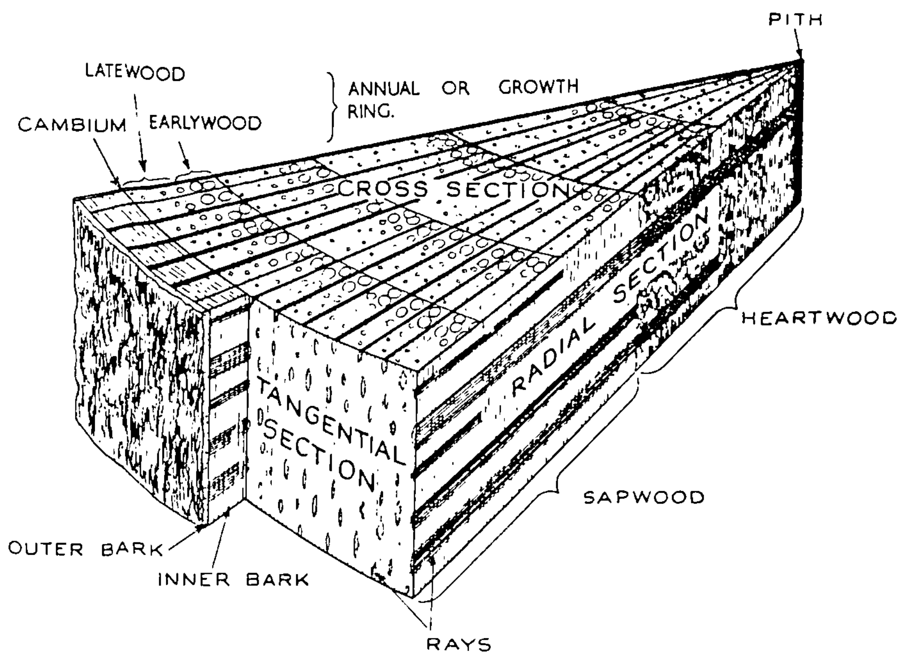


Figure (2.2) Diagrammatic illustration of a wedge segment cut from a five year old hardwood tree showing principal structural features. *(Dinwoodie, 1981)* of a trunk. The entire cross-section of the trunk fulfills the function of support but conduction and storage is restricted only to the sapwood and growth in diameter is restricted to the cambium. The widest and inner part, the heartwood, consists of cells which were once part of the sapwood. With time cell changes occur and the functions of conduction and storage cease. The heartwood is often distinguished by its darker colour due to extractives.

The cambium is a thin layer of living wood cells and lies between the bark and the woody part of the trunk. Growth occurs from this layer. In tropical climates, growth is generally continuous but in temperate regions, climate affects the growth. In the winter, the cambium is dormant but in the spring and through the growing season, the cells divide radially to form daughter cells which further develop to form wood cells or bark. Where the seasons affect growth, annual rings are formed. Wood formed in the early or spring part of the growth season (earlywood) has a low density, but towards the latter or autumn part of the season (latewood), the density increases as growth slows and cells have a greater cell wall thickness. In winter, growth ceases and with the new spring season, a transition between the high density latewood and the low density earlywood is formed.

The radial growth of the trunk must also accommodate the branches. In so doing, knots are formed. Where the cambium of the branch is still alive at the point of fusion with the cambium of the trunk, a continuity in growth will arise although the cell orientation will change. This results in a green or live knot. However if the cambium of the branch is dead, a black or dead knot is formed where no continuity exists. Such knots may drop out during sawing of planks.

At the microscopic level, the structure of wood remains sophisticated. Figure (2.3) shows an electron micrograph of a temperate hardwood, oak. As a three dimensional section, the figure shows the cell types present and their arrangement. The transverse section shows the large vessels present which serve to transport fluid up the tree. Also present are smaller cells for support called fibres. Annual ring interfaces of

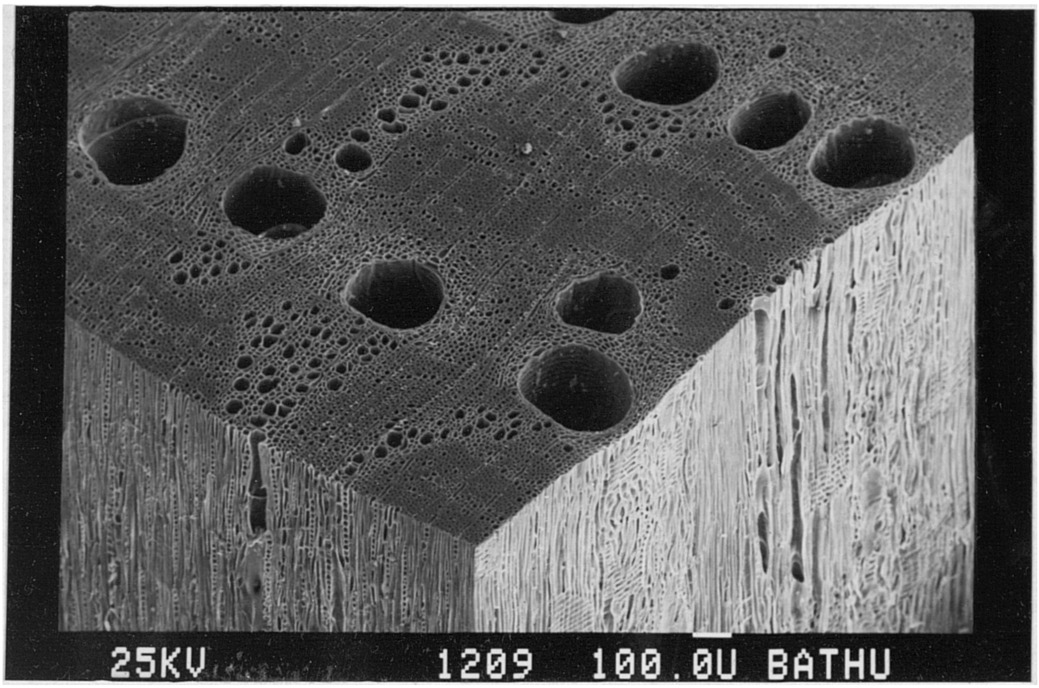


Figure (2.3) Transverse, tangential longitudinal and radial longitudinal sections through English Oak. *(Ansell and Tsai, 1984)*

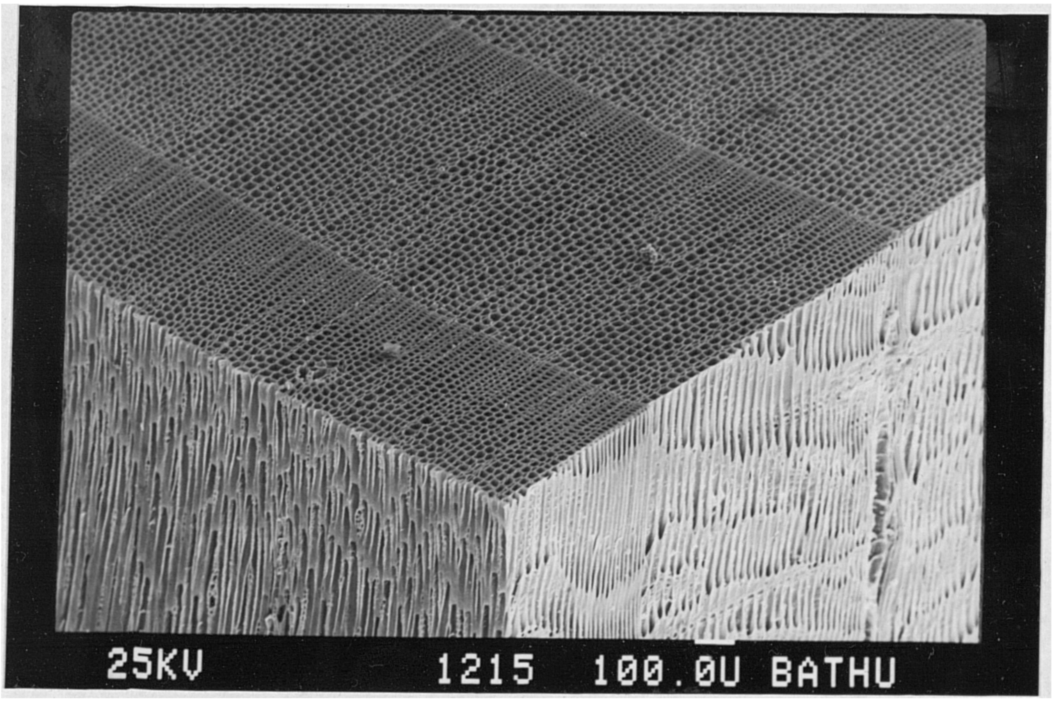


Figure (2.4) Transverse, tangential longitudinal and radial longitudinal sections through Scots Pine. *(Ansell and Tsai, 1984)* varying density can also be seen where the latewood growth ends and the earlywood begins. As seen in figure (2.4), softwood structure is much simpler, without large vessels, and is composed mainly of tracheids. This differ from hardwood fibers in serving both functions of support and fluid transport. Tracheids are in fact also present in hardwoods but in small amounts. In both hardwoods and softwoods, ray cells or parenchyma cells are also found. These cells may be seen in figures (2.3) and (2.4) on the radial longitudinal section lying in the radial direction. They are much shorter with thinner cell walls than tracheids or fibers. They function mainly for storage of food although their presence does affect properties in the radial direction.

2.2 Variability and Defects in Wood

Variability in the properties of wood is perhaps its greatest deficiency. Different species of wood have very different mechanical properties due to genetics affecting cell wall thickness, distribution of cell types, and other factors. This may in fact be an advantage as it provides a wide range of woods to select from. However, variations within each species are also present and can be quite considerable.

With a single tree, variability is systematic. Length of cells, thickness of cell wall, grain angle, microfibrillar angle of the S2 layer all show systematic trends from the centre of the tree to the bark and upwards from the base to the top of the tree. Environmental factors affect properties from tree to tree and are therefore more random. The relation is complex but for example in softwoods, an increased growth rate generally results in a decrease in density and mechanical properties. All these factors contribute to the scatter in any measured property of wood.

However, defects in wood can be much more specific and they contribute most to reducing its strength. They are often due to unusual situations in the environment or may be a product of processing. Reaction wood is particularly important defect in timber used as structural members. When bending stresses are present in growth, as with a tree at an incline or in large branches in hardwoods, the distribution of growth promoting hormones is disturbed causing formation of abnormal tissues. In softwoods, compression wood is formed as the tissue develops on the compression side of the trunk. The tissue is characterized by an abnormally high lignin content, higher microfibrillar angle in the S2 layer and a generally darker appearance. Compression wood is more brittle and has a lower tensile strength. In hardwoods, the growth is on the tension side and is therefore referred to as tension wood. It has a high cellulosic content imparting a rubbery characteristic to the fibres which causes difficulties in sawing and machining. It raises the tensile strength of the timber but reduces its compressive strength.

Another defect particularly significant with Khaya ivorensis and other low density tropical hardwoods is brittleheart. This defect is due to the slight shrinkage of the outer layers of the tree after its formation resulting in it being in a state of tension and the core in compression. This compressive stress builds up as the tree grows resulting in a critical state when the stress exceeds the compression strength and yield occurs. Shear lines form on the cell walls which weaken the timber.

Checks and shakes are defects which greatly reduce the stiffness and strength. They are defined as cracks perpendicular to the grain. Checks usually form from drying stresses while shakes are usually present in trees but appear on processing. Members with these defects when stressed behave as multiple members and have a weak shear strength and low stiffness.

Defects can also be found within the grain. Ideally the wood should have a straight grain but often this is not the case. If the trunk is crooked, diagonal grain will result in the timber or the grain may spiral resulting in a twist in the grain direction. The region around a knot is also not straight grained. Dead knots weaken timber especially and tests have shown the number of knots per unit area of timber has an important influence on strength. Other defects found in timber include compression failures (often the result of tree felling), insect and fungal injuries. These defects can all seriously affect strength and careful selection procedures must be used to avoid unexpected failures.

2.3 Mechanical Properties of Wood

2.3.1 On Determining Mechanical Properties

A comprehensive range of standard tests are available which describe in detail the methods of determining various static mechanical properties. In the U.K. the BS 373:1957 'Methods of Testing Small Clear Specimens of Timber' is available detailing a range of tests. A wider range of tests are also available in the ASTM Standard D143-52. These includes testing methods for measuring the following properties:

(1) 3-point static bending
(2) Compression parallel to grain
(3) Compression perpendicular to grain
(4) Shear parallel to grain
(5) Tension parallel to grain
(6) Tension perpendicular to grain
(7) Hardness
(8) Impact bending
(9) Toughness
(10) Cleavage perpendicular to grain
(11) Nail withdrawal

These tests are based on small, clear specimens requiring careful control of specimen selection. Specimens may either be in the green condition or have a 12% moisture content. Some of the tests are only for comparative purposes, such as hardness and cleavage, and to make practical use of the other test results require adjustments using factors to derive a working stress. The results therefore do not directly represent the strength of larger scale timber, glue-laminated timber and plywood as used in industry. Therefore other ASTM Standard tests are available for testing these directly. The ASTM Designation D198-76 details tests on full -size lumber in bending, compression parallel to grain and tension parallel to grain. These tests, which may also be used on glued-laminated timber, include the effects of defects, moisture content, species, size and other variables in their results which are relevant to the application.

2.3.2 Elastic Deformation

Tested in tension, compression or bending, a specimen of wood will initially deform almost linearly with increasing load but deviations occur at higher loads. Wood therefore follows Hooke's Law in the early part of the test until the point of deviation known as the limit of proportionality. Figure (2.5) illustrates the load-deflection characteristics of timber. In tension, the limit of proportionality is much higher, around 60% of ultimate load, than in compression which occurs at between 30% to 50% of ultimate. With wood, a modulus of elasticity can therefore be defined although its limits must be recognized. It's value is also dependent on the test variables such as loading rate.

Another important feature of timber is its anisotropy. From its structure, its is not surprising that the properties all vary according to the three mutually perpendicular axis, longitudinal, radial and tangential. Ignoring the fact that the tangential face is

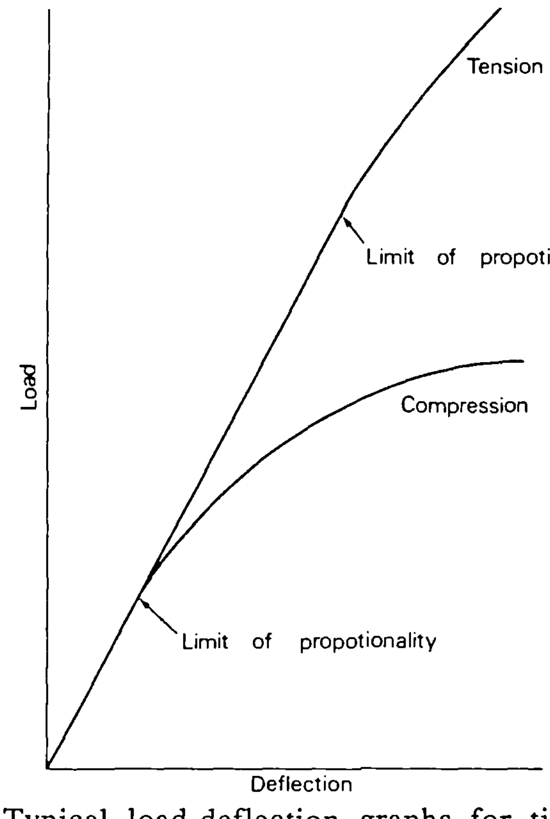


Figure (2.5) Typical load-deflection graphs for timber in tension and compression parallel to grain with the assumed limit of proportionality indicated. *(Dinwoodie, 1981)* curved, timber can then be described as having orthotropic symmetry and the theory of elasticity can be applied. It is not intended here to discuss the theory but to note that considered in this manner, a general description of the deformation of timber under any system of stress is possible. Also, there are only nine independent constants that need be defined, three elastic moduli, one in each of the L, R and T directions; three shear moduli, one in each of the principal planes LT, LR and TR; and three Poisson's ratios namely υ_{RT}, υ_{LR} and υ_{TL}. A comprehensive set of these constants for various species are available from Hearmon (1948).

2.3.3 Factors affecting Strength and Elastic Modulus

Defects discussed in section 2.2 are obvious factors affecting strength and modulus. Other factors which also affect strength and modulus are also present and are of a more general nature such as test environment, material conditioning and treatment. The following is a list of all these factors briefly described.

(a) **Grain angle** : The degree of anisotropy between the longitudinal and transverse planes in timber is as high as 48:1. Therefore depending on the angle of grain of the specimen, strength and modulus will vary greatly.

(b) **Density** : The greater the cell wall thickness, the greater is the density of the wood and the greater the strength and modulus. Figure (2.6) shows the results of many species of wood plotted to relate specific gravity and compression strength. As a means of quality control for a single species, the density-compression strength relationship may be considered to be linear.

(c) **Ratio of Latewood to Earlywood** : Latewood is about 150%-300% stronger than earlywood. This is mainly due to the difference in cell wall thickness but other more fundamental differences in the cells also make it weaker.

(d) **Microfibrillar Angle** : The microfibrillar angle of the S2 layer, measured by X-ray diffraction, markedly affects strength and modulus. The higher the angle is from the axis of the cell, the lower the strength and modulus.

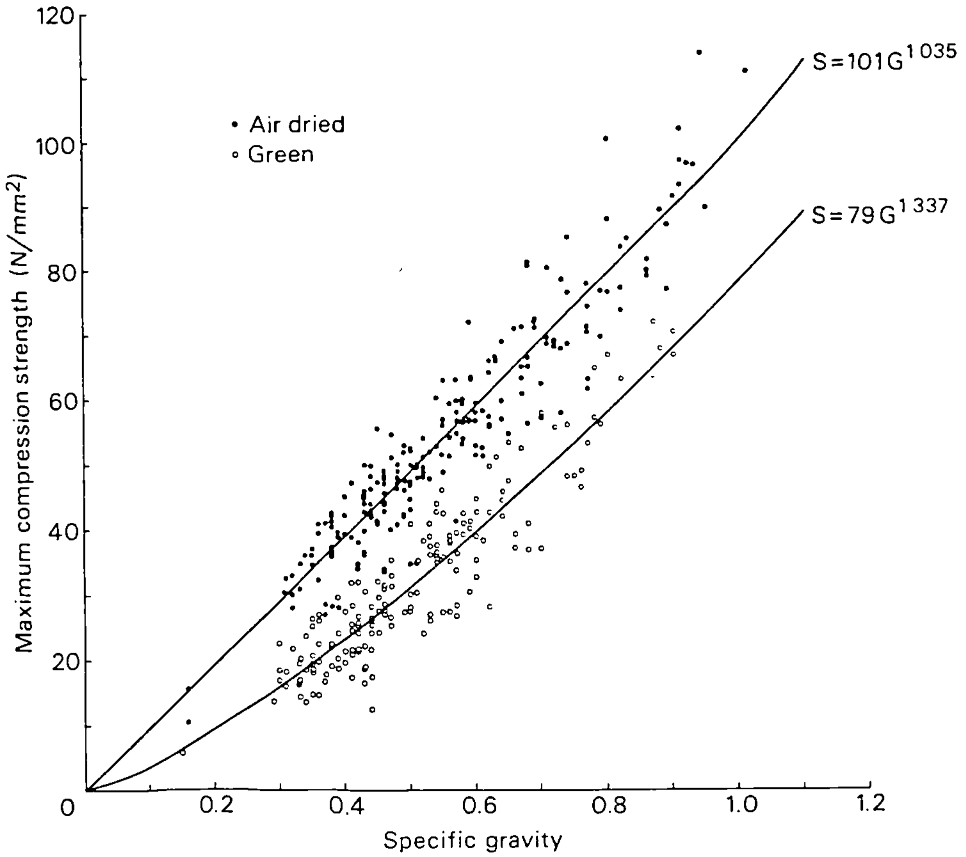


Figure (2.6) The relation of maximum compression strength to nominal specific gravity. *(Lavers, 1983)*

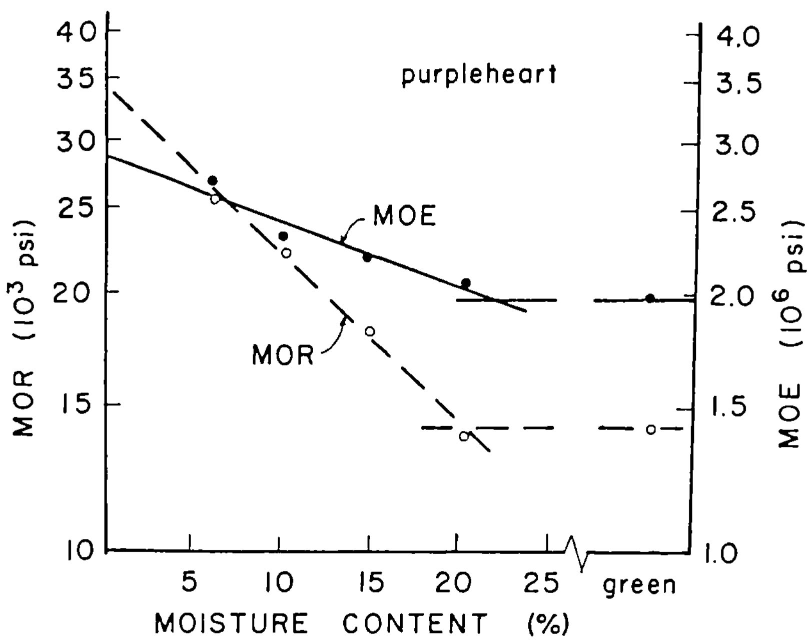


Figure (2.7) Typical linear relationship between log mechanical properties and moisture content. *(Bodig and Jayne, 1982)*

(e) **Defects** : The effect of defects on strength has been briefly discussed earlier. In the application of Wind Energy Converter blades, reaction wood, brittleheart, compression failures, checks, shakes, and other strength reducing defects should be removed by good visual quality control. The effect of knots is more complicated but in general, they should be as few as possible.

(f) **Moisture** : Moisture affects virtually all the physical properties of wood. Strength and modulus as shown in figure (2.7), are greatly reduced with increasing moisture content up to around 20% to 25% above which there is no significant difference. The point of inflection in the graphs is known as the fibre saturation point and is related to the point when any additional moisture in wood is related to an increase in the free water in the cell cavities.

(g) **Temperature** : Between +200°C and -200°C and at constant moisture content, strength and modulus can be considered to be linearly decreasing with increasing temperature. There are complications however in that properties may not be reversible when exposed to above 95°C for short periods, or 65°C, if it is for a longer period of time.

2.4 Failure and Fracture Morphology

2.4.1 Tension Parallel to the Grain

There are four distinct macroscopic types of failure observed in wood loaded in tension parallel to the grain. As shown schematically in figure (2.8), they are (a) splintering tension, (b) combined tension and shear, (c) diagonal shear, and (d) brittle tension. Closer studies using microtensile specimens have revealed a difference in the failure types of the latewood and earlywood. For latewood, a shallow zig-zag fracture plane appears to dominate while earlywood in contrast usually fail with a vertical fracture plane and horizontally across the thin cell walls. Studies by electron microscope of the fracture surface in latewood shows that the fracture occurs either in the S1 layer or as is more common, between the S1 and S2 layers. It appears therefore that shear

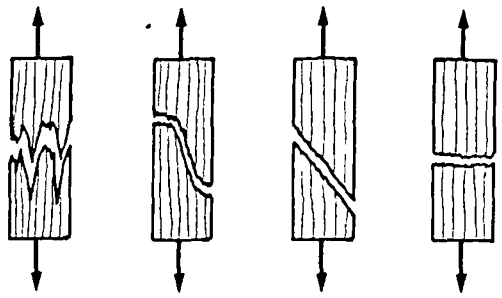


Figure (2.8) Failure types of clear wood in tension parallel to grain: (a) splintering tension, (b) combined tension and shear, (c) shear, and (d) brittle tension. *(Bodig and Jayne, 1982)*

within the cell wall is the dominant mode of failure. The exception is brittle tension which is observed only if the specimen has been compressed prior to failure in tension.

Attempts have been made to estimate the theoretical tensile strength of wood. There are two principal views here; chain scission or rupture of the primary C-O-C covalent bonds (Meyer, 1950), and chain slippage where secondary hydrogen bonds are broken. Of the two models, calculations have shown that chain slippage is unlikely, requiring a much higher failure stress than chain scission. Including factors like finite chain length and the presence of amorphous regions in the model for chain scission, it is estimated that the minimum theoretical tensile strength is of the order of 1000-7000 MNm^{-2} (Mark, 1967).

Considering that the tensile strength of wood is of the order of 100 MNm^{-2}, the theoretical calculations are at least a factor of 10 too high. With the observations made in microscopic studies of latewood and earlywood, the current view of failure in tension is one where failure is initiated by shear. Between the S1 and S2 layers, the opposite orientations of the shear stresses would result in very high shear stresses and it is often observed that delaminations occur there. Calculations by Mark (1967) of the theoretical stresses in the various cell wall layers at the point of failure has indicated that these shear stresses are such as to initiate failure. It therefore appears that both the microscopic observations and developed theories agree that failure in tension is primarily by shear.

2.4.2 Compression Parallel to the Grain

The work of Dinwoodie (1968, 1974, 1978) and Keith (1968, 1971, 1972, 1974) has established a clear picture of the development of compressive damage in wood. Compressive failures have always been recognized in wood because of the characteristic crease formed. It does not lead to total separation of the specimen except at very high strains. In tests, a range of types of failures have been found as shown in figure (2.9). Usually, failure is by shear with the crease easily visible. This often develops gradually as a slow yielding process although high strength specimens sometimes shear suddenly.

Microscopic studies suggest that the yielding process begins at a stress much less than the ultimate. Dinwoodie (1968) suggests damage begins at as low a stress as 25% of the ultimate though Keith (1971) considers damage begins from 60%. Certainly, around 60% of the ultimate strength, a marked increase in damage has been observed corresponding to the deviation from linearity in the stress-strain curve. The compression damage in wood takes the form of kinks or slip lines in the cell walls. This can be observed using polarized light through 20μm thick microtomed section (figure (2.10)). These kinks are irreversible in nature and result from shearing of the cell wall. Often "X-shaped" compression failures are observed as well as "<-shaped" failures.

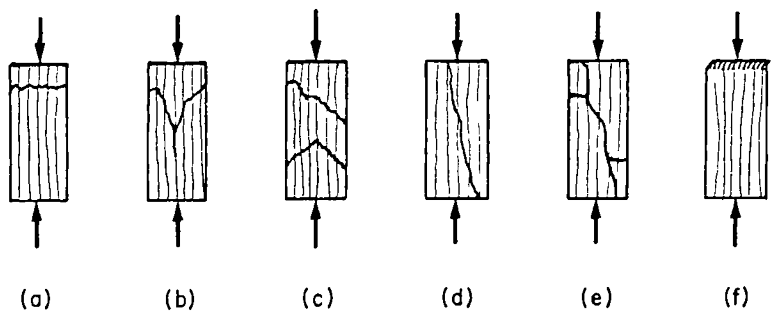


Figure (2.9) Failure types of nonbuckling clear wood in compression parallel to grain: (a) crushing (b) wedge splitting, shearing, (d) splitting, (e) crushing and splitting, (f) brooming and end rolling. *(Bodig and Jayne, 1982)*

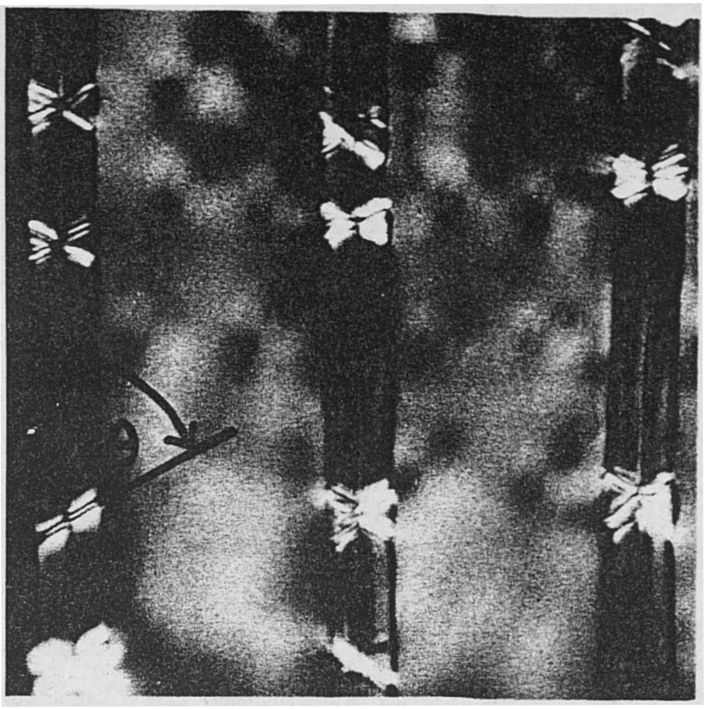


Figure (2.10) Formation of kinks, in the cell walls of spruce timber during longitudinal compression stressing. *(Dinwoodie, 1981)*

As illustrated in figure (2.11), this is a consequence of the direction of the deformation in the individual cell walls of a compound of two which are joined at the middle lamella. With increased stress and strain, the number of kinks increase and become more prominent. The line of kinks formed develops horizontally on the radial plane, but on the tangential plane develops at an angle of 45° to 60° to the vertical. Only at or beyond the ultimate stress is this line of damage visible to the naked eye in the form of the crease.

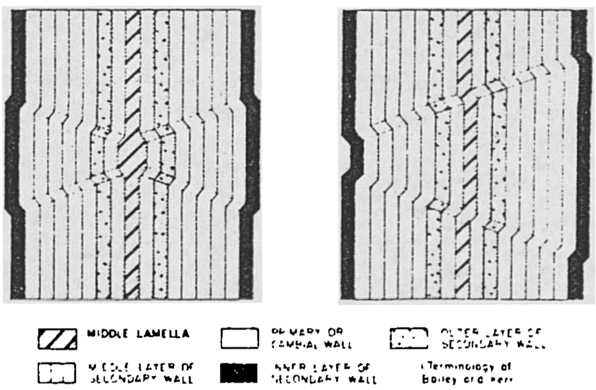


Figure (2.11) Diagram illustrating the appearance of minute compression failures in the compound cell wall viewed in radial longitudinal section. *(Keith,1968)*

2.4.3 Static Bending

In bending, the specimen is subjected to a compressive stress on one side with a tensile stress on the other. At the centre of the specimen, or more accurately along the line of neutral axis, there is no stress. In many materials, the tensile strength is less or about equal to the compressive strength. Therefore, the standard beam formula used in analysis of bend tests, gives strengths approximately equal to the tensile strength of the material. However with wood, the compressive strength of clear timber is only about a third of the tensile strength hence strictly, the beam equation is not applicable but is used to define the modulus of rupture for wood. Also, compression failure would occur well before the tensile strength of the wood is reached. However, since compression failure is progressive, redistribution of stresses must occur as the load is increased beyond the compressive strength. This stress redistribution can be achieved by the movement of the neutral axis towards the tension side increasing the cross-sectional area for the compressive load and reducing the section carrying the tensile load. This would also effectively lower the failure load of the specimen.

Modelling of the stress distribution in a bend specimen was reviewed by Malhotra and Bazan (1980). Using the simplified stress profile shown in figure (2.12),

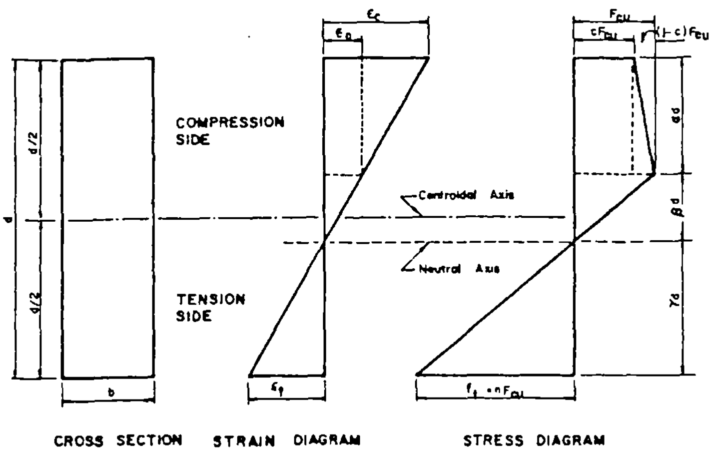


Figure (2.12) Theoretical stress and strain distribution across the depth of a beam of rectangular cross section in the inelastic range. *(Malhotra and Bazan, 1980)*

Malhotra and Bazan derived the following expressions for the ultimate bending moment, M_u and the distance of the neutral axis measured from the tensile face, γ, at ultimate bending moment.

$$M_u = F_{cu}\frac{bd^2}{6}\left[\frac{3N}{N+2}\right] \qquad .....2.1$$

$$\frac{\gamma}{d} = \frac{2N+1}{(N+1)(N+2)} \qquad .....2.2$$

where N is the ratio of the ultimate tensile strength to the ultimate compressive strength, F_{cu}. The specimen dimensions are b, the width, and d, the depth.

2.4.4 Fracture Mechanics

The fracture mechanics approach to strength of materials is well established and developed in its application to isotropic and to a more limited extent, orthotropic materials. In its essence, it proposes that the critical strain energy release rate of a crack propagating through an infinite sheet of a homogeneous material is a constant. Multiplied by the Young's modulus and taking the square root, the term critical stress intensity factor, K_{1C} can be defined which is particularly useful since it is a material property, independent of crack length. Therefore knowing its value, this factor provides the means to calculate the critical flaw size for a material under stress. This approach has been found to be particularly useful for brittle materials which is sensitive to cracks and therefore a definition of strength is subjective.

Its applicability to timber has been much researched and discussed in the literature. The assumptions made in the basic concept of fracture mechanics does not directly apply to wood. Wood is orthotropic and inhomogenous. However, modifications are possible to apply fracture mechanics to orthotropic materials. In the crack opening mode (usually referred to as Mode 1), Atack et al (1961) showed that fracture mechanics is applicable to wood for the fracture planes of RL and TL, ie. for crack extension along the direction of the grain. Schniewind and Centano (1973) measured the K_{1C} values for Douglas fir in all six fracture planes. Two planes, the LT and LR, where cracks were propagated across the grain had values of 2.4 and 2.7 $MNm^{-3/2}$ respectively which was a factor of ten higher than the other fracture planes. However in general, the applicability of fracture mechanics to these two tough fracture planes is poor. For the other two fracture modes, forward shear and transverse shear, research has been very limited.

Chapter 3

Time Dependent Behaviour of Wood

3.1 Introduction

The properties of materials may be considered as two ideal types - the elastic solid and the viscous liquid. However, wood, like most polymeric materials is a neither an elastic or a viscous material. It is not purely elastic as strain continues to increase even when stressed below the proportional limit. It is not purely viscous as viscous liquids have no definite shape and flow irreversibly under stress. Having an intermediate characteristic, wood is termed viscoelastic. As such, the stress-strain behaviour of wood is strongly time dependent.

The time dependent nature of wood properties is in many applications very important. Deflections in beams and other types of wooden members under long term loading is often critical to their performance. Furthermore, failures can occur under sustained loads which are less than their ultimate static loads. Design engineers must therefore have a clear means of quantifying time dependent properties of wood.

Many factors also affect the time dependent behaviour of wood in creep. The magnitude of stress, the rate of stress or strain and the duration of load are all important. The condition of the wood is also important, its moisture content and temperature being the most significant. All these have been studied in many different ways, the most common are:

(a) **Creep.** This is the change in strain with time under a constant stress.

(b) **Stress Relaxation.** A corollary of creep where the stress changes with time under constant strain.

(c) **Duration of Load.** The dependency of failure strength on the length of time under stress.

(d) **Rate of Loading.** The rate of loading affects a variety of mechanical properties of wood.

(e) **Damping Capacity.** The absorption of energy during oscillatory loads or deflection is a consequence of viscoelastic behaviour.

(f) **Intermittent Loading.** A variation of creep and duration of load where the load is held for a period, removed for a period and repeated. This arises out of real life situations where loads are rarely only dead loads.

Much has been published within all these areas as evidenced by the number of review papers on the subject of time dependent behaviour (Schniewind, 1968; Sugiyama, 1967; Grossman et. al, 1969; Grossman, 1976; Ugulev, 1976; Westlund, 1976). Textbooks on wood invariably devote a substantial section on this phenomenon (Bodig and Jayne, 1982; Dinwoodie, 1981). These references and others have been used as the basis of the following review which, while not intending to be extensive, covers the most important and relevant ground.

3.2 Duration of Load

When loaded over a period of time, failure in timber will occur at a stress much less than that for short term tests. This is known in the timber industry as *Duration of load.* It is also referred to as creep-rupture or static fatigue. Such a reduction in strength of wood has enormous implications for the strength of wood that can be used in the design of timber structures.

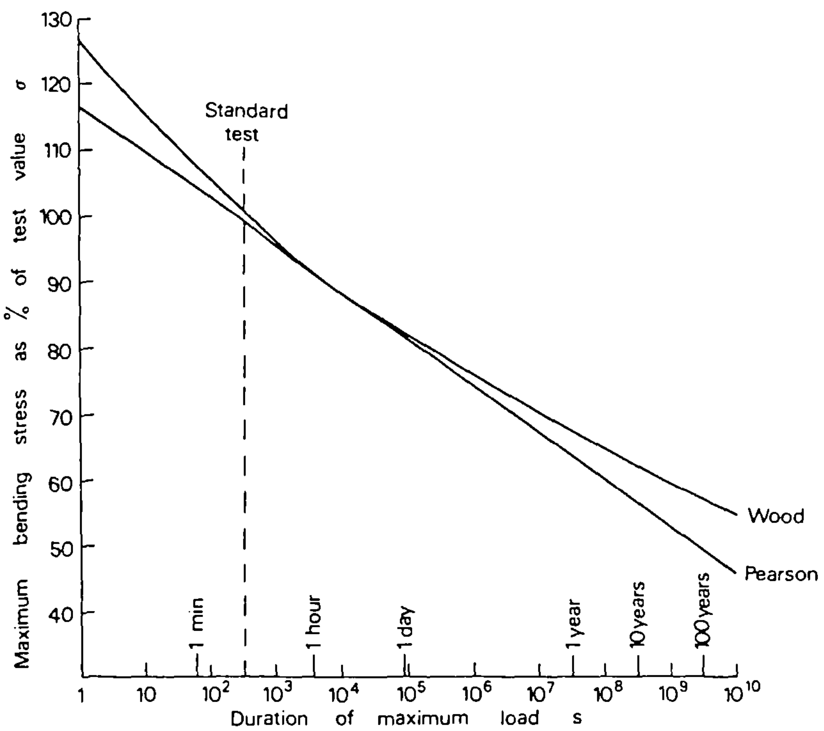


Figure (3.1) The effect of duration of load on the bending strength of timber. *(Dinwoodie, 1981)*

A great deal of experimental work has been carried out to characterize duration of load. The modulus of rupture or bend strength has been found to be nearly proportional to the logarithm of time. Figure (3.1) shows two relationships based on experimental tests from two early papers. The curvilinear relationship established by Wood (1951) is based on a hyperbolic function fitted to test results for small clear Douglas fir specimens. Such a relationship indicates a levelling off at the lower loads suggesting a stress level at which failure will not occur. The linear relationship of strength to the logarithm of time due to Pearson (1972) was derived from previously published results for different species, moisture contents and solid or laminated timber. The regression line through the data is given by

$$s = 91.5 - 7\log_{10}t \qquad .....3.1$$

where s is the percentage stress level and t, the duration of maximum load. Such a linear relation will not have a critical stress level below which failure will not occur.

More recent work with full size lumber has found however that both the relationships of Wood and Pearson do not accurately apply. The work of Madsen (1978), Mindness, Madsen & Barret (1978) and Foschi & Barret (1982) with lumber size specimens suggested that for higher stress ratios, the severity of duration of load is less with times to failure all occurring above the Wood or Pearson lines. The general trend of the results appeared to be opposite to that of Wood suggesting a downturn in duration of load for lower stress ratios. However, longer term tests suggest that the slope diminishes and appear asymptotic to a stress level of about 50%. To model this behaviour, Barret & Foschi (1978) used a damage factor and suggested possible damage accumulation rate functions. Another approach by Nadeau, Bennet & Fuller (1982) uses fracture mechanics and the concept of slow crack growth. This is discussed more fully in Section 3.3 in the context of rate of loading effects.

Where the level of the duration of load varies with time as is common in real life conditions, the direct application of duration of load relationships such as equation 3.1 is not strictly applicable. The use of a high load in order to conservatively characterize the duration of load strength is often used. Gerhards (1979) suggested the use of a cumulative damage approach in the form similar to Miner's Rule for the fatigue of metals. This may be expressed using a residual lifetime factor, g, where g = 0 indicates failure.

$$g = 1 - \sum \frac{t_i}{L_i} \qquad .....3.2$$

t_i is the period under a load SL_i and L_i is the life time associated with that load. No experimental verification of this however is available.

3.3 Rate of Loading Effects

The rate of loading to failure of small defect free specimens of wood has been found to influence its strength, stiffness and proportional limit (Sugiyama, 1967). These properties have been found to increase with rate of loading as schematically illustrated in figure (3.2). This tendency has been shown to occur in tests made in bending, in compression parallel to the grain and other stress states. Sugiyama reported work by Liska (1950) which showed that the strength ratio (in %), P, followed the equation:

$$P = 121 - A\log_{10}T \qquad .....3.3$$

where A=8.5 in compression and 7.5 in bending. T is the time to failure in seconds.

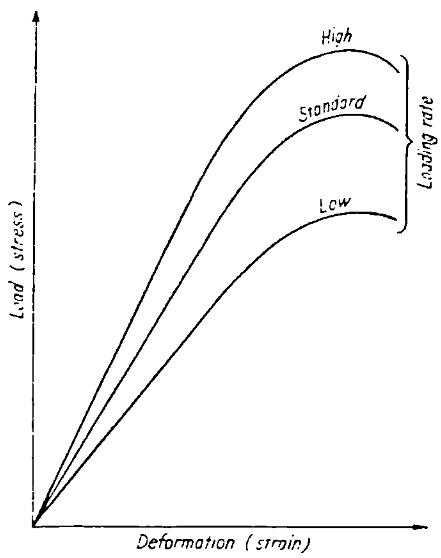


Figure (3.2) Effect of loading rate on stress-strain curve. *(Sugiyama, 1967)*

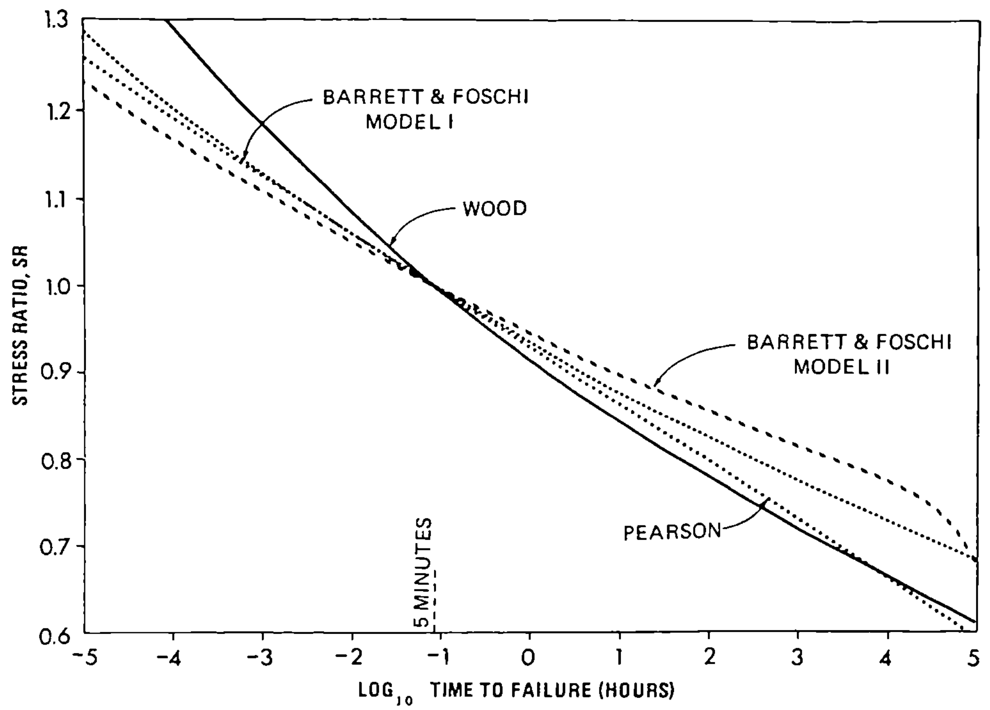


Figure (3.3) Relationships for effect of time to failure on stress ratio for ramp loading. Stress ratio, SR, is defined as the ratio of the failure strength to the failure strength when the duration of stress is 5 minutes. *(Spencer, 1978)*

This implies that the strength of wood will be higher at higher loading rates. Strickler & Pellerin (1973) found in tensile tests on Western Hemlock and Douglas-fir, that only the proportional limit was affected but not the modulus and strength. It should be noted though that the moisture content of their specimens was relatively low at 6% in Western Hemlock and 10% for Douglas fir. Spencer (1978) reviewed various analytical and experimental relationships from the literature. Figure (3.3) shows the behaviour of these relationships reviewed all indicating a significant increase in strength with loading rate.

Recent work (Spencer, 1978; Nadeau et. all, 1982; McLain &Woester, 1986) however has cast some doubt over this simple approach of modelling data from tests with small defect free specimens. Such an approach may be misleading especially since commercial lumber is never defect free. In flexural tests with commercial Douglas-fir lumber of Grade 2 or better, Spencer found that only in comparing the higher strength samples is there an increase in strength with rate of loading. Lower strength specimens showed less increase and may in fact show a decrease in strength. As figure (3.4)

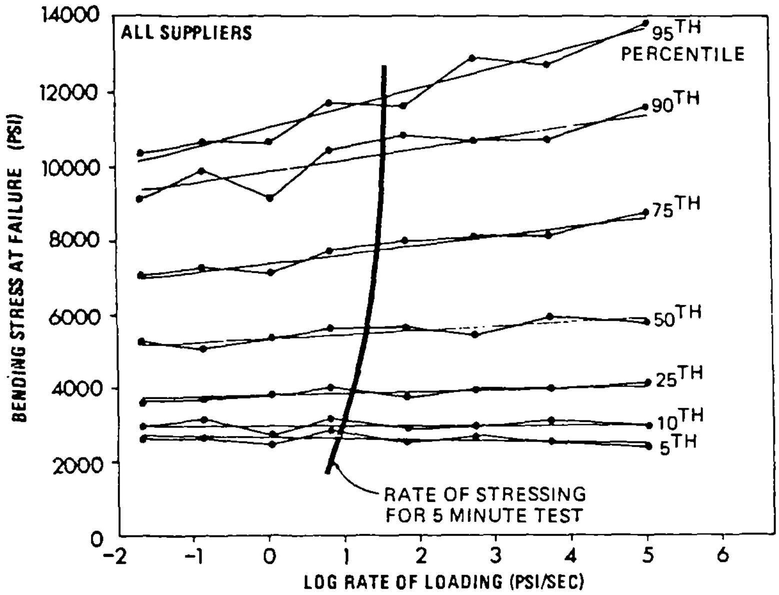


Figure (3.4) Effect of rate of loading on bending strength at different failure probability levels.*(Spencer, 1978)*

shows, at the 95th percentile, there was an increase of about 40% for the very fast rates but at the 5th percentile, the strength appeared to be independent of loading rate. Nadeau, Bennet & Fuller (1982) in testing notched and unnotched Douglas-Fir showed that only in unnotched specimens is there a rate effect. They also showed a correlation with Spencer's result where the notched specimens were of the same mean strength as the 5th percentile strength. Using a simple fracture mechanics model, Nadeau et all were able to predict rate effects for Douglas-fir based on initial strengths. They also suggested an explanation for the observed result showing that there exists two regions in stress-rate behaviour - the higher rate region where strength is independent of rate and the lower rate region where strength is influenced by subcritical crack growth. For lower strength specimens where defect sizes are larger, the boundary between the two regions would shift to lower stressing rates. Therefore short term rate effects will not be found but duration of load effects will still occur since there, crack growth rate is slow. It was acknowledged however that the model is simplistic as it assumes opening-mode fracture when mixed-mode fracture is more the situation. Nevertheless the theory is very attractive.

3.4 Behaviour Under Sustained Loading

3.4.1 Describing Creep and Creep Recovery

When a viscoelastic material is loaded and unloaded, the typical creep and recovery curve obtained is as shown in figure (3.5). Creep is defined as the time dependent deformation under constant load. Recovery follows when the load is removed resulting in a decrease in deformation as a function of time.

If creep causes failure, three distinct stages of deformation can be identified as shown in figure (3.6): primary, secondary and tertiary. The changing strain rates between the three stages suggests a period of stabilization of stress(primary), a transitional period(secondary) before the final failure process(tertiary). The length and degree of these three stages vary greatly depending on the condition of the material and its load level. Generally, the primary stage almost always occurs. The secondary stage can be very extended or very short while the tertiary stage may be nonexistent with sudden catastrophic failure.

In describing creep, three components of the deformation are often defined.

(a) Elastic deformation. This is instantaneous and fully recoverable, ∂_E.

(b) Delayed elastic deformation. This is time-dependent and recoverable, ∂_{DE}.

(c) Viscous flow or plastic deformation. Permanent and nonrecoverable, ∂_V.

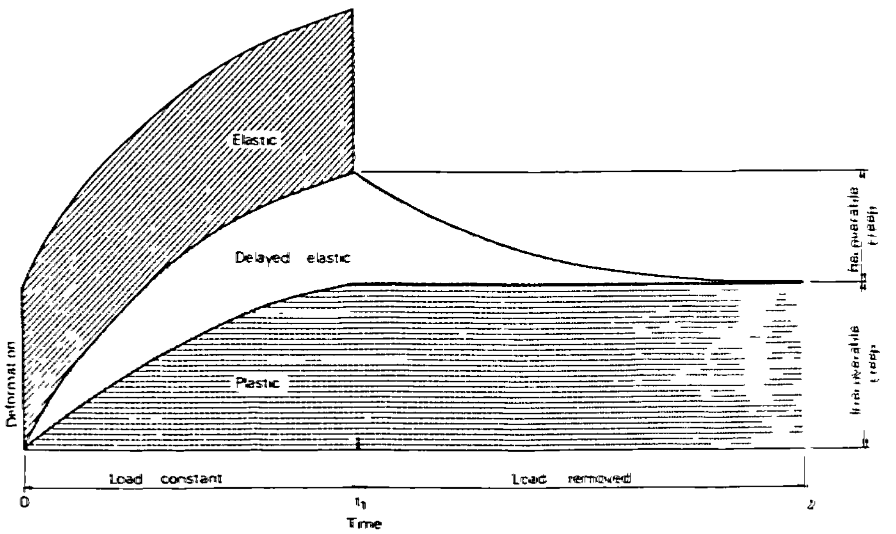


Figure (3.5) The various elastic and plastic components of the deformation of timber under constant load.*(Dinwoodie, 1981)*

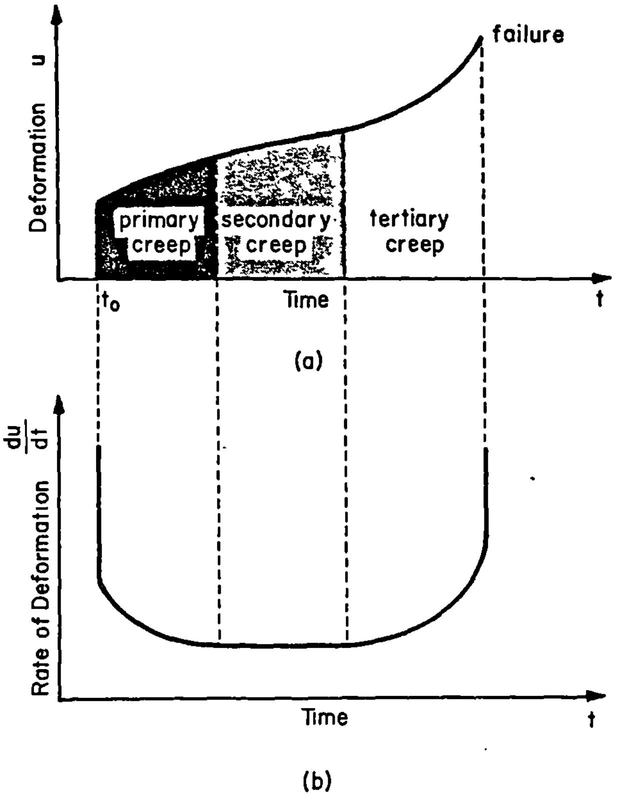


Figure (3.6) (a) Stages of creep. (b) Creep rate.*(Bodig and Jayne, 1982)*

These components are simplistically illustrated in figure (3.5). The most basic form of viscoelasticity is linear viscoelasticity where the elastic and viscous components are considered as linear following Hooke's Law and Newtonian fluid flow. The delayed elastic deformation behaves as a combination of linear elastic and linear viscous behaviour. The behaviour of these three components with wood however is not straight forward. While elastic deformation is generally linear, the viscous component may not be linear. It is sufficient here to describe total creep deformation, ∂_T, as the sum of the three components.

$$\partial_T = \partial_E + \partial_{DE} + \partial_V \qquad .....3.4$$

The concept of a linear viscoelasticity is described in greater detail in section (3.3.4).

In experimental studies of creep, certain parameters and terms may be defined namely creep compliance and relative creep. Creep compliance, also referred to as specific creep, is defined as the ratio of strain (which is time dependent) to the applied constant stress.

$$C_c(t) = \frac{\text{(varying) strain}}{\text{applied constant stress}} \qquad .....3.5$$

Relative creep, also known as the creep coefficient, is defined as a ratio of the time dependent strain over the time independent strain. This may be expressed as

$$C_r(t) = \frac{\partial_t}{\partial_0} \quad \text{or} \quad \frac{\partial_t - \partial_0}{\partial_0} \qquad .....3.6$$

where ∂_t is the deflection at time t and d_0, the initial deflection. It may also be defined as the change in compliance with time expressed as a ratio of the original compliance.

Another term, the stress ratio, used to describe the stress level. It is defined as the ratio of the applied stress level over the static failure strength. In flexure, this would be the extreme fiber stress over the modulus of rupture.

3.4.2 Creep of Wood

Factors Affecting Creep of Wood

The creep trajectories of wood depend greatly on the stress ratio. Figure (3.7) illustrates the different trajectories at different stress ratios in bending. The higher the stress, the higher the relative creep. In his excellent review, Schniewind (1968) reported

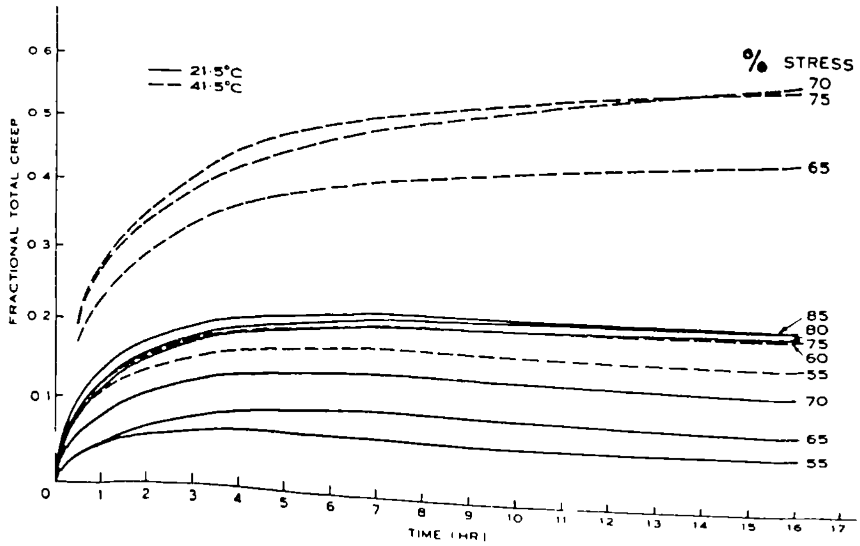


Figure (3.7) Creep curves of Hoop pine at two temperatures and various stress ratios.*(Bodig and Jayne, 1982)*

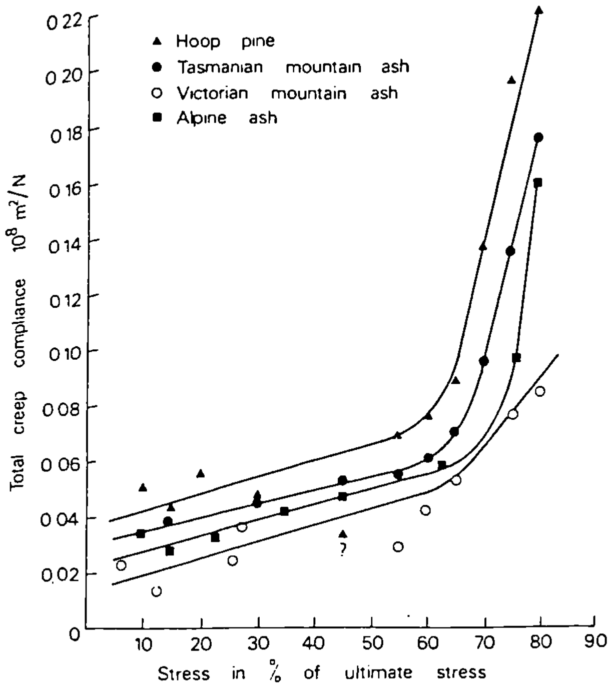


Figure (3.8) The relation of total creep compliance to stress as a percentage of ultimate for four Australian species loaded in bending for 20 hours.*(Dinwoodie, 1981)*

that where moisture content and temperature is constant, at very low stress ratios, wood behaves as a linear elastic solid following Hooke's Law. At intermediate levels however, wood is linearly viscoelastic with increasing deviations from linearity at higher stresses. Figure (3.8) shows the deviation from linear viscoelasticity for various species between 56-60% of flexural strength. For modes of testing other then flexure, a similar deviation has also been found. In tension it has been found to occur as high as 75% of ultimate strength although a great variation of the level of deviation has been reported (Dinwoodie,1981). In compression parallel to the grain, the onset of nonlinearity appears to occur at 70%. The stress level for this is much lower then that in tension although it translates to about the same level as in flexure. Once nonlinearity occurs, much of the increased deformation is nonrecoverable and has been associated with progressive structural change.

As figure (3.7) shows, creep is also greatly affected by temperature. By increasing the temperature from 21.5°C to 41.5°C, the relative creep of Hoop pine almost doubles. In fact, increase in temperature accelerates creep in the primary, secondary and tertiary stages, with failure occurring at shorter times. Significantly, most of the increase is due to the irreversible component. Also, cycling between low and high temperatures repeatedly will result in even greater creep.

Moisture is one of the most important modifiers of wood response. The plasticising effect of moisture invariably increases the creep compliance. The higher the moisture content, the greater the creep compliance. This effect has been observed in tension perpendicular and parallel to the grain, compression, torsion and bending.

If the moisture content of wood is cycled from dry to wet and back repeatedly, the creep deformation will follow a cyclic pattern but with only a partial recovery. Armstrong and Christenson (1961) investigated this phenomenon and found a great increase in the nett deflection with a greatly reduced time to failure. The results of Hearmon and Paton (1964) is shown in figure (3.9). It is significant that the creep of beech at 93% RH tends to remain constant after approximately 15 days while for the same load level but with cyclic changes in moisture, the creep increases by a factor of

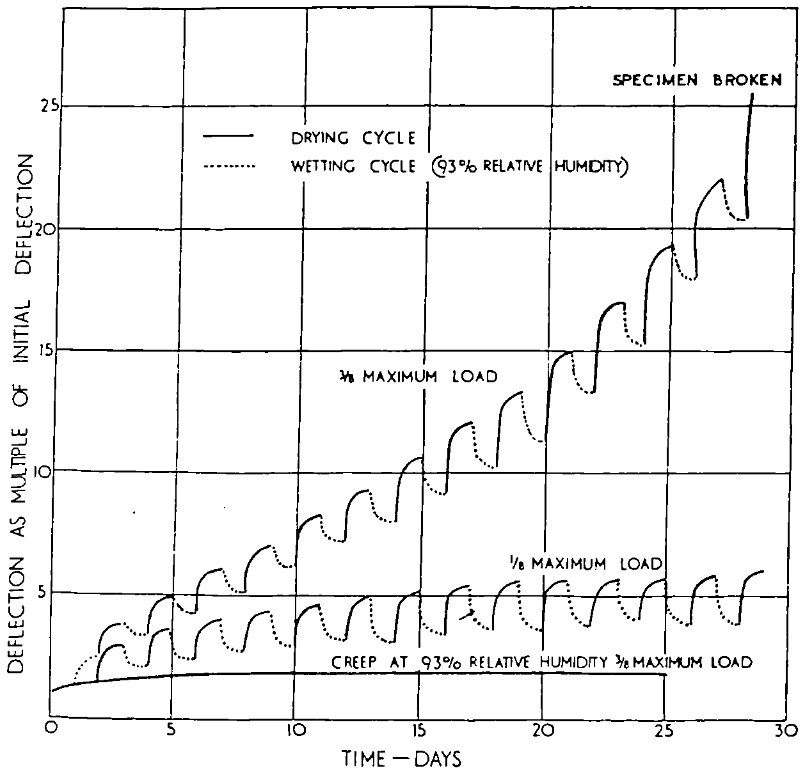


Figure (3.9) The relationship between deflection and length of exposure cycle.*(Hearmon and Paton, 1964)* nearly 20 and is nearly 25 times the initial deflection. Even at a lower load, a cyclic moisture content results in a greater amount of creep deflection. Also for the same load level, failure occurred in less than 30 days whereas the uncycled moisture specimen would not be expected to fail for a much longer period. Another characteristic seen in figure (3.9) is that the creep strains increased during the drying cycle and recovery occurs, though not totally, during the wetting cycle. An exception to this is the first wetting cycle. This exception however depends on the initial moisture content (Schniewind, 1968). Hearmon and Paton (1964) found that the increase in creep deflection is dependent on the range of moisture change and the load level. Grossman (1976), in reviewing this effect reported that the deformation is little affected by the period of the moisture cycle, only its moisture step.

The recovery following cyclic moisture under constant load is also remarkable. Armstrong and Christensen (1961) found that on removal of load, the elastic component recovered immediately followed by a small amount of delayed elastic recovery. Therefore the specimen was still considerably deformed. However, this is not truly permanent as when the unloaded specimen is taken through another moisture cycle or cycles, a large part is recovered.

Boltzmann's Superposition Principle and Time-Temperature Superposition

In most polymeric materials that are linearly viscoelastic, the Boltzmann's principle of superposition applies. This principle states that the total creep occurring by a sequence of stress increments is equivalent to the superposed sum of the creep at each of the incremental stress level. At low moisture content and temperature, this principle has been found to be applicable to wood. However, above the limits of linearity, this principle no longer applies.

The time-temperature superposition principle is also another important principle applicable to many linearly viscoelastic polymers. Here viscoelastic behaviour at one temperature can be related to that at another temperature by a change in the time scale only. However, in Schniewind's (1968) review, this principle has been found to be not applicable to wood. Extensions of the principle to time-temperature-moisture content has been attempted but found to have limited or no application.

3.4.3 Theories of Wood Structure and Time Dependency

As yet, time dependency is far from clearly understood. No comprehensive theory of the processes occurring during creep which results in failure is available. The problem is complicated by the complex interactions of creep with temperature and moisture content. Conceptually, creep processes may be explained according to the elastic, delayed elastic and viscous components.

Elastic strain being instantaneous and fully recoverable must be associated with the straining of molecular bonds. The molecular organization, described briefly in section (2.1), suggests a complex system composed of many different substances with amorphous and crystalline areas. A complete description of all the sources of elastic strain is not feasible, however it is clear that many types of bonds would be involved; both intramolecular and intermolecular. The important idea here is that bonds are not broken or formed.

Following the molecular level of description, delayed elastic or recoverable strains are ascribed to the uncoiling and recoiling of polymer chains. Cellulose, hemicellulose and lignin macromolecules would be involved in this process. In the absence of external forces, a polymer will take a shape which maximizes its randomness and minimizes its free energy according to the laws of thermodynamics. Under external forces, polymers chains will seek to reorientate, breaking and making secondary bonds to establish a new thermodynamic equilibrium. As this occurs, a complex redistribution of stresses would occur in parallel causing other areas to reorientate until total stability is reached.

Cellulose is a more linear molecule than hemicellulose and lignin is highly branched. Therefore, cellulose will reorientate while lignin will slowly flow transferring stress to the cellulose. This will occur until the new orientation of cellulose fully accommodates the stress and the retarded elastic response ceases. On removal of stress, the process occurs in reverse as the polymers seek to return to their original thermodynamic equilibrium with a random, low energy state. While cellulose seeks to return to its original orientation, the lignin will retard the process resulting in a delayed elastic recovery.

Chow (1973) proposed an alternative theory suggesting from experimental observations that the process is a two stage one involving all the three major wood constituents. The first stage concerns the initial response to the application of stress. His experimental studies in molecular motion of the constituents of wood suggested that the carbohydrates crystallize while the lignin reorientates directionally differently to the carbohydrates in the paracrystalline or amorphous regions. The directional difference in molecular movement causes molecular interference between the lignin and carbohydrates to occur which results in stress being transmitted through the lignin network. This reduces the stress burden on the carbohydrates. The second stage follows as the system recovers its equilibrium. Lignin therefore serves as an energy transfer medium and it is postulated, serves also as an "energy sink" maintaining and controlling the energy created by the stressing.

The irreversible strains or viscous component of creep has been associated with the failure and reconstitution of secondary bonds. Once a bond is broken, load is transferred to other areas allowing new bonds to form. The energy barriers present in breaking bonds makes this a time dependent process according to the kinetics of the system. Hydrogen bonds have been identified as the main source of these bond failures and formation. Since the hydrogen bond has water molecules as a crosslink between the carbohydrate molecules, the diffusion of moisture through wood would greatly affect viscous flow. However, it appears that such a mechanism cannot wholly explain irreversible deformation and cannot explain the deformation under cyclic moisture condition. At moderate to high stress levels, it is suggested that the amount of irreversible creep is closely associated with the development of incipient failures. The work of Dinwoodie (1968) and Keith (1971) showed that under compressive stresses, compression kinks develop. The number and severity of the damage depends not only on stress level but also the duration of load. Such compression kinks which develop into compression creases would account for a considerable amount of irreversible deformation for creep in compression and bending.

3.4.4 Modelling Creep

Mathematical modelling of experimental creep test results for polymeric materials have been extensive (Bodig and Jayne, 1982). The application of these models to wood however has been more limited with the Power Law or parabolic equation being the most popular. The parabolic equation takes the form

$$\varepsilon = \varepsilon_0 + at^m \qquad .....3.7$$

where ε_0 is the instantaneous elastic strain, t the time, with a and m, experimentally determined constants. Where instead of strain, relative creep is related to time, the Power Law model would result.

$$\varepsilon_r = At^m \qquad .....3.8$$

This equation has been found to be extremely successful in the modelling of wood especially for primary creep. It is in fact found to fit creep trajectories better than linear viscoelastic models described below (Hoyle, 1985). Hoyle has also successfully used the Power Law to fit creep subjected to cyclic moisture content.

Schniewind (1968) reported that a logarithmic model has also been used for short term experiments in tension parallel to the grain. This model may be expressed as

$$\varepsilon = a + b\log(t) \qquad .....3.9$$

where a and b are constants.

Sugiyama (1967) used two empirical equations for creep in bending above and below the limit of linear viscoelasticity. For creep below the limit,

$$\varepsilon = (a + b\alpha)t^N \qquad .....3.10$$

and for creep above the limit,

$$\varepsilon = a'\alpha^b t^{m\alpha^{-n}} \qquad .....3.11$$

where α is the stress ratio, t, the time, d the % creep and the other terms are constants.

A more classical approach to viscoelasticity is the theory of linear viscoelasticity. The basis of this approach, as suggested at the introduction of this chapter, is the combined use of the Hookean model of a linear elastic solid and the Newtonian model of a linear viscous liquid. Conventionally, the Hookean model is expressed as a spring and described as

$$P_e = ku_e \qquad .....3.12$$

where load P_e is linearly related to elastic deformation u_e with a spring constant k. Newton's law on viscous flow states that load P_v is proportional to the velocity gradient in the liquid, du_v/dt.

$$P_v = r\left(du_v/dt\right) \qquad .....3.13$$

This behaviour is often represented by a dashpot with viscosity r.

By combining these two models, two simple models are possible to describe viscoelasticity. To model the stress-strain behaviour under constant deflection or under stress relaxation conditions, the Maxwell body of a spring and dashpot in series may be used. This is illustrated in figure (3.10a). Under constant stress, this model combines the elastic and viscous components of creep. A Kelvin or Voight body of a spring and dashpot in parallel however is often used to model the delayed elastic strain. This model as illustrated in figure (3.10b), allows a slow transfer of load from the dashpot to the

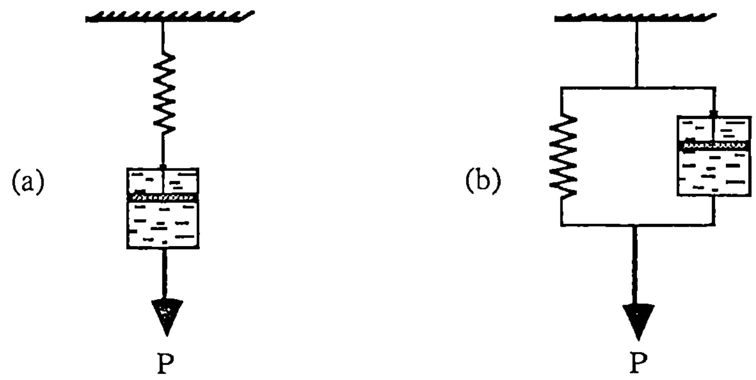


Figure (3.10) (a) Two element Maxwell body for describing stress relaxation.

(b) Two element Kelvin or Voight body for describing recoverable strain.

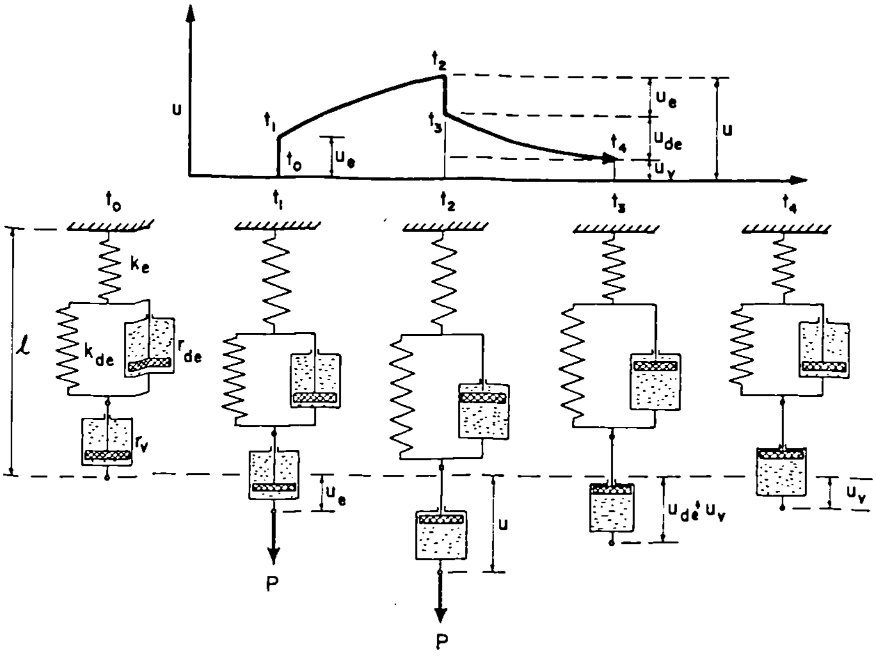


Figure (3.11) A four element burger body representing creep behaviour.*(Bodig and Jayne, 1982)*

spring as the displacement of the dashpot increases. The two bodies can therefore be combined to provide a four element body which includes an elastic, delayed elastic and viscous component as illustrated in figure (3.11). The mathematical expression of the four element body may be expressed as follows:

$$u_{Total} = u_{Maxwell} + u_{Voight}$$

$$= \frac{P_0}{k_m} + \frac{P_0}{r_m} + \frac{P_0(1 - \exp(-k_v t/r_v))}{k_v} \qquad .....3.14$$

where k_m and r_m are the spring constant and viscosity constants of the Maxwell body, while k_v and r_v are those of the Voight body. The ratio, k_v/r_v, is often referred to as the retardation time constant, τ. The experimental fit of this model is reasonable. Improvements can be made to the model by adding more Voight bodies in series extending the expression for creep. However, should the viscous component of creep be non-linear as in compression creep at high stresses (Keith, 1974), such an approach would not be accurate.

3.5 Intermittent Loading in creep

Intermittent loading or cyclic loading is not immediately distinguishable from fatigue. Indeed, the dividing line between the two may be quite arbitrary. Intermittent loading is define here as loading where the time dependency is of greater importance than the number of cycles to failure. Therefore while under load, creep must occur and while unloaded, some recovery must also be present. Limiting the scope further, intermittent loading is only where the loading pattern does not include reversals, whether it be constant or variable and of load or deflection. In other words, the loading pattern cannot include a combination of tension and compression. This limitation is imposed as much of the interest in intermittent loading arises out of situations which in the building industry are not classified as fatigue but rather of as combined 'dead' and 'live' loads .

The response of wood under intermittent loading is dependent on whether it is subject to primary, secondary or tertiary creep. Assuming equal time loaded and unloaded, under primary creep conditions, the recoverable component of creep would dominate compared with the nonrecoverable component. This is illustrated in figure (3.12). The change in residual deformation, Δu, therefore decreases with every cycle.

$$\Delta u_1 > \Delta u_2 > \Delta u_3 > \ldots\ldots\ldots\ldots\ldots > \Delta u_n. \qquad \ldots..3.15$$

Under secondary creep conditions, full recovery of the recoverable creep would occur with only the nonrecoverable component as residual. Therefore,

$$\Delta u_1 = \Delta u_2 = \Delta u_3 = \ldots\ldots\ldots\ldots\ldots = \Delta u_n. \qquad \ldots..3.16$$

Once into tertiary stage, the nonlinear increase of strain would therefore imply that,

$$\Delta u_1 < \Delta u_2 < \Delta u_3 < \ldots\ldots\ldots\ldots\ldots < \Delta u_n. \qquad \ldots..3.17$$

It is important to note that it is possible that the loading cycle will initiate with primary creep, continue into secondary creep, and finally tertiary creep until failure. In which case, a combination of the above will apply. It is assumed also that Boltzmann's superposition principle applies implying a linear viscoelastic response. This means that intermittent load response may be predicted from known creep response.

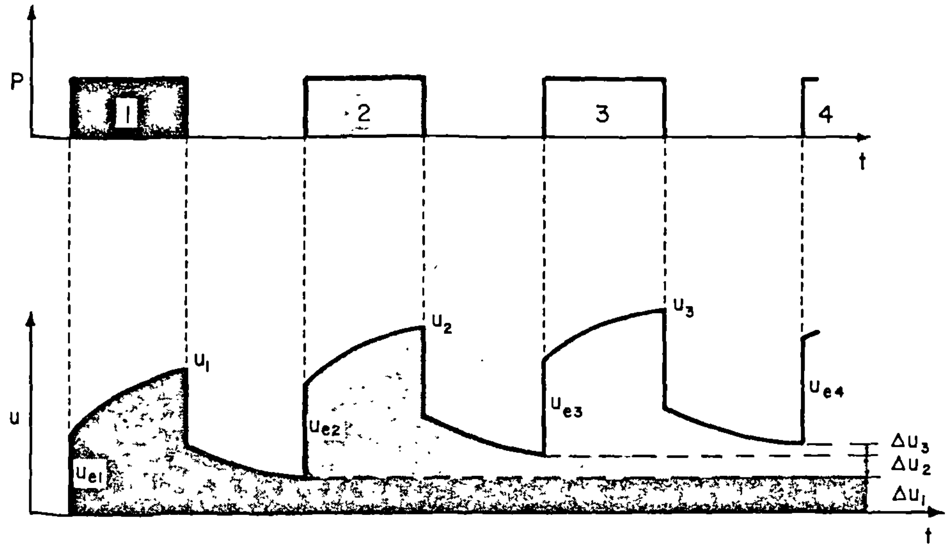


Figure (3.12) Constant load level cycling in the primary creep range.*(Bodig and Jayne, 1982)*

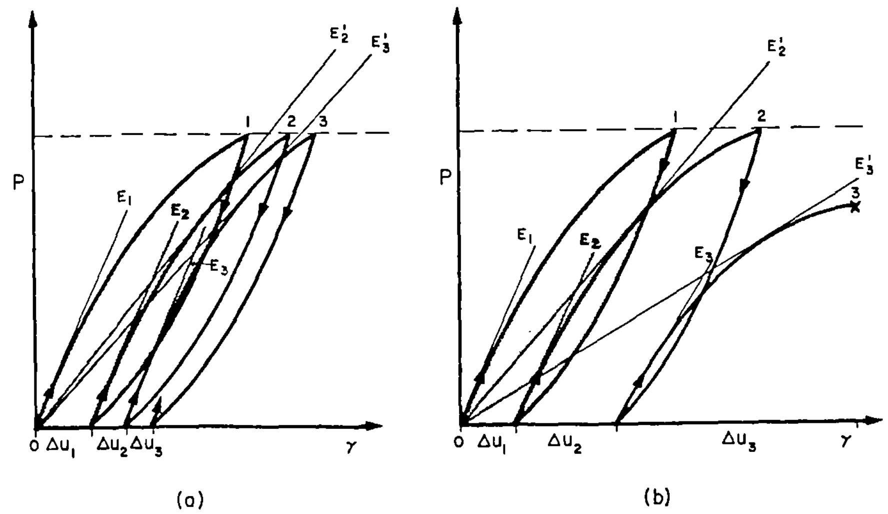


Figure (3.13) Load-deformation relationship in cycling with a constant rate of deformation to a constant load level: (a) primary creep range, (b) tertiary creep range.*(Bodig and Jayne, 1982)*

Should the load pattern, instead of being a square wave function, involve a triangular wave function, ie. a constant load rate, the deformation-time response would be different. However, the change in the residual deformation would still follow equations 3.15 to 3. 17 according to the creep stage. With regards to the changes in stiffness with each intermittent load cycle, there is also a difference depending on the creep conditions. Bodig and Jayne suggests that

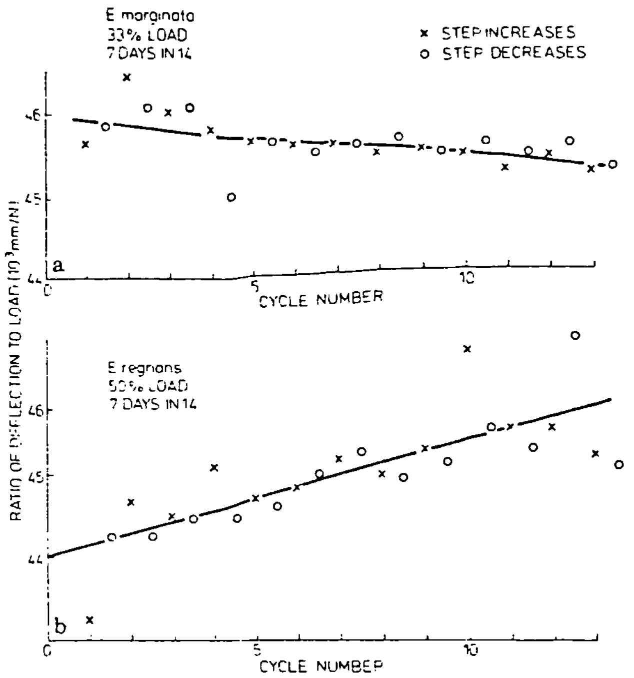


Figure (3.14) Step deflections due to successive applications and removals of load: (a) low stress level, (b) high stress level.*(Nakai and Grossman, 1983)*

$$E_1 < E_2 < E_3 < \dots\dots\dots < E_n. \quad \text{for primary creep} \quad \dots\dots 3.18$$

$$E_1 = E_2 = E_3 = \dots\dots\dots = E_n. \quad \text{for secondary creep} \quad \dots\dots 3.19$$

$$.E_1 > E_2 > E_3 > \dots\dots\dots > E_n \quad \text{for tertiary creep} \quad \dots\dots 3.20$$

The stiffness defined above does not include the residual deformation caused by the previous cycles. If the original starting point for the deformation is used, then the tangent modulus must consistently decrease with every cycle. Figure (3.13) shows the load-deformation relationship in the primary and tertiary stages illustrating the change in stiffness.

Sugiyama (1967) presented results under intermittent load, that indicated that the reduction in modulus of rupture was much smaller than under constant load. The curve joining the peaks of creep deformation was similar to the creep curve for constant load. A difference was also noted above and below the limit of linear viscoelastic behaviour of wood (see section 3.4.2). Nakai and Grossman (1983) found, as shown in figure (3.14), that the change in deflection over the applied load was level or decreased slightly when the load levels were low. At higher loads, the deflection over load increased with each cycle. This suggests that the stiffness of the specimen increased when the load level was low but decreased at a high load consistent with equations 3.18 and 3.20

above. This also pointed to the difference in behaviour above and below the limit of linearity. Their results suggested that Boltzmann's principle of superposition may be used with a fair degree of confidence.

CHAPTER 4

FATIGUE OF WOOD

4.1 Introduction

Considering the complexity of fatigue and the number of papers published on fatigue of metals and composites, very little work has been done on the fatigue response of wood. Publication has been very sparse over the years with interest developing only from the early 1940's. Probably, the view of fatigue of wood prior to World War II can be summed up by Dr. Fokker, the noted aircraft designer, who once stated that "fatigue in properly seasoned wood is unknown" (Lewis, 1960). The total number of publications since 1940 number less than fifty, which probably reflects the decrease in the use of wood for structural members in aircraft. The fact that wood had served well in aircraft such as the De Haviland Mosquito bomber and in troop gliders made such studies appear redundant. In recent decades, much of the papers published on fatigue have come from Japan and some occasional publications from other countries.

Much of the published data on fatigue of wood has been based on constant amplitude deflection tests. The limitations of early test equipment meant that constant load tests were difficult to perform. Constant deflection tests however limit the relevance of the data obtained since wood is susceptible to creep and a reduction in modulus. The peak loads applied to the specimen therefore decrease significantly with time. This can result in a test which can continue indefinitely without visible signs of failure, so the point at which it ends becomes arbitrary. Direct comparison between different published results is therefore difficult. Lewis (1946) has given a clear analysis of the problems of constant deflection fatigue testing with wood. Difficulties in gripping of specimens to avoid localised damage is also a problem. Various specimen geometries have been used. Kommers (1943) attempted to use necked specimens and contoured specimens which gave a constant bending stress from a cantilever arrangement. These were found unsatisfactory as specimens developed longitudinal splits due to shear stresses. Straight sided specimens with wood inserts at the grips were found to be the best. Other configurations used in published results include three point and four point bending but in all cases, some localised damage had to be accepted.

Fatigue tests in the literature are generally described as repeated or reversed fatigue tests. Reversed bending tests refer to tests where the stress level changes between tension and compression with a mean stress of zero. Repeated fatigue simply means that the stress level does not change sign and the minimum stress is nominally at zero. Rotating bending is approximately similar to reversed bending with perhaps some torsion and the entire circular surface is cyclically stressed rather than just the top and bottom surfaces of a flexural specimen.

4.2 Fatigue Life Data for Wood

Despite the few publications, a wide range of different wood and wood laminates have been evaluated in constant deflection fatigue. Kommers (1943) tested in repeated and reversed bending fatigue, Sitka spruce, Douglas fir, five-ply yellow birch and five-ply yellow poplar.The results for all the reversed bending fatigue tests, as shown in figure (4.1), fall in a single band when plotted as a percentage of the static strength. The four types of wood had a fatigue strength of around 27% of static strength at 50 million cycles. Dietz and Grinsfelder (1943) in flexural fatigue tests on 2 and 3 ply birch plywood found fatigue strengths of 25% of static at 2 million cycles. Jenkins (1962) in tests on 4-ply birch plywood obtained similar results. Imayama and Matsumoto (1970) in tests on Sugi using constant load amplitude tests in 3-point bending found the fatigue strength at 1 million cycles to be around 35%. In tests on solid and glued laminated Japanese Cypress bonded with urea or phenolic resin, Ibuki et. al. (1962) estimated fatigue strengths of 15-20% at 10 million cycles.

In rotating bending Fuller and Oberg (1943) found results which superimpose over Kommers' results. Their tests were on maple and yellow birch and they are shown plotted together with Kommers' results in figure (4.1). Maku and Sasaki (1963) tested solid and 4-ply urea or phenolic resin glue laminated wood of Hinoki in rotating bending. The results for all three types were coincident when plotted as a percentage of their respective static strengths. At 10 million cycles, a fatigue strength of around 20-

25% was found. For repeated flexural fatigue, Kommers (1943) found only slightly higher fatigue strengths in solid Douglas fir and Sitka spruce. At 50 million cycles, the fatigue strength was approximately 35-40% of static.

In all the above results, it is clear that there is some uncertainty as to the correct fatigue strength of wood. However, it is also apparent that when plotted as a function of percentage static strength, the S-N data closely coincide. This means that solid wood and laminated wood do not fundamentally differ in fatigue behaviour. Sterr (1963) however concluded differently from results on tests with large sized specimens. He suggested that fatigue strengths of laminated beams were 23% higher than solid beams, density and moisture being equal. The above results also suggest that the type of resin used has no influence on fatigue strengths. Ota and Tsubota (1966,) in a series of papers, on Tanguile laminated with phenolic resin, polyvinylacetate resin and casein glue, concluded that the resin did not affect the fatigue behaviour especially when the repeated deflection was small. Ibuki et al. (1963) however found urea resin marginally better then phenol laminated Japanese cypress.

The form of the fatigue curve plotted as stress against the logarithm of cycles (S-N curve) is also subject to some uncertainty. The results of Kommers clearly suggests that the data is asymptotic to a horizontal line as seen in figure (4.1).

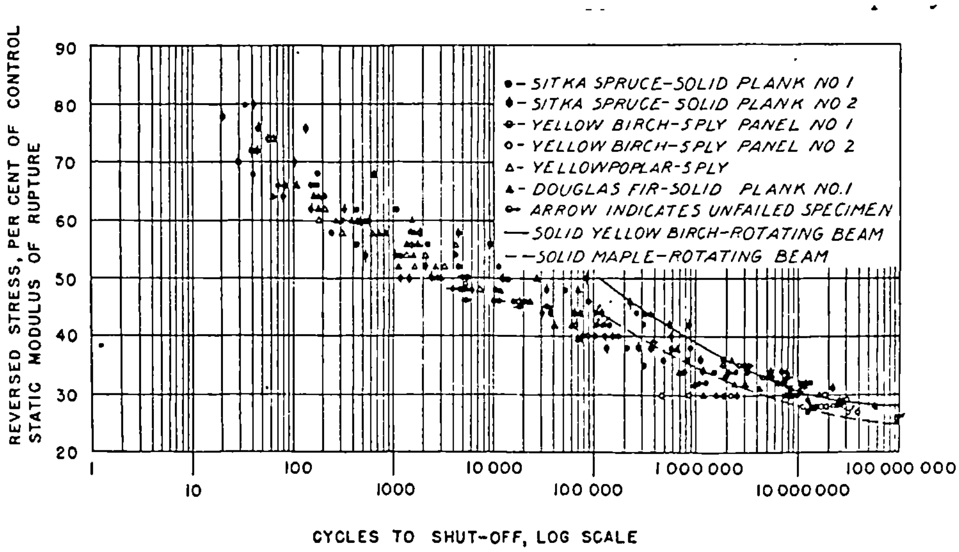


Figure (4.1) Results of tests to determine the endurance of wood and plywood when subjected to reversed bending stress. Included in the plot are rotating bending fatigue data of Fuller and Oberg (1943).*(Kommers, 1943)*

However, the significance of this is not wholly convincing. The use of constant deflection tests means that the peak load levels would decrease during the test. A relatively high test frequency (30 Hz) was used. This would result in some adiabatic heating which would dry up the specimen giving longer fatigue lives. Most of the other published fatigue results are for less than 5 million cycles and linear regression lines were used to describe the data. McNatt (1978) in considering published data on particle boards and hardboards recommended the use of linear regression analysis to extrapolate data to 10 million cycles.

Also of great interest is the effect of joints on fatigue strength. Maku and Sasaki (1963) examined in rotating bending fatigue, various configurations of scarf and butt jointed wood laminates. No reduction in fatigue life was found with scarf joints of all configurations, and results were coincident with unjointed specimens. Significantly, the static strength of all the scarf jointed specimens were about the same as unjointed specimens. Some configurations of butt jointed specimens were however much weaker than unjointed specimens and correspondingly, their fatigue strengths were much lower. However, expressed as a percentage of static strengths, the results were coincident with unjointed specimens. Lewis (1951) in tensile fatigue tests on solid and scarf jointed Douglas fir found no difference in the fatigue curves.

4.3 Factors affecting Fatigue Life

The influence of moisture must be of singular importance in assessing the fatigue properties of wood. However, no comprehensive study on the influence of moisture content on the S-N curve of wood is available. Sekhar, Sukla and Gupta (1963, 1964) showed that at a fixed stress level, in torsional fatigue, a higher moisture content greatly reduced the fatigue life. This is not surprising since a higher moisture content would reduce static properties. If S-N data was plotted as a percentage of static strength, it is unclear whether moisture content would be a factor. Freas and Warren (1959) found that at 50% of static strength, both dry (11% MC) and wet (>30% MC) specimens survived 9 million repeated stress cycles. Lewis (1962) compared green and air-dried southern pine and Douglas fir. The specimens were quarter scale bridge stringers with dimensions 2 by 4 by 43 inches. Green specimens with straight grain were found to have a fatigue strength at 2 million cycles of 50% of static strength for southern pine and 55% for Douglas fir. The result for air dry specimens were slightly higher at 60% of static for both species. However, the green specimens did not fail in 10 million cycles unless the stress level was high enough to produce compression wrinkles in the extreme fibers. The failure mode was different with either progressive compression damage followed by shear or the compression damage reaching such an extent that load could not be sustained. Air dry specimens failed by progressive compressive damage followed by simple or splintering tensile damage.

The effect of temperature has not been investigated in fatigue. However, the frequency of tests can raise the temperature of the specimen through adiabatic heating. Kommers found that in reversed bending (38% of static MOR) at 30 Hz, the rise in temperature can result in moisture loss of 1% in one hour. This must have an effect on fatigue strength. At 40 Hz, Imayama and Matsumoto (1970) found temperature rises of about 5°C until close to failure when temperature rises can be as much as 20°C. No indication as to what the effect would be on the S-N curve is available.

The effect of density has been briefly investigated. Sekhar and Sukla (1979) found the fatigue life at 30% to 45% of MOR increases with specific gravity. The tests covered a wide range of species with moisture contents of around 11% to 15%. Sieminski (1960), found the fatigue strength increased with density. Sapwood was differentiated from the more dense heartwood with sapwood showing equal fatigue strength to heartwood. The proportion of early wood to latewood was also found to have an effect.

Lewis (1962) compared straight grained specimens and specimens with a 1:12 slope of grain. Straight grained specimens were found to have slightly higher fatigue strengths although for green specimens the reverse was found. Air-dried straight grained specimens showed compressive failures followed by final tensile failure while specimens with 1:12 slope of grain usually failed in cross-grain tension without compression. He also investigated the effect of coal-tar creosote preservative treatment on fatigue strength. Treatment not only reduced static properties but, when plotted as a percentage of static strength, also showed some reduction in fatigue resistance.

The effect of notches, holes and checks on the fatigue life has been studied by Lewis (1962), Ibuki et al (1962, 1963) and Maku and Sasaki (1963). Maku and Sasaki showed the stress concentrating effect of holes and surface notches. Most of the fatigue failures were found to develop from these stress concentrations. Lewis tested the effect of artificial checks on Southern pine and Douglas fir. The static properties were greatly reduced by the checks as was the fatigue strength. Expressed as a percentage of static strength for comparison, a small decrease in fatigue performance was found in the checked specimens. Ibuki et al compared the effect of holes of various diameters and side notches on flat plate bend specimens of solid and glue laminated Japanese cypress. Comparison was made using the fatigue notch factor, ß.where ß is the ratio of the fatigue strength of solid (or laminated) specimens to fatigue strength of specimens with holes or notches (the fatigue strength was calculated using the reduced cross sectional area due to the hole). Remarkably, for solid Japanese cypress, the fatigue notch factor was below 1 implying an increased fatigue strength due to the hole or notch. Glue laminated wood had fatigue notch factors between 0.8 and 1.2. Also in general, the specimens with holes of 7.1mm diameter had a greater fatigue strength than those with holes of 2.6mm diameter. The specimens were 24mm in width with centrally located holes. This suggests that wood is not very notch sensitive in fatigue.

4.4 Property Changes During Fatigue

The residual strength of wood as affected by fatigue cycling was first investigated by Kommers (1943). 5-ply Sitka spruce plywood was fatigued without stress reversals for 5000 cycles at various stress levels. Specimens were then statically tested to failure, either in the same direction as the fatigue stress or in the reversed

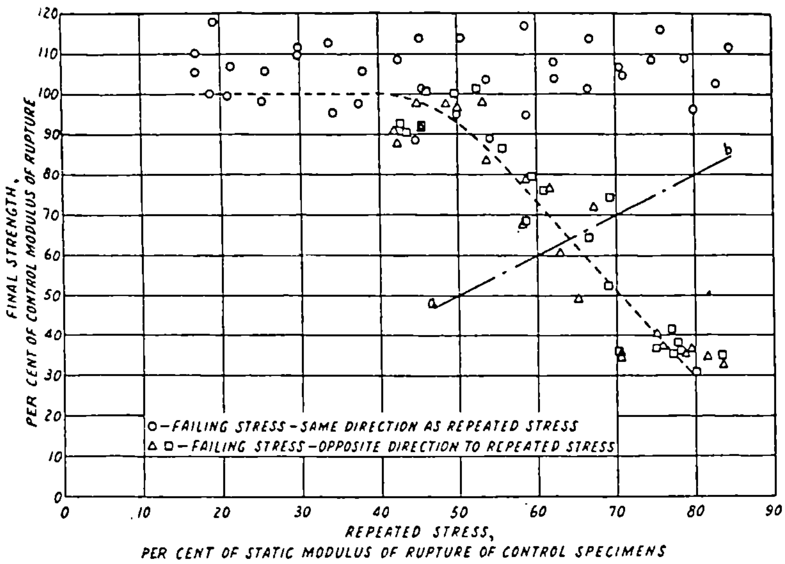


Figure (4.2) Results of tests to determine the effect of 5000 repetitions of stress on the residual flexural strength of five-ply Sitka spruce plywood.*(Kommers, 1943)*

direction. As figure (4.2) shows, when tested in the same direction, even at 85% of static strength, no reduction in strength, indeed possibly an increase in strength resulted. That however may be due to drying out of the specimen as a result of adiabatic heating. The damage in the compression side during fatigue cycling is evident in the data from residual strength tests in the opposite direction to the repeated fatigue stress.

The effect of ten cycles of bending or compressive stress on Sitka spruce and Douglas fir was also studied by Kommers (1943). Although the stress level was at around 95% of static, no reduction in strength was found but the modulus of elasticity decreased with each cycle, the greatest decrease after the first cycle. Kellogg (1958, 1960) found no significant change in tension modulus after 100 cycles except for tests at very high levels of strain. Kommers found that after 5000 repeated fatigue cycles above 80% of the flexural strength, specimens began to show a large decrease in modulus. Imayama and Matsumoto (1970) found a sharp drop in dynamic modulus towards the end of the fatigue life of the specimen. This was combined with a rise in damping close to failure. Rose (1965) however found an increase in modulus with a decrease at higher peak stresses.

While the modulus may remain constant, Kellogg found in repeated tension cycling, the residual strain after each cycle increased similar to the creep of wood under a constant load. Gildwald (1961) also found this behaviour as did Noak and Stockmann (1969). The change in residual strain with number of cycles was found to fit the power law model as expressed in equation 3.7 with t, the time variable, replaced by N, the cycle number. This behaviour was found in the nine species tested and even at very low peak strain levels. Kellogg also showed that the amount of creep was largely independent of species if the level of fatigue strain rather then stress was the same. This is perhaps another indication of the relation between density, modulus and strength.

The temperature changes due to adiabatic heating in the specimen under reversed bending fatigue at 40 Hz was investigated by Imayama and Matsumoto (1970, 1974). They described the temperature change as a four stage process as illustrated in figure (4.3). They found that the slope of stage I increased with the fatigue stress level. Stages II and III dominated the fatigue life of the specimen although for shorter fatigue lives, their proportion of the number of fatigue cycles was less. Microcracks were also observed around the transition between stage II and stage III.

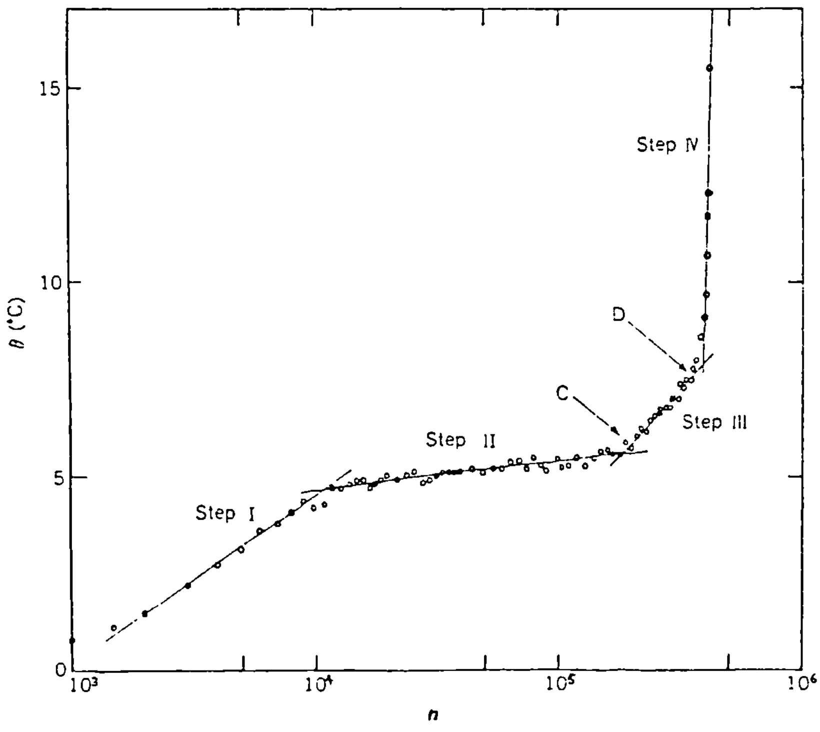


Figure (4.3) The four stage temperature rise due to adiabatic heating under reversed flexural fatigue.*(Imayama and Matsumoto, 1974)*

The acoustic emission from wood during fatigue was studied by Dobraszcyk (1983). He found that at very high fatigue stress levels (90%), the emissions were greatest during the first cycle with emission during the loading as well as unloading. With every subsequent cycle, the emissions were less with emission occurring around the load cycle peak. In comparing the acoustic emission from samples tested to failure after fatigue at various strain levels, the total number of acoustic events decreased with increasing fatigue strains and also the acoustic events began at lower strain levels. Sato, Noguchi and Fushitani (1983) also showed that acoustic events occurred in decreasing quantities with every cycle. Figure (4.4) shows the result of ten tension cycles with the specimen taken to failure at the tenth cycle.

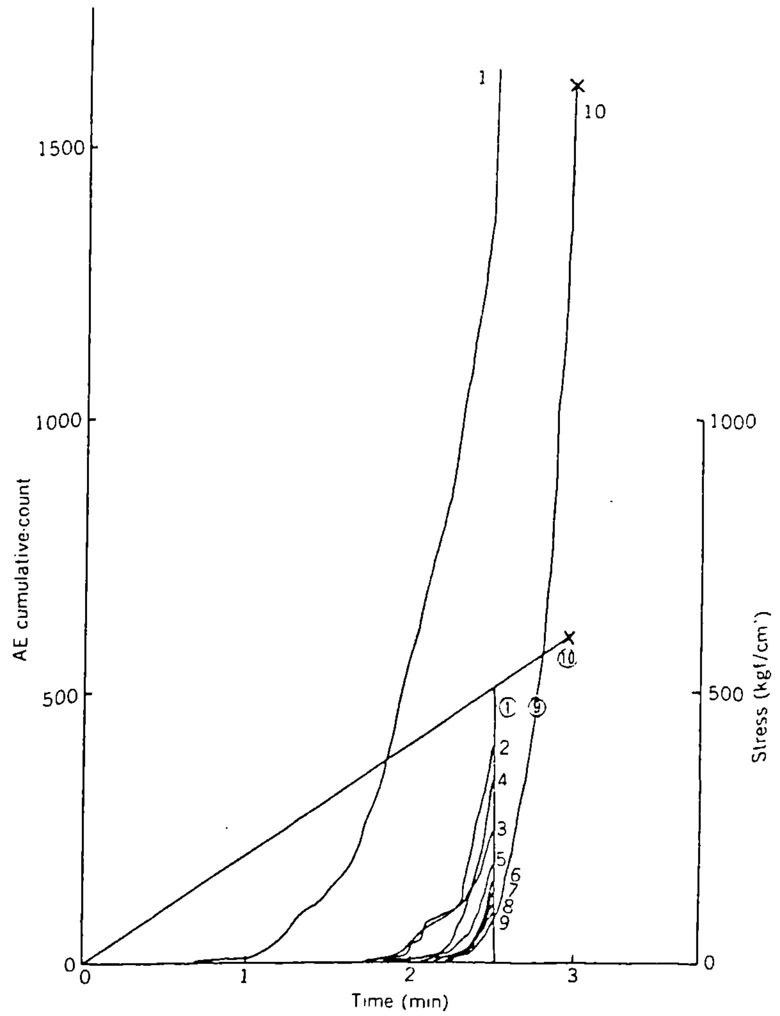


Figure (4.4) Cumulative acoustic emission count for each of the 10 cycles of load with the specimen loaded to failure in the 10th cycle. .*(Sato et al, 1983)*

The many property changes observed in a fatigue test points to a damage mechanism that begins from the very first cycle. The contribution of damage in every subsequent cycle appears to decrease. This is evidenced by a reduced rate of decrease in modulus, a decrease in rate of "creep" strain, temperature change and acoustic emission. There appears to be no correlation of this "damage" with residual strength however suggesting that only in the final damage development is it strength reducing. This is evidenced in stage III and IV of the temperature rise reported by Imayama and Matsumoto. Uncertainties however do exist as the picture of property changes is far from complete, particularly in the modulus and "creep" changes.

4.5 Fatigue Mechanisms

In flexural fatigue, it is generally accepted that the low compression strength of wood implies that in the compression side, some failure would occur. This is most evident in tests on green wood of structural dimensions (Lewis, 1948). In air-dry wood however, final failure is always in the tension side. Maku and Sasaki (1963) also observed compression damage in rotating bending specimens after 10 million cycles. However, apart from structural collapse as observed in green wood, such compression damage is not the mode of final failure observed. Imayama and Matsumoto (1970) observed the development of microcracks on the tension face in three-point flexural fatigue and correlated it with stage III, a stage of steep temperature rise in the specimen. Kollmann and Schmidt (1962) also found microscopic structural damage in fatigued telegraph poles with separation of cells and spiral fractures in the cell wall following the S2 winding direction. It therefore might be suggested that a combination of mechanisms are at work. Most other descriptions of fatigue failure mechanisms such as the extension of the Reiner and Weisenberg's fracture theory by Bach (1973) follow from time dependent arguments although its relevance to a cycle-dependent fatigue situation is debatable.

Chapter 5

Design Considerations for Wind Energy Converter Blades

5.1 The Load Spectrum of WEC Blades

The load spectrum of any machine is by nature complex. Every loading possibility must be considered and an understanding of the sources of loads is important. With Wind Energy Converters (WECs), the load spectrum is complicated further by dependence on their type and location of operation. Two basic configurations of WECs exist, the Horizontal Axis Wind Turbine (HAWT) and the Vertical Axis Wind Turbine (VAWT), Figure (5.1) illustrates some variants of these two basic types. The wind speed distribution varies with location hence even identical machines can have very different load spectrums. These two complications mean that each WEC must be properly analysed or unexpected disasters can easily occur. This research concentrates on the HWP300 designed and built by James Howden Ltd. It is the first of a number of machines sited at Burgar Hill in the Orkney Islands. This is a three bladed, horizontal axis machine of medium size having a blade diameter of 22 meters. The wooden blades of this machine were subcontracted to Gifford Technology Ltd. who designed and constructed them.

Horizontal axis machines of this type experience a number of sources of fatigue

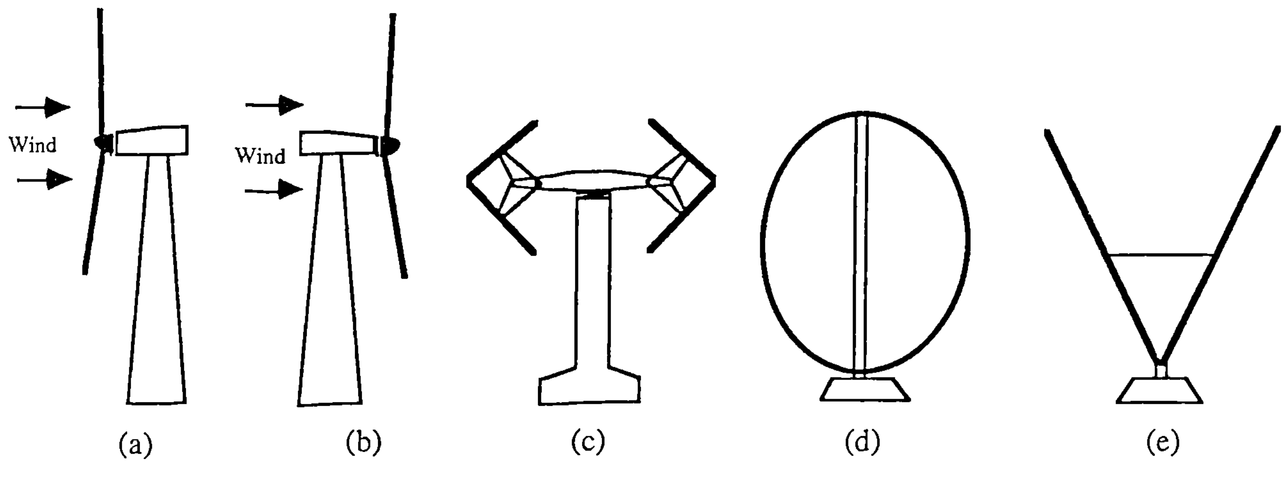


Figure (5.1) Basic Wind Energy Converter configuations. (a) Upwind HAWT, (b) downwind HAWT, (c) Musgrove VAWT, (d) Dairrius VAWT and (e) the Inverted Cone VAWT.

load (Pretlove and Worthington, 1983). These may be considered as deterministic loads and stochastic loads. Deterministic loads arise from many sources. They include gravity loads which induces edgewise bending with the frequency corresponding to the rotational speed of the blade. The HWP300 was found to have very low stresses due to gravity loads. This is directly due to the exceptional specific properties of wood. The most severe fatigue stresses are due to aerodynamic loadings. This induces flapwise bending and some torsion. The stresses here show a few fundamental frequencies. The first is due to wind shear. The wind speed profile from the ground upwards is always in a form of an exponential curve with zero wind speed at ground level and increasing wind speed with height. This means that with every cycle, the blade passes through a region of low wind speeds and a region of high wind speeds. Another fundamental

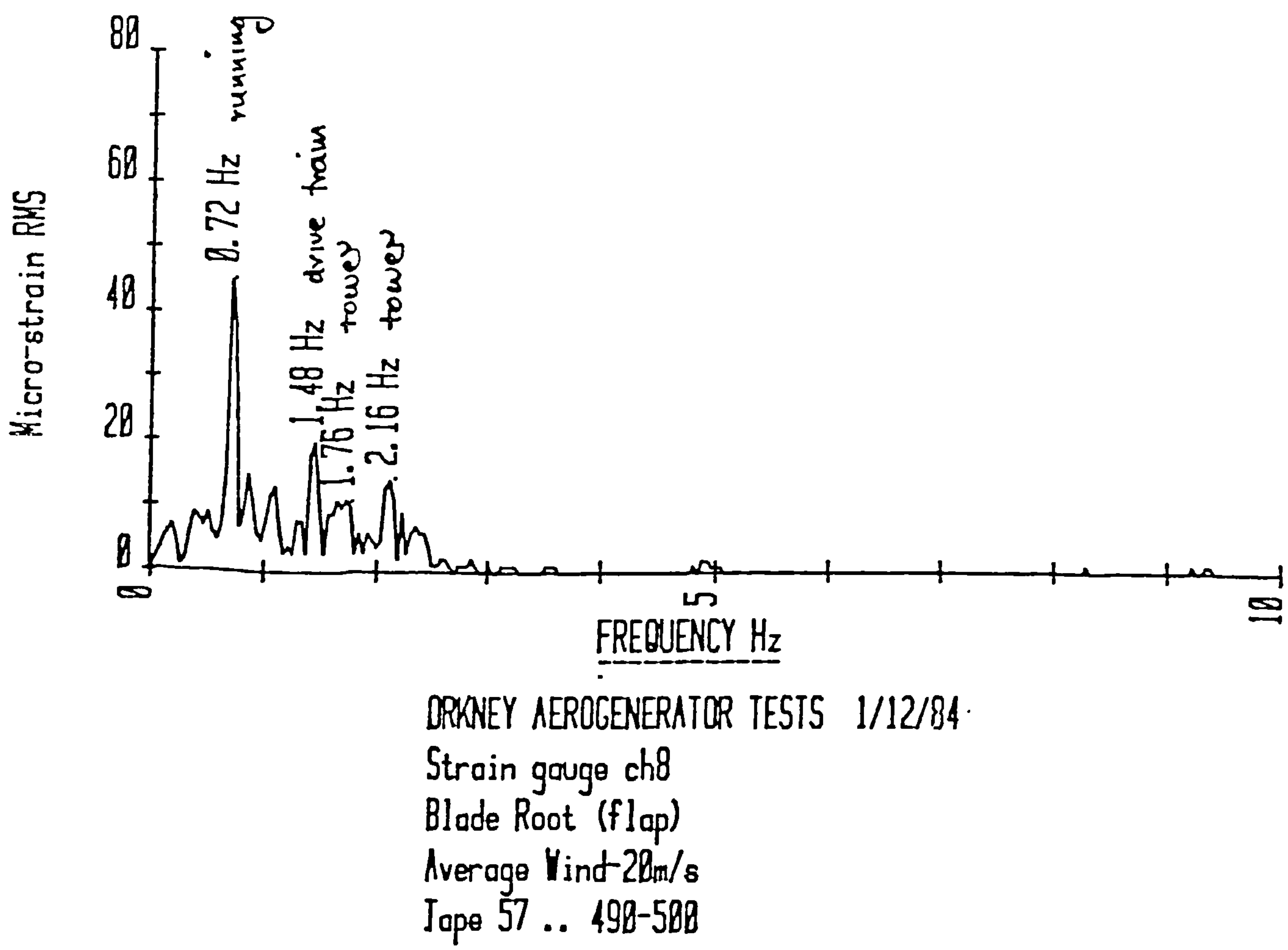


Figure (5.2) Sample frequency spectrum of the flapwise strains near the root of the HWP-300 blade. *(Smith, 1984)* frequency arises due to blade resonance. Every blade has a resonant frequency and careful design is necessary to minimize this effect. The vibrations of the tower add another fundamental frequency. Figure (5.2) shows the frequency spectrum of the flapwise strains (Smith, 1984).

Stochastic loads arise from wind turbulence. While deterministic loads may be predicted from the WEC design, stochastic loads are difficult to ascertain. They are usually estimated from knowledge of the wind spectrum and turbulence of the installation site of the machine. Computer simulation programs of the loadings on the blade are now available and further developed.

The complete load spectrum of a WEC blade must include the operational characteristics on top of the wind speed effects. A WEC does not operate throughout the wind speed regimes but only within a specific band. Above and below the threshold wind speeds, the machine would automatically shut down stopping the blades . With the HWP300, this is achieved by rotating the top third of a blade which results in an aerodynamic stall condition hence slowing down the blades. Disc brakes are also present to bring the blade to a complete standstill. The shut down and start up procedure must contribute its own load characteristics and indeed, the high wind shutdown results in the most severe stresses. The HWP300 also uses its blade tips to regulate its blade rotational speeds so as to generate a reasonably steady voltage. This means that loads are proportional to wind speed and only direct measurements of the entire wind speed spectrum is necessary.

Although the above description of loads on a WEC is specific to the HWP300, the principle of assembling the data from the instrumented blade is general to all machines. The procedure requires,

(1) Obtaining data with the machine operating under different wind speed conditions. This is not straight-forward since wind speeds are not controllable. Continuous monitoring is necessary and careful editing is required.

(2) Transitions in wind speed alter the loading on the blade. Again careful analysis of the load and wind characteristics is necessary. Computer analysis to segment the

entire data collected is required.

(3) Assembling the load spectrum by combining representative parts of the data in proportion to the annual wind speed data.

(4) Operational characteristics such as maintenance shutdowns, or long periods of parking are finally included.

(5) The final step is one of editing the entire load history assembled. This cuts down the time needed to test the specimen to failure while not seriously affecting the predicted life. This may be done by filtering out the very low amplitude alternating loads.

The assembled load history may then be used to estimate the fatigue life of the blade and to assess life prediction models.

5.2 Wood Design Methods

The many years of use of wood in industry and extensive research by forest products laboratories throughout the world, notably in USA and Canada, has resulted in a comprehensive design approach to wood. However, traditional applications for wood are primarily in areas that involve long term static loads and where cyclic loads are small in number. Under such conditions, fatigue may be safely ignored and hence standard design procedures do not have the scope to consider fatigue (AITC, 1974). However, before considering developments in fatigue design, it is instructive to examine static design approaches to examine how fatigue could be incorporated. There are two basic static design standards, the traditional Working Stress Design (WSD) approach as given in the Wood Designers Handbook (AITC, 1974), and the newer probability based design or Limit States Design (LSD) approach currently being incorporated into design codes (Goodman, 1981).

5.2.1 Working Stress Design

This approach attempts to relate laboratory testing to real loading conditions by the use of an array of factors. The starting point is the laboratory tests carried out using small clear specimens of wood selected for freedom of any defects and conditioned to a standard moisture content (i.e. air-dried condition - 10% to 15% moisture content depending on species). From a host of laboratory tests, a certain variation in results will be obtained. This variation is assumed to be characterized by a normal distribution function and to establish the allowable unit stress, the ASTM Designation D245-70 calls for the use of the lower 5% exclusion limit, l_5. This allowable unit stress is therefore the strength value of the wood species based on the probability that only 5% of the sample population will fail at that unit stress.

Although basic variations of wood properties is accounted for in the above, the extension of controlled laboratory test results to visually graded lumber in real use requires further modifications to the test results. This is achieved by using correction factors (Bodig and Jayne, 1982).

(1) Duration of Loading, k_t

(2) Safety factor, k_s

(3) Special condition, k_p

(4) Defects, k_d

(5) Special grading, k_g

(6) Moisture condition of test material, k_c

(7) Moisture condition of lumber, k_f

These factors are established from all the mechanical test data that appear relevant. Where test data is absent (or results of which are not obtainable from handbooks), the experience of the designer and material supplier is relied on.

The duration of load factor, k_t, is the factor used to correct for the time dependent strength of wood. Data suggests that the stress level for a clear wood specimen to last for ten years is 62.5% of the short term strength. Accordingly, 0.625 is the value for k_t. To account for high shorter term loads such as snow loads and wind loads, factors are applied as illustrated by figure (5.3). The stress ratio is based on a ten year load. The treatment then is an iterative process to determine which combination of

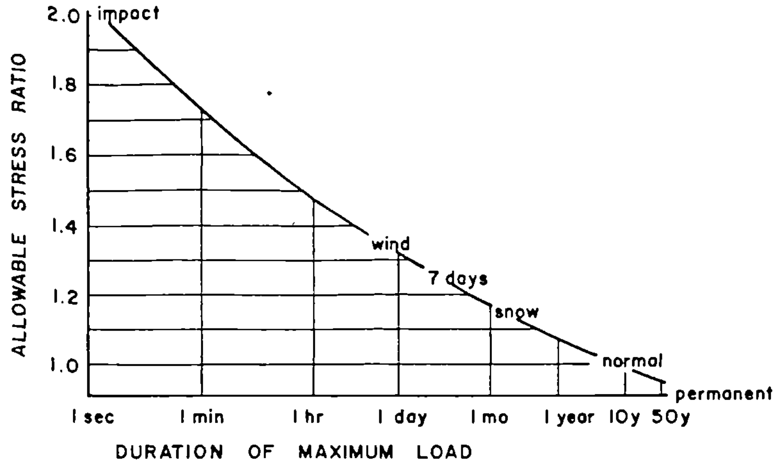


Figure (5.3) Adjustment of allowable stress to duration of maximum load.

(Bodig and Jayne, 1982)

loads, while applying the factor for the load of the shortest duration, would give the largest size member required. It is important to note however, that in the calculations, the modulus of elasticity is time independent.

The safety factor k_s,is a design factor against accidental or unpredictable high loads. The l_5 strength value only accounts for the material variability tested in three point flexure. Table (5.3) lists some correction factors used for softwood lumber. The values for hardwoods are shown in parentheses. The large reduction in allowable load in shear, F_v, is due to the somewhat unpredictable behaviour of wood in shear.

The special conditions factor k_p, is applicable to account for size effects and most importantly, the depth of beam in bending. The presence of defects in commercially graded lumber is accounted for by assigning a particular value of k_d for each grade. Lumber is generally visually graded at various defect densities. The special grading factor k_g, considers the effect on allowable stress of preservative treatments or fire retardent treatment which affects wood species differently. All the factors are applied and their effect may be reflected in the general equation :

$$F = l_5 * k_t * k_s * k_p * k_d * k_g * k_c * k_f \qquad .....5.1$$

where F represents a particular allowable stress (Bodig and Jayne, 1982).

The use of the strength of small clear specimens as the starting point results in

Table (5.3) Correction factors used in the derivation of allowable unit stress values for visually graded lumber as applied to small, clear specimens. *(Bodig and Jayne, 1982).*

Property	Correction Base	Duration of Load, k_t	Safety, k_s	Special Adjustment, k_p	Strength Ratio, k_d	Special Grading, k_g: All	Special Grading, k_g: Medium Grain	Special Grading, k_g: Close Grain[3]	Special Grading, k_g: Dense[4]	Moisture Condition, k_m: ≤4 in. Thick, $M < 15\%$	Moisture Condition, k_m: ≤4 in. Thick, $M < 19\%$	Moisture Condition, k_m: ≤4 in. Thick, $M > 19\%$	Moisture Condition, k_m: ≥5 in. Thick
F_b	$\sigma_{5\%}$	0.625	0.76 (0.70)[2]	0.82	$k_d < 1.0$	1.00	1.00	1.07	1.17	1.35	1.25	1.00	1.00
F_t	$\sigma_{5\%}$[1]	0.625	0.76 (0.70)	1.00	depending	1.00	1.00	1.07	1.17	1.35	1.25	1.00	1.00
F_c	$\sigma_{5\%}$	0.625	0.84 (0.76)	1.00	on the	1.00	1.00	1.07	1.17	1.75	1.50	1.00	1.00
$F_{c\perp}$	$\bar{\sigma}_{pl}$	0.625	1.00	1.00	severity	1.00	1.00	1.07	1.17	1.50	1.50	1.50[5]	1.00, 1.10[6]
F_v	$\sigma_{5\%}$	0.625	0.39 (0.36)	1.00	of defects	1.00	1.00	1.00	1.00	1.13	1.08	1.00	1.00
E	$\bar{E}$	1.00	1.00	0.94		1.00	1.00	1.00	1.05	1.20	1.14	1.00	1.00, 1.02[6]

[1] Tensile strength values parallel to grain are set approximately $\frac{5}{8}$ of MOR.
[2] Values in parentheses are applicable to hardwoods only.
[3] Douglas-fir and redwood only.
[4] Douglas-fir and southern pine only.
[5] Applies to any degree of seasoning below fiber saturation point.
[6] For partially dried lumber.

the use of the many factors which unfortunately lead to a considerable uncertainty as to the level of safety in the design. It would always be desirable to reduce the number of factors required to arrive at the allowable stress F. To do so, it has recently become preferable to test full-size structural lumber or perform ingrade-testing. This alleviates the need for most of the factors and the ASTM Designation D198-76 details the testing procedure. This is however both time consuming and expensive.

5.2.2 Limit-States Design

While the WSD method has been used for many years, its drawbacks have been well recognized and the need for an alternative method has resulted in the development of probabilistic approaches. These methods come under various names; Limit-States Design, Reliability-Based Design, Load-Resistance Factor Design, all being similar. The advantage of this new method is that it seeks to quantify the risk level of the design, in other words, to guarantee that an unacceptable level of in-service failures does not occur (Ang and Cornell, 1974; Aplin and Keenan, 1977) This is a recognition that structural problems are often non-deterministic and that there is a risk involved - no matter how small.

The LSD method begins with a definition of the limit states. A structure may be considered as no longer able to satisfy the requirements for which it was designed in two categories of states (Zahn, 1977):

(1) Ultimate limit state (buckling, rupture of main member, collapse due to fatigue or creep etc.)

(2) Serviceability limit states (excessive deflection, excessive vibration, cracking of non-structural elements etc.)

These two categories of states indicate the nature of the level of risk each state defines.

To quantify the level of risk, the probability of failure, P_f, is calculated - a probability of one indicating certainty of failure and zero indicating impossibility of failure. The required limit state then defines the acceptable failure probability. In the

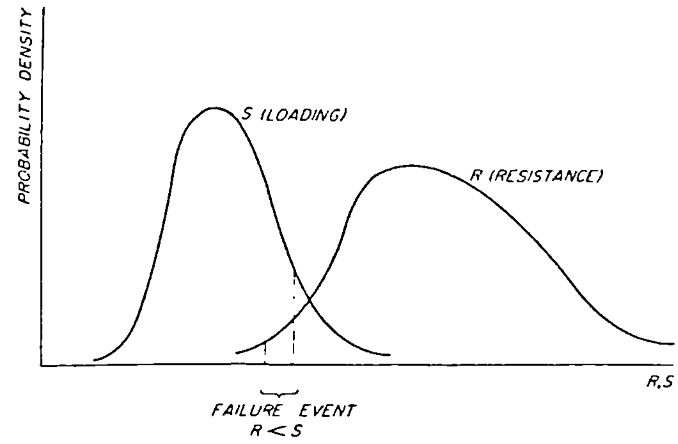


Figure (5.4) Probability density functions of loading and resistance showing typical failure event. *(Zahn, 1977)*

ideal case, the two governing factors for failure are known - the load S, and the resistance (or strength) R. These are defined using the probability density functions f(s) and f(r) respectively as illustrated in figure (5.4). Here R and S are independent variables and hence the probability of failure is given by;

$$P_f = P(R < S)$$
$$= \int \{ \int f(r)dr \} f(s)ds \qquad \ldots .5.2$$

However, in structural design, the exact form of the probability functions is not determinable. The alternative is to assume the form of the probability distribution - normal, log-normal, Weibull, etc., have all been suggested. Using these standard distributions has the advantage also of being able to calculate the moments of the distribution. The simplest technique then is to use the second moment model where only the first two moments of the distribution, the mean and variance, are used. The complete procedure of the second moment model is beyond the scope of this report and a paper by Zahn (1977) contains its full description.

5.2.3 Comparison of WSD and LSD

The WSD method has the advantage of simplicity and a long history of reliability. One suspects that structures built by this method are generally over designed. This may be acceptable in many applications but with WECs, weight is a critical factor and any means of reducing it is desirable. Its chief disadvantage however is that it does not provide the engineer with any indication of the survivability of the product.

The LSD method is new and promises to overcome this disadvantage. It is however inherently complicated to use and a large data base is necessary so as to be able to ascribe a more accurate probability density function. This is important as it is the overlap of the tails of the S and R probability density functions that defines the failure probability (see figure (5.4)). It is also the definition of the tails that is most difficult as they are very sensitive to the size of the data group and the type of probability functioned assumed. Limit-States Design can still be applied and be rigorously logical and semantically pure.

5.3 Fatigue Design

As suggested above, both WSD and the LSD do not have the capacity to consider fatiguein its current state. The most straightforward way of accounting for fatigue is simply the addition of another factor following the WSD approach. Lewis (1960) suggested this while concluding that the current approach with the duration of

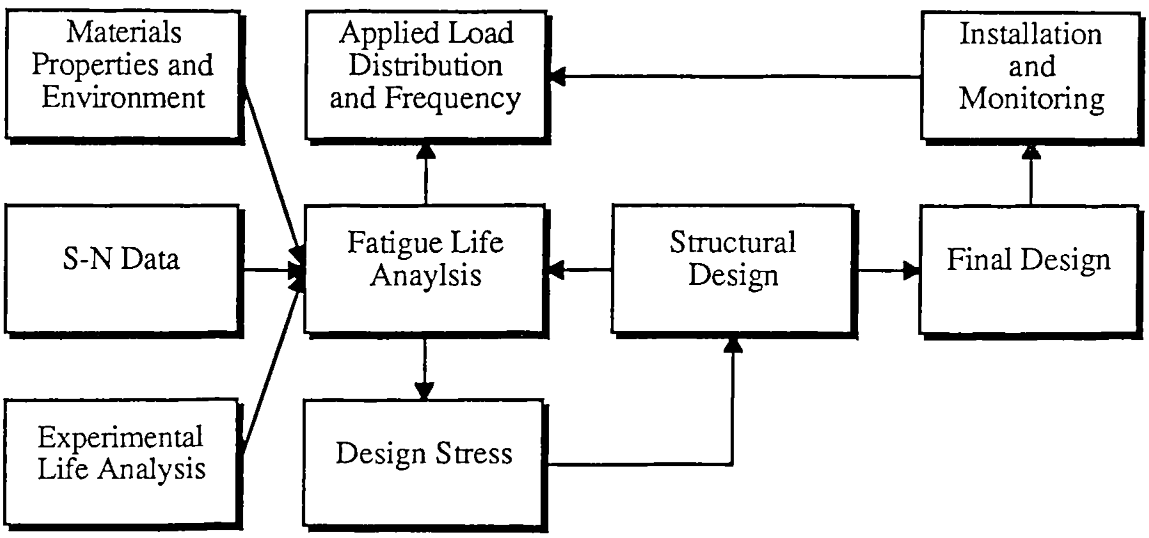


Figure (5.5) Schematic of a fatigue design approach.

load factor would account for fatigue up to 10^7 cycles. The additional factor could be determined by simply extrapolating S-N (applied stress vs. number of cycles to failure) curves to the required number of cycles. This method however would ignore the effect of mean stress and complex loading and it is the only feasible option at present given the current understanding of wood fatigue.

A more comprehensive approach is to develop fatigue design as has been done for metals (Lewis, 1960). Research has also been underway in fatigue design of fiber composites for a number of years and the lessons learnt would also be instructive (Schijve, 1972). Fundamentally fatigue design seeks to relate the real loading conditions of a component to S-N data through a life prediction model. The procedure may be illustrated as in figure (5.5) where the life prediction model is central. Crack propagation methods are the alternative. However, such approaches will not be specifically considered in this research as wood, like composites, is notch insensitive except in the grain direction.

There are various types of approaches to fatigue life prediction each having its advantages. Various references exists which gives excellent reviews on the subject (Osgood, 1982; Schijve, 1972; Gerharz 1982). Osgood (1982) classifies the approaches in three basic groups.

(1) Linear Cumulative Damage based on S-N data or ε-N data and cyclic properties.

(2) Nonlinear Cumulative Damage based on S-N data for each type of specimen configuration.

(3) Cumulative Damage from Damage Boundaries or Modified S-N curves.

Avoiding a description of all the different theories, it suffices simply to look at the most commonly quoted model, the Linear Cumulative Damage model which embodies Palmgren-Miner's rule. The model introduces the idea of progressive damage to the material and uses a damage parameter D, which may be given as,

$$D = \sum \frac{n_i}{N_i} \qquad .....5.3$$

where n_i is the number of cycles at a particular stress amplitude, S, and N_i is the corresponding number of cycles to failure derived from the S-N curve. The damage parameter D, can therefore have a value between 0 and 1 where 1 indicates failure. This model and indeed many other models, have significant limitations. A consequence of using a damage parameter is that the true mechanisms of fatigue damage accumulation are often ignored. It is an empirical parameter and its relevance and reliability is often suspect. This is particularly the case when the Palmgren-Miner's rule is used with composites where an error of 300% has been found (Schijve, 1972). The composite researchers have therefore attempted to incorporate a more mechanistic approach although these theories have yet to be developed to a state where they are applicable to design.

It is impossible to consider all the many facets of fatigue life prediction in the brief examination above. It has only so far been possible to provide a flavour of the difficulties. A list the main considerations involved follows:

(1) Residual strength. This is a useful concept in seeking to follow the strength of the material as it is subjected to fatigue. It provides some physical significance where as damage parameters do not.

(2) Fracture mechanics. With metals where mechanisms are conceptually well understood in terms of crack initiation and propagation, fracture mechanics is particularly useful. Its relevance to wood is debatable but this approach has been suggested (Gerharz,1982).

(3) Reliability. This cannot be ignored and the LSD approach or variations of it can be incorporated in some life prediction models.

(4) Load Characteristics. Any model tends to assume certain load characteristics. The Palmgren-Miner's rule neglects the effect of load sequence and the load history.

To sum up, the wood designer requires a design procedure which accounts for fatigue and provides a measure of reliability. The approach is therefore one which involves fatigue life prediction. This however is a complex subject which requires much research.

Chapter 6

Fatigue Testing: Development of a Computer Control and Monitoring System

6.1 Fatigue Test Methodology

A fatigue test can be conducted in a wide variety of ways with many different variables to consider. Traditionally, fatigue testing was limited by the type of machine available. This meant having to test under conditions of constant deflection, whether it be in axial loading, bending, torsion or a combination of these modes. Today, the wide range of machines available allows almost the entire range of fatigue tests to be carried out. This means not only constant deflection but also constant and variable load or any other denominator such as stress intensity factor in a fatigue crack propagation test. In the context of determining the fatigue properties of wood, the capability for load control is essential although, as noted earlier in Chapter 4, almost all published results to date are based on constant deflection tests. For any form of stressing (axial, bending or crack propagation etc.) fatigue tests can be classified into Constant Amplitude and Variable Amplitude.

6.1.1 Constant Amplitude Tests

The simplest of all to perform, constant amplitude tests still requires careful consideration before tests are carried out. The following defines the terms used in describing fatigue tests.

σ_{max} = Maximum peak stress

σ_{min} = Minimum peak stress

R = stress Ratio = $\sigma_{min} / \sigma_{max}$

σ_{alt} = Alternating stress = $(\sigma_{max} - \sigma_{min})/2$

σ_{mean} = Mean stress = $(\sigma_{max} + \sigma_{min})/2$

In the past, fatigue tests were conducted at a constant mean stress with a varying alternating stress. The trend however is to use constant R ratio with varying peak stress as it defines the nature of the fatigue test better. In axial fatigue, the R ratio can vary between zero and one for tension-tension tests or infinity to one for compression-

compression. Where stress reversals occur, the R ratio is always negative. Where the tension stress is always greater than the compression stress, the R ratio varies from minus one to zero. Where the compression stress is greater, it varies from minus one to minus infinity. For bending fatigue, the R ratio can only vary between zero and 1 for bending in one direction only and zero and -1 for when the stress reversals occur.

An important consideration in fatigue testing is the frequency at which the tests is carried out. This consideration is particularly relevant for polymeric materials including wood due to adiabatic heating. Imayama and Matsumoto (1970) has already shown that for wood significant adiabatic heating occurs at a frequency of 40 Hz. Sims and Gladman (1978) argues that keeping a constant frequency for different load levels is not sufficient. Figure (6.1) illustrates the difference between keeping a constant frequency and a constant rate of stress application (RSA). If samples are tested at different stress levels and if the RSA is to be kept constant, the frequency has to be changed. This mode of testing has the effect of isolating the effect of rate on the fatigue life of the specimen. It has also a practical advantage in speeding up tests at lower stresses which would have a corresponding longer life. The equation for the calculation

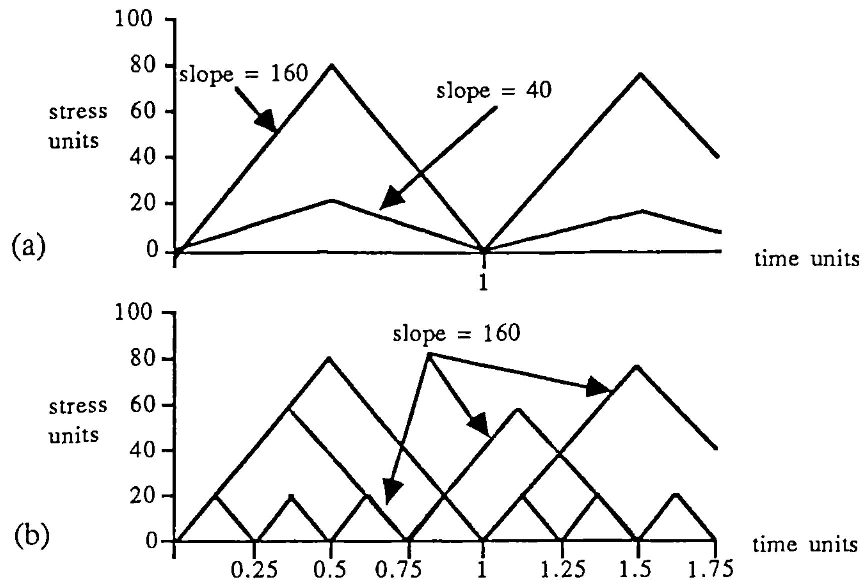


Figure (6.1) Comparison of fatigue tests at (a) constant frequency and (b) constant rates of stress application of the frequency for constant RSA may be expressed as

Frequency = (rate of stress application)/(4 * Alternating stress).....6.1

It should be noted that this expression is derived for fatigue tests using a triangular waveform. In most fatigue tests however, a sinusoidal waveform is used where it becomes the rate of change of the RSA that is constant. However as an approximation, the equation may still be used.

6.1.2 Variable Amplitude Tests

As the name suggests, with variable amplitude tests, the amplitude of the load (or deflection) is not constant but may vary in some predefined way. The basic reason for performing such tests is simply that engineering components in real life do not experience a constant amplitude fatigue regime. Also it is not straightforward to extend fatigue life from constant amplitude tests to real life variable amplitude. The only way to do so is to use Life Prediction Models, the most popular being the Miner's Rule of cumulative damage. This rule however is empirical and was proposed in the context of metal fatigue and has been shown to be unreliable for other materials such as composites (Shutz & Gerharz, 1971). Also for metals this rule is by no means accurate. Therefore it is still necessary to simulate service conditions in tests especially for critical components. However, as Figure (6.2) shows there are many possibilities in performing these tests.

The most expensive and difficult to conduct would be random tests with the number of stress levels not predetermined. The other approach is to reduce the complex waveform into some groups of levels which can then be intimately mixed in a Random Test or in a Block Program Test. By doing so, standard test programs have been established such as FALSTAFF etc. The creation of such standard programs in the context of Wind Energy Generators is discussed in Chapter 5.

Besides simply simulating service conditions, the design of a variable amplitude fatigue program can yield much information. The most obvious of these is in the

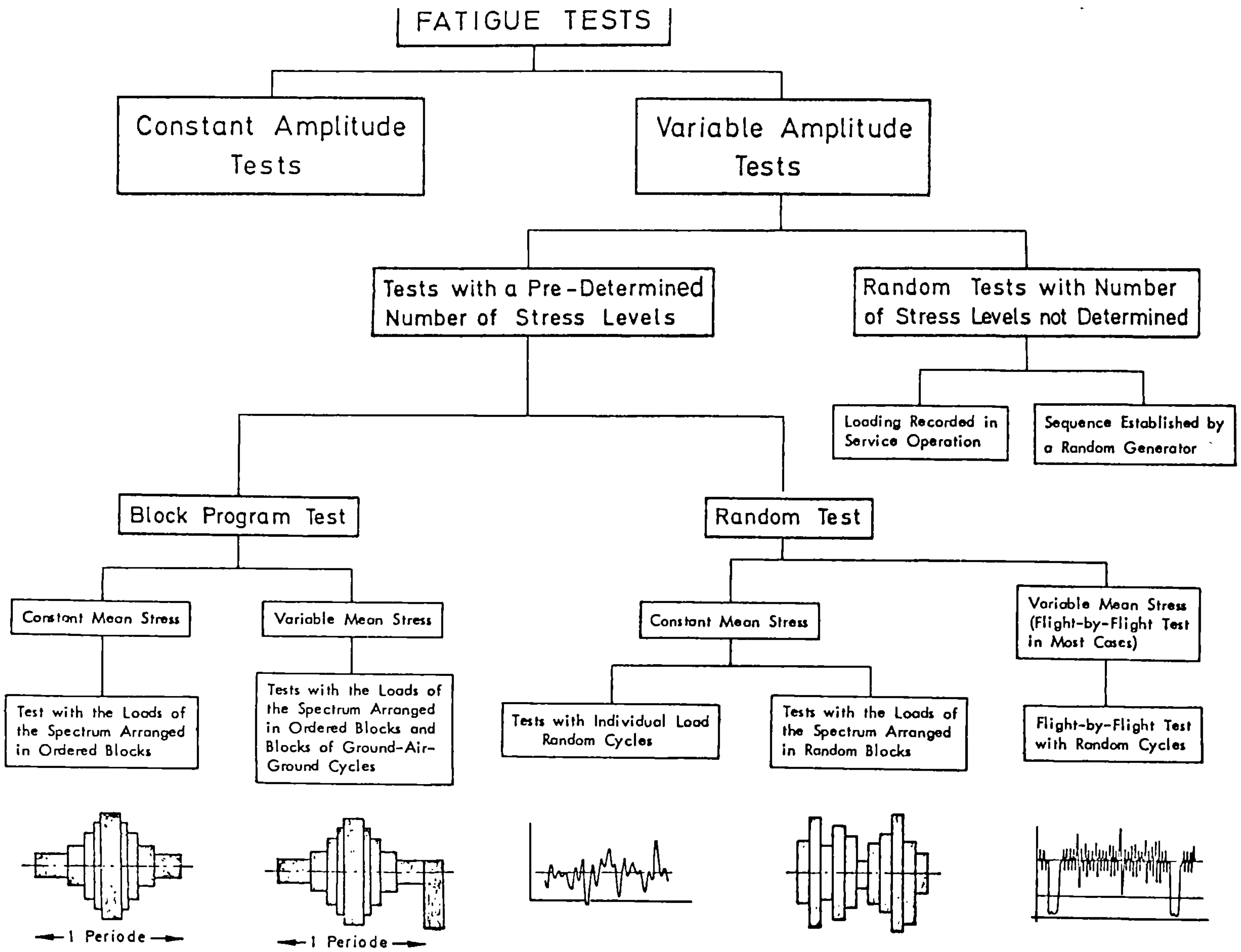


Figure (6.2) Most usual Variable Amplitude Tests. *(Schutz, D. 1981)*.

assessment of theoretical life prediction laws or the derivation of empirical laws. These laws may be specific to particular types of load conditions or may be more general like Miner's Rule. However, deviations from these laws often occur and the errors involved must be assessed. These errors can arise for many reasons but the most important are sequence effects. Sequence effects arise from inherent material properties. For example, in some metals, high load cycles followed by low load cycles have been found to extend fatigue life compared to prediction and the reverse reduces fatigue life. This effect is thought to arise from work hardening or the creation of residual stresses at high loads which extends the fatigue performance of the material at low loads (Osgood, 1982). With wood, summation of cycles of stress reversals with compression-compression and tension-tension loads are not likely to be straight forward and requires investigation.

6.1.3 Fatigue Test Machines

Over two decades, fatigue machines have greatly improved in capability, reliability and cost effectiveness. The developments are in two main categories - servohydraulic machines and the use of computers. Nowack (1981) has reviewed the types of fatigue machines available and this will be briefly summarized here.

Fatigue machines can be broadly categorized into two types, resonant and non-resonant machines. Within these two categories there are mechanical, electro-magnetic, hydraulic and servohydraulic machines. Figures (6.3) and (6.4) schematically illustrates the working principle of some non-resonant and resonant machines respectively.

Those types of machines where the loads are generated in a direct drive mode are of the conventional non-resonant category. Screw drives, hydraulic drives, cam drives or crank drives with adjustable eccentric or electro-magnetic drives are the oldest types. Although the simplest, these machines are not ideally suited for fatigue tests. These machines suffer the problem of slow test speeds (<1 Hz) or slightly higher test speeds if the loads are very small (<5 kN). However accepting low speeds, mechanical

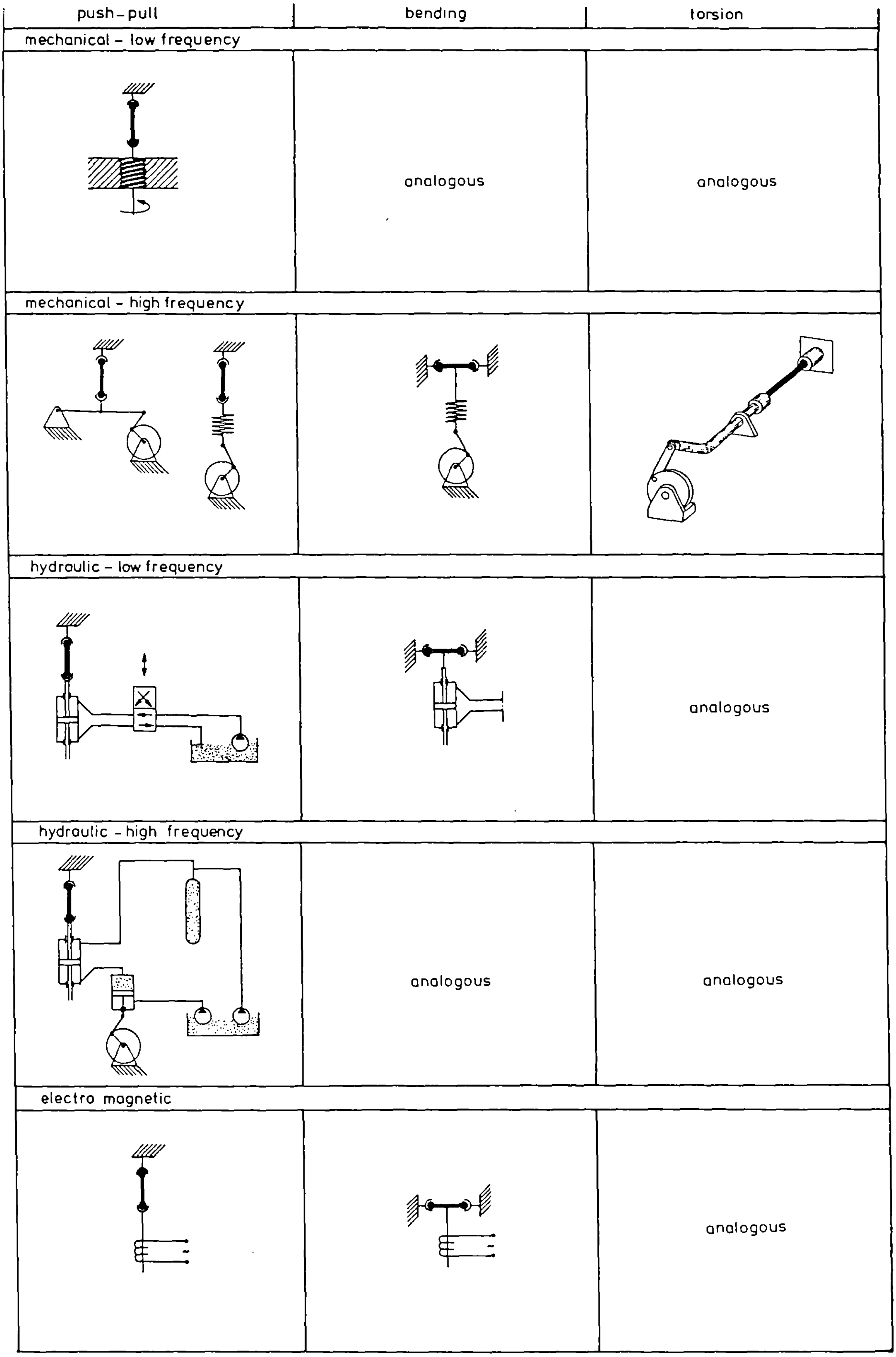


Figure (6.3) Conventional non-resonant direct drive machines. *(Nowack, 1981)*

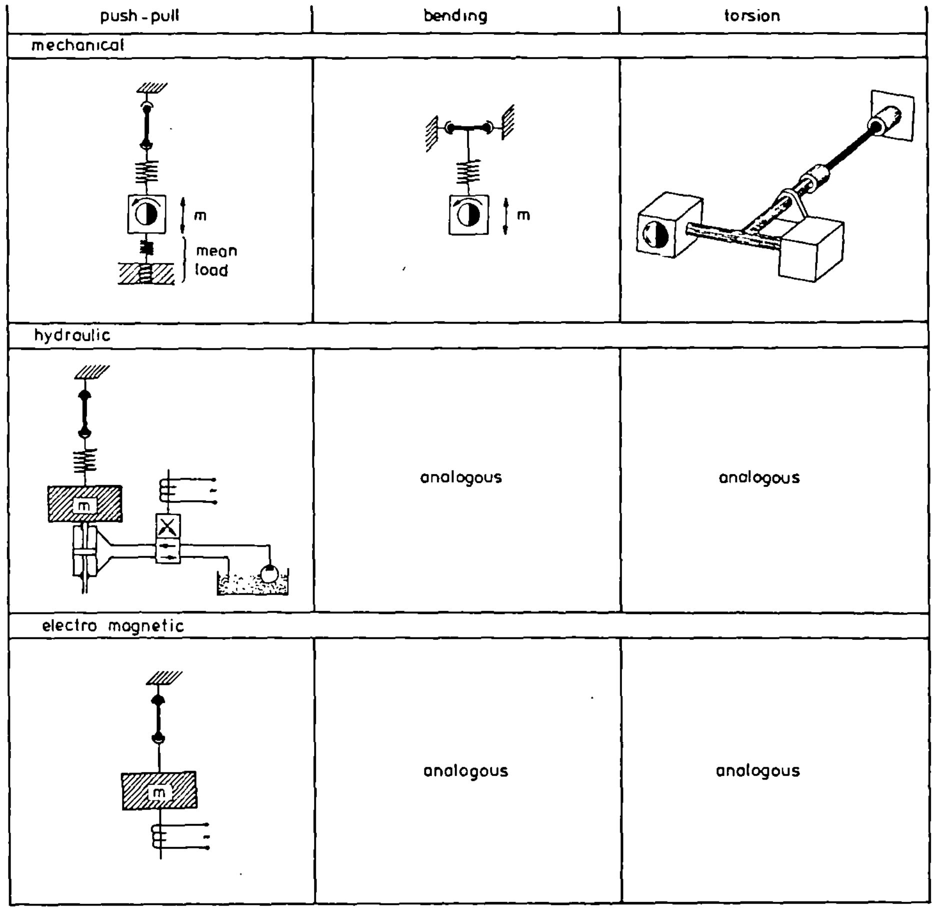


Figure (6.4) Conventional resonance fatigue test machine. *(Nowack, 1981)*

and hydraulic machines can achieve extremely high loads. They do have the advantage arising from their slow test speeds, that they are fundamentally easy to control and can establish reliable load or deflection control conditions.

Conventional resonant drive fatigue machines utilize the principle that a big spring mass system can be excited to perform sinusoidal oscillations with comparatively small amounts of energy, if the energy is supplied to the system at a rhythm and close to the resonant frequency of the whole system. It is important therefore that the energy dissipated by the specimen (eg. in thefrom of hysteretic heating) and the general damping of the system is small. This allows the undisturbed oscillations of the system to be rapidly built up. It is a characteristic of resonant machines that the load is gradually built up hence it is more applicable to constant amplitude tests where specimens can tolerate frequencies in excess of 10 Hz. They are therefore extremely useful in long term fatigue tests and have low running costs.

The servohydraulic fatigue machines are available in non-resonant and resonant forms, but they are not limited to constant amplitude tests. Since the following section is concerned with the development of a computer system to control a non-resonant servohydraulic system, much more detail is given of its design principles and characteristics. The resonant system is of less interest and is basically an extension of the non-resonant system, again utilizing the principle of a large spring mass.

Servohydraulic machines are commonly used today in laboratories as general purpose fatigue machines. They are extremely versatile and capable of applying a wide range of loads and frequencies. The basic working principle is shown in figure (6.5) and consists of four basic elements- the hydraulic pump, the servovalve, the actuator and the servo or feedback loop controller. One or more hydraulic pumps are used to keep the hydraulic fluid at a constant pressure of around 3000 psi. with accumulators to level out any instantaneous variations. The servovalve is the most critical part of the system as it controls the hydraulic flow into the actuator. The whole system is then controlled by means of a feedback loop controller.

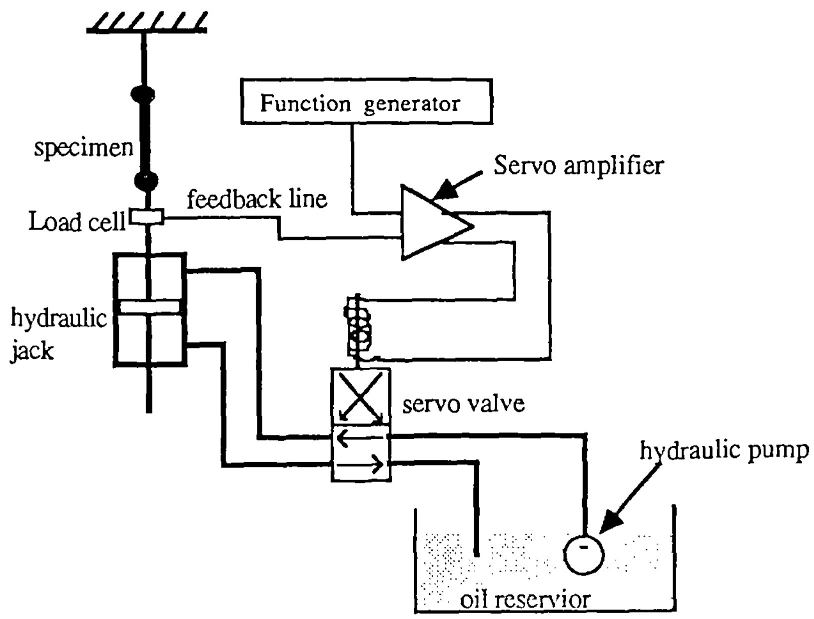


Figure (6.5) Basic components of a servohydraulic fatigue machine.

The controller functions to ensure that the actuator performs as required. An electrical command signal is fed into the servovalve so that its position determines that a certain amount of hydraulic fluid flows into the appropriate chamber of the actuator which causes a force at the test specimen and a feedback (control) signal. This feedback signal is commonly from either a load cell or LVDT although any other source relevant to the type of test may be used. The command signal is balanced with the feedback signal by means of an integrator circuit so that what is required by the command signal is achieved. In practice careful optimization of the integrator circuit is necessary as two conflicting requirements exist. The first, the deviation between the command and feedback must be small, and second, the closed loop must be stable. Instability occurs when the resonance frequency of the system is reached or the phase angle of the servovalve reaches 90°. In tests, conditions of this nature rarely occur however in the context of computer control this is important as should the command signal of the computer be incorrectly defined instability can occur (and has been experienced). Because a phase difference always occurs with the command signal leading the feedback, attempts to use a computer to track and control the fatigue machine as a secondary feedback loop can result in severe difficulties. This will be discussed further in the design of the computer control system.

Two factors govern the design and hence the performance of a servohydraulic system. First, the size of the actuator determines the maximum load capability of the system. A larger actuator diameter provides a higher maximum load capability. Second, the flow rate of hydraulic fluid as supplied by the hydraulic pump combined with the number of servovalves limits the stroke amplitude versus the frequency capability of the system. Figure (6.6) shows the performance limits of a typical system. At higher frequencies, the maximum stroke amplitude that can be achieved is smaller. Also, this limitation is dependent on the peak loads required and is due to the compressibility of the fluid. To obtain high frequencies and stroke amplitudes, large or multiple hydraulic pumps are required. This limitation often exhibits itself during a fatigue test when the specimen looses its stiffness resulting in a fall in the peak load to compensate for the

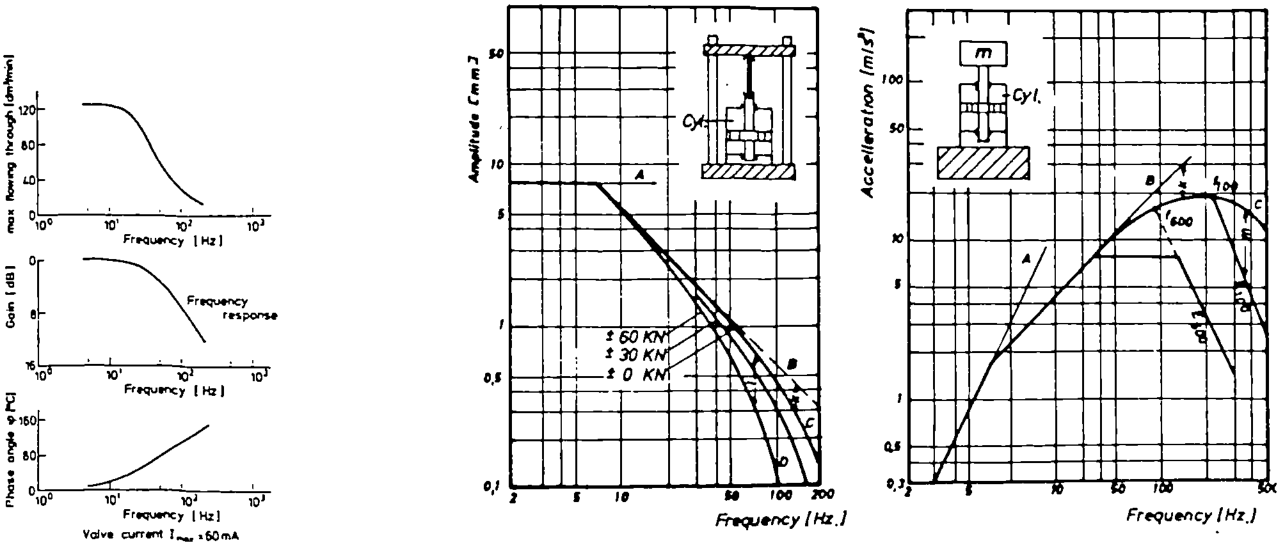


Figure (6.6) Example performance diagrams of a servovalve and of a servohydraulic fatigue machine. *(Nowack, 1981)*

increased stroke required. This effect may be avoided by specifying a bigger pump and increasing the number of servovalves. Alternatively, adjustments are necessary in the command signal. For some materials this is not significant. However, polymers, some composites and wood do show significant reductions in stiffness.

Another consequence of the limitation is experienced in variable amplitude fatigue tests. Where the loads change sharply from high amplitudes to low and vice versa. Overshoots and undershoots occur resulting in incorrect loading. Lanciotti (1983) proposed a scheme to reduce this problem by performing dummy runs of the load program measuring each peak and correcting the load program from that measured. By repeating this the error involved was significantly reduced. With wood, a realtime correction system would be of great benefit.

6.2 Why a New System?

6.2.1 Introduction

The development of a new computer control system has to be examined in the light of the effort and cost required. To use the latest in computer technology may be a good reason but one that can hardly justify the many hours required in developing the hardware and software. This section therefore reviews the current systems available on the market and those developed by various laboratories. The decision to develop one's own system can therefore be examined in the context of what is required and what is available. However given that the decision to develop is taken, the nature of developmental work is such that hindsight often exposes errors in judgement. Underestimation of the time scale of a task is always the Achilles heel in development. The experience of the author is no different.

Computer control systems may be considered as large scale systems and small scale systems. Large systems are those using large mini computers for controlling one or even multiple actuators. These are common in large fatigue laboratories where large structures or an array of servohydraulic machines are used (Nowack et. all, 1979). Commonly, PDP 11s or Minc 11/73s are used costing upwards of £20,000 for the computer hardware alone. These systems are beyond the needs of the test program in this research. Also it is interesting to note that the capability of these computers only match modern microcomputers in speed with less memory available and cost ten times as much. Therefore, they will not be discussed.

6.2.2 Basic Requirements of a Computer Control System

The basic functions required of the control system in its original conception may be stated as follows:

(1) Cycle by cycle monitoring of peak loads and peak strains or deflection of the specimen up to a test frequency in excess of 20 Hz. Data monitored should be stored on magnetic discs for post test analysis.

(2) The system should be able to accurately control the loading of the sample by the servohydraulic machine ensuring that load fluctuations associated with the changes in specimen compliance are corrected for.

(3) Complex loading. A minimum capability of programmed block loading with possible extensions to more complex random loads.

(4) The system should be low cost, easily duplicated and uncomplicated to use requiring no modifications to the servohydraulic machine.

6.2.3 Small Single Actuator Computer Control Systems

There are a variety of systems described by their designers in the literature. Various commercial systems have also come on the market, some since the development work of the author began. It is noted however that these systems are generally designed for static testing and are therefore either unsuitable for the needs of this project or need substantial modification. It is instructive however to consider their designs and limitations.

Most systems described in the literature have been designed in the context of fracture mechanics type testing. A total of seven different systems were reviewed all of them using different computers although Digital Equipment's range of computers were most popular with the LSI/11 (Styles & Baker, 1978), PDP 11/23 (Jablonski & Lee, 1983), and Minc 11/23 (Griffiths, 1983) being used, the last being the most recent machine. Other computers include the Apple II (Fleck & Hooley, 1983), Texas Instrument's TM 990 (Barker & Smith, 1985) and the Commodore 8296 (Smith and Abbot). Each system has its advantages although those based on Digital Equipment's computers tend to be more expensive (>£10,000). Commercial systems available from Dartec or Instron are often based on Digital Equipment computers and therefore costs are prohibitive. These systems however have the advantage of being well designed but are biased towards static testing and therefore require some custom software (which would have to be purchased).

The concept of using computers to monitor fatigue crack propagation is similar in all the published systems. The implementations however do vary greatly. Much depends on the type of signal being monitored. The Krak-gauge which is an adhesively bonded foil gauge makes less demands on the computer than the potential drop method which correspondingly demands less than the Crack Opening Displacement (COD) method. This is because the COD method requires the following of an oscillating displacement signal while the signal of the other two methods varies slowly. With the compliance measurements necessary in the COD method, this is not very dissimilar to the system required. A few limitations however exists in all these systems. Firstly, sampling rates tend to be slow which means that the test frequency is slow. Fleck and Hooley overcame this by interrupting the test cycles during data logging. With a more powerful computer, Jablonski and Lee used "burst mode" which meant data was continuously sampled and stored for a short time followed by the analysis. Either way, the compliance is not continuously monitored and cycles are lost which would be significant for shorter term (<10,000 cycles) fatigue tests. It is evident therefore that a much more advanced computer system is necessary.

Another consideration is the ability of the computer control system to accurately impose the load level required of the fatigue machine during test. As discussed earlier, servo-hydraulic fatigue machines do not maintain peak loads when the compliance of the specimen changes. In complex loading the problem is more serious due to the phase lag of the feedback signal to the command signal. The computer controller should automatically adjust for such changes. This can only be achieved if the monitoring of the load signal is performed in real time. Low cost systems are not able to perform this automatic correction and only with the more powerful PDP11/23 and Minc 11/23 systems is this possible.

It is interesting to note that for the system using the Commodore 8296 (Smith and Abbott), the IEEE488 interface was used hence the computer was in a secondary role, not directly controlling the fatigue machine. Digital volt meters with peak detectors and IEEE function generators were used controlled via the IEEE488 interface. Using this scheme the computer serves in a supervisory role and hence load correction becomes possible. The simplicity and development time for such a system is relatively low but it suffer from drawbacks. It is basically costly since each individual component such as the function generator, digital volt meter, peak detectors, etc., costs as much if not more than the computer itself. Digital volt meters are slow in sample rates and peak detection is not accurate. System extensions to perform complex load tests are not possible without giving the computer direct control using digital to analogue converters hence reverting to a system more similar to the others.

6.2.4 Problems and Principles in Computer Control of Servohydraulic Machines

The principles associated with the computer control of a servohydraulic fatigue machine and solutions to some of the problems involved were described briefly above. They may be considered in four categories.

(1) **Sampling Rate.**

The hardware and software must be able to handle the task of monitoring a signal of at least 100 Hz and generating a signal of up to 50Hz continuously. While a very smooth sinusoidal waveform is desirable in signal generation, in practice this is not strictly necessary since the hydraulics will not be able to respond to smooth steps. However, should the steps be too large, the ability of the hydraulics to follow decreases. In the system developed, 180 steps are used. The sampling rate required for monitoring the feedback signal depends on the sophistication of the peak detection routine. For a simple routine as used in this system, the sampling rate for 12 bits accuracy should be at least 90 to 180 samples per cycle depending on the signal amplitude. However the feedback signal from the fatigue machine may not be purely sinusoidal. Slackness in joints and clamps contribute to give some higher frequency components to the feedback signal. Therefore in practice it is desirable to exceed the minimum sampling rate by a factor of 3. Otherwise the accuracy of the system might be much less.

(2) **Phase lag between command and feedback signals.**

Phase lag creates problems, especially for variable amplitude tests, in

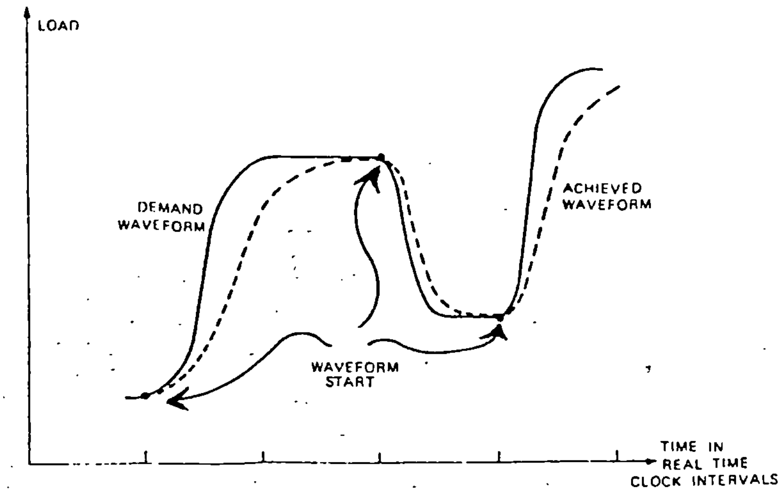


Figure (6.7) Schematic of possible shape for a demand waveform generated incorporating demand holding. *(Craig, 1978)*

ensuring the accuracy of the peak loads on the sample. The phase lag has the command signal leading the feedback and therefore making any direct comparison of the two signals to make fine corrections is not possible. One solution is to generate command signals of half sine waves from peak to peak with dwell times at the peaks. As illustrated in figure (6.7), when the feedback signal has reached the peak, comparisons and adjustments to the peak can then be made. Lanciotti (1983) suggested an alternative solution which will avoid the slower test speeds due to the dwell periods. The whole variable amplitude test block is initially used and the corresponding actual peaks achieved in the test are recorded. The errors associated with each peak are then calculated and the entire block can be corrected as follows:

new input peak = old input peak + (required peak - old output peak).....6.2

This procedure was applied to a dummy specimen prior to the actual test. It can conceivably be extended, repeating the procedure and applied in real time on a test specimen provided there is sufficient computing power.

(3) **Stiffness Changes.**

When the fatigue machine has a pump with a low flow rate, stiffness changes in the specimen can result in a large drop in peak loads as the test progresses. This problem is overcome when the routine for load peak correction as described above is applied. However, the problem is more gradual here hence if Lanciotti's method is used, it has to be applied continuously throughout the test. Also correction must be carried out for constant amplitude tests although this is necessary only occasionally, eg. every 100 cycles.

(4) **Signal Noise.**

Electrical equipment inevitably suffers from signal noise. This can arise from many sources. It can be due to its own circuitry although good design would ensure that this source is minimized. Generally, the level of signal noise is sufficiently low however if this not the case, for low frequency applications such as fatigue testing, high frequency filters can be added easily. External sources inducing signal noise into the equipment are however more difficult to overcome.

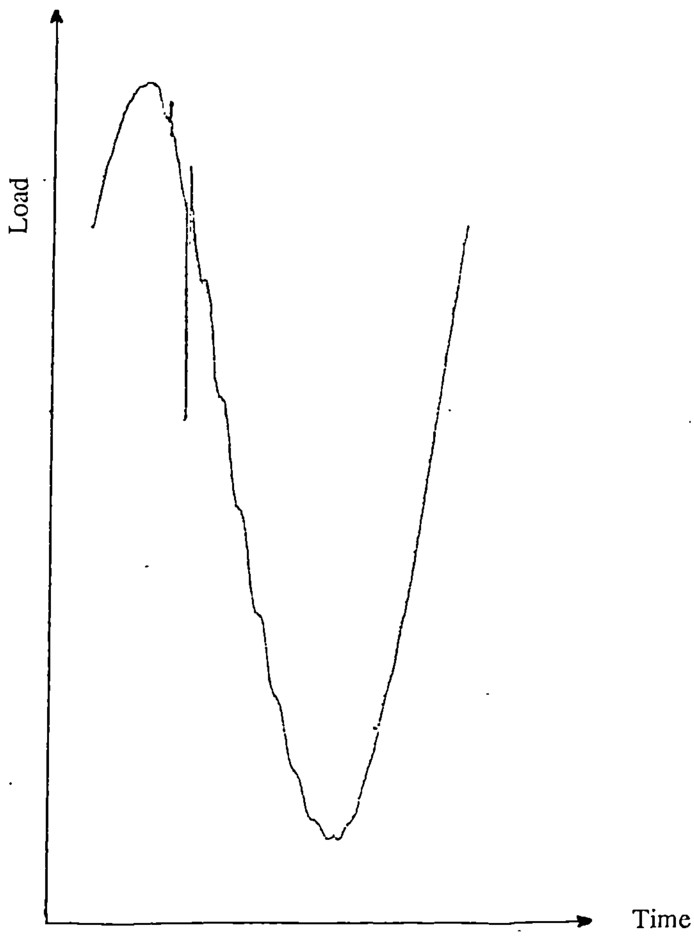


Figure (6.8) Example of a transient and its effect on the output signal waveform.

Radio frequency and mains supply transients are the most common sources. These often arise from adjacent heavy current equipment such as electric motors and relay switches. A typical occurrence monitored during a fatigue test is illustrated in figure (6.8). Such occurrences would interfere with load correction procedures as they are random and can lead to over corrections. No workable scheme has yet been found which eliminates this problem and it can only at best be contained.

6.3 Design of SArGen - Signal Analyser and Generator

6.3.1 System Architecture

The concept of the new computer control system is illustrated in figure (6.9). It centres around a custom designed 6800 microprocessor based computer (called SArGen) with a digital to analogue converter (DAC) and analogue to digital converter (ADC). An Apple Macintosh computer is used as the user-system interface which supervises the SArGen and stores relevant data on 3.5" magnetic disks for later analysis. The idea of this design is to separate the task of basic peak to peak detection and function generation from online data analysis, decision making and data storage. In other words, SArGen acts as an intelligent interface between the Macintosh computer and the servohydraulic machine. The use of the 6800 based computer with 8K of firmware (software in Read Only Memory or ROM) and 48K of Random Access Memory (RAM) provides the necessary 'intelligence'. A high speed serial link is used for communication with Macintosh. It may be argued that all the tasks could be performed by just one powerful computer like the Macintosh or IBM PC, however the software would largely have to be written in Assembly language and timing difficulties would occur especially at higher test frequencies. By separating the tasks between Macintosh and SArGen, the complications are reduced and only the firmware for SArGen needs be written in Assembly Language. Software for the Macintosh is largely written in Pascal with only the software for the serial interface in Assembly language.

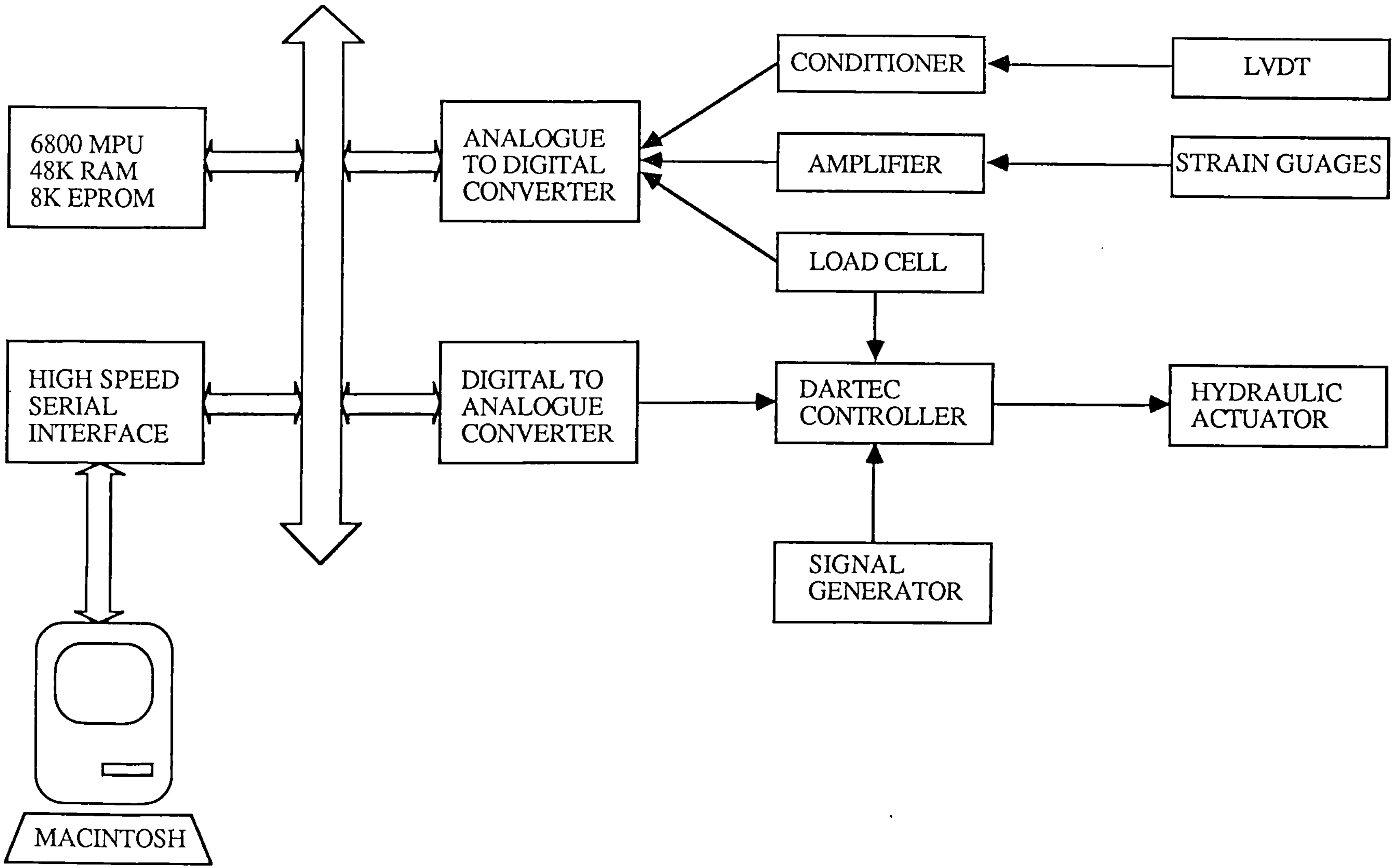


Figure (6.9) SArGen system architecture for monitoring and controlling fatigue machine.

6.3.2 SArGen Hardware Design

SArGen is designed in modular form with plug-in cards for each of its functions. There are 5 cards apart from the power supplies - microprocessor, memory, serial interface, DAC and ADC cards. The assembly is capable of taking additional cards: a LVDT transducer card and a 3 channel autozeroing strain gauge amplifier card are included.

The 6800 microprocessor unit or MPU forms the heart of SArGen. It is an 8 bit processor running at 2 MHz and capable of addressing 64K of memory. All the address, data and control lines have been made available on the backplane (see figure (6.9)) so that the full capability of the MPU can be exploited. A particularly useful feature of this processor is its interrupt capability which enables external timing signals to be used to cause the 6800 MPU to perform specific tasks at regular intervals. This facility is used in signal generation. The choice of the 6800 MPU is mainly dictated by familiarity with it. It is not the most powerful MPU available but based on conservative calculations, was deemed adequate for the task.

The memory of SArGen is decoded to give 48K of RAM located from $0000 to $BFFF and 8K of ROM located at E000 to FFFF. The remaining 8K is made available as 8 chip select lines for input/output peripherals. The address of the select lines are C000, C400, C800, CC00, D000, D400, D800, DC00. The 8K of ROM is organized to use two 4K EPROM chips which can then be independently programmed when necessary. Four of the select lines are used in the basic system for the serial interface, ADC, DAC and programmable timer (for interrupts). The other four may be used for other devices as necessary. With the program stored into ROM, no down loading is necessary and the system is ready from switch on. The 48K of RAM is used as a temporary store of variables and as a buffer for data either to be used in digital to analogue conversion or vice versa before transmitting to the Macintosh.

The analogue to digital converter card uses a high speed 12 bit converter. It is capable of conversion rates of up to 20 KHz giving accuracy of 1 in 4096 or 0.025%.

Four input channels are available so that four different analogue sources can be monitored. To avoid time delay differences when the four channels are monitored (it takes over 50 μs between conversions) the card is designed so that all four channels are sampled and the signal held until all four channels (or the required number) are converted. This means that readings from the four channels are effectively performed at the same time. This is important as in a fatigue test, all incoming signals are varying rapidly and significant errors would result if readings were not taken simultaneously. For example, with a 10 Hz sine wave, errors of greater than 0.5% can easily occur.

The digital to analogue converter card also uses a 12 bit converter. This is of a standard design. The card also has a programmable timer. This is a device which can be used as a counter or a timer which causes an interrupt signal to be generated to the MPU at regular intervals set by the software program. In this case, the timer function is used to vary the frequency of the sine wave generated to the fatigue machine. Interrupts can occur at rates of 1 MHz down to 16 Hz. Since 180 discreet points (therefore interrupts necessary) are used to define a sine wave, this corresponds to the generated signal frequency of 5,555 Hz to 0.088 Hz (in practice the highest frequency is only about 500 Hz due to software delays).

Macintosh has the capability of communicating with other computers at speeds of up to 0.9 MBits/sec using a RS232/422 interface. The serial interface card of SArGen uses this interface and is designed to communicate with Macintosh at a speed of 0.5 MBits/sec. At this speed data transfer can be achieved very rapidly with very little delay.

6.3.3 SArGen Software Design

There are three components to the software. SArGen has a 4K ROM programmed in 6800 Assembler language. The main control program on the Macintosh is programmed in compiled Pascal with the serial communication software part coded in 68000 assembler for speed.

With the Macintosh supplying commands to SArGen, the range of commands that must be programmed into SArGen must be complete for all possible control requirements. The commands are coded as single byte characters with data bytes following if required. Table (6.1) lists all the commands and their respective command code used. Most of these commands are associated with the setting up of test parameters and manual control of the servohydraulic machine.

Table (6.1) Command codes used in the software for communication between SArGen and Macintosh.

Code	Hex. Code	Command Description
1	01	Start ADC and DAC processing
96	60	Terminate processing
3	03	Transmitting 128 bytes of data
4	04	Transmitting 256 bytes of data
5	05	Transmitting 512 bytes of data
6	06	Transmitting 1024 bytes of data
128 to 255	80 to FF	Transmitting data: 0 to 127 bytes
8	08	Manual DAC control
80	50	Terminate manual DAC control
10	0A	Option select: continuous ADC. Must be followed by two bytes for time delay.
11	0B	Option select: ADC with peak detection.
13	0D	Option select: DAC sine wave generation. Must be followed by 6 data bytes.
14	0E	Option select: ADC off
15	0F	Option select: DAC off
16	10	Transmitting program code. Followed by two bytes of program size and program code.
17	11	Send XON Character
18	12	Send XOFF Character
20	14	JSR to RAM program
21	15	Set Strain Gauge amplifier offset
33 to 42	21 to 2A	JSR to ROM2: locations E000 to E012
113 to 127	71 to 7F	Channel selection

The structure of the program in ROM is illustrated by the flowcharts in Appendix A. The complete listing of the program is also given in Appendix C. On turning on SArGen, the program immediately goes through an initialization routine to ensure the DAC is set at zero volts, the timer is turned off and all the memory in RAM is working. Also necessary is to set up the serial interface chip for the correct communications protocol ie. 8 bits data character with even parity and 1 stop bit. Once initialization is done, SArGen is ready to perform any of the tasks required. It therefore goes through a routine waiting for a command to be received from Macintosh. When a command is received, it immediately checks if it is a command to perform the main test. If not, it goes into a subroutine to check if the various other command options are required. It then returns and repeats this until the command to start test is invoked.

Among the command options are two to control the digital to analogue converter. Command code 16 turns off this facility but code 15 commands a sine wave to be generated. Six data bytes must be provided with this command to specify the maximum peak, minimum peak and conversion rate. When a test is to begin, two tables are calculated from the given maximum and minimum peaks to describe the two half waves of a sinusoidal waveform. This is done by scaling a quarter wave table stored in ROM to give a half wave starting from the the minimum peak to the maximum peak (91 values) and another half wave from the value after the maximum to the value before the minimum peak (89 values). A total of 180 conversions are therefore used to describe one complete cycle. This also means that a conversion rate 180 times faster then the required cycle frequency is required (eg. 1800 conversions per second for a 10 Hz wave frequency). The timer has to be set to give this conversion rate. Once set, the timer will interrupt the 6800 to perform D to A conversion at that rate. The routine that initiates the conversion itself is a selfcontained program separate from the main program. It is invoked by the jump vector which the MPU, when interrupted by the timer, automatically goes to. On completion of the interrupt routine, the MPU returns to the location of the main program where it was at the point of interrupt. The interrupt routine is therefore invisible to the main routine except for the small delays associated with servicing the interrupt.

Two test options are available: fatigue testing and static testing. Static testing simply enable continuous sampling and A to D conversions to be performed at a preset rate (up to 1000 samples and conversions a second). This is therefore a simple routine with a loop to perform these conversions. A software delay is used to adjust for different sample rates. The routine for fatigue testing performs A to D conversions at the full speed of the 6800 MPU. It includes routines to detect the peaks of the input cyclic signals. The maximum peak is detected by comparing successive conversions until the latest value is less than the maximum sampled value by a given threshold amount. Once detected, the maximum is stored and the software begins seeking for the minimum peak. This is found when the the value is greater then the minimum sampled by the same threshold amount. The value of the threshold is not preset but variable and can be adjusted by software. It is important that the threshold is not too small as then spurious noise signals would be deemed as peaks or not too large resulting in no peaks detected. A value of 128 (representing 0.6 Volts) has been found to be ideal.

When a specimen fails, SArGen must be able to terminate its fatigue testing routine itself since such an event could occur when the machine is unattended. Fatigue machines generally have trips to do this, shutting off the pump or isolating the machine from the signal generator and switching to a safe mode. To reduce complications in the development, this facility of the fatigue machine is used hence SArGen does not need to directly stop the test but only recognise it. This it does by recognising the fact that when a specimen fails, the cyclic input signals stop hence no peaks can be detected. If two cycles have been generated by the DAC, without any peaks detected, this invariably means sample failure. The program therefore performs the required termination routine.

6.3.4 Macintosh Software Design

The Display Screen for User Operation

The Macintosh side of the software has been developed making full use of the windows, icons, menus and mouse facilities of the machine. There are two separate programs for fatigue testing and static testing (see Appendix D for listing of fatigue testing program). The program for static testing is a subset of that for fatigue testing being simpler in concept. The image of the screen during fatigue testing is illustrated in figure (6.10). The menu line at the top contains the following options.

(a) : These are Desk Accessories and are installed by the user for the machine.

(b) **General** : A general control menu.

  Clear replies : Clears the data displayed on screen.

  Clear Data : Clears the memory of the data from preceding test.

  Save Data : Saves data from the completed test in an alternative file.

The following selects the active channels for data input.

  Channel 1

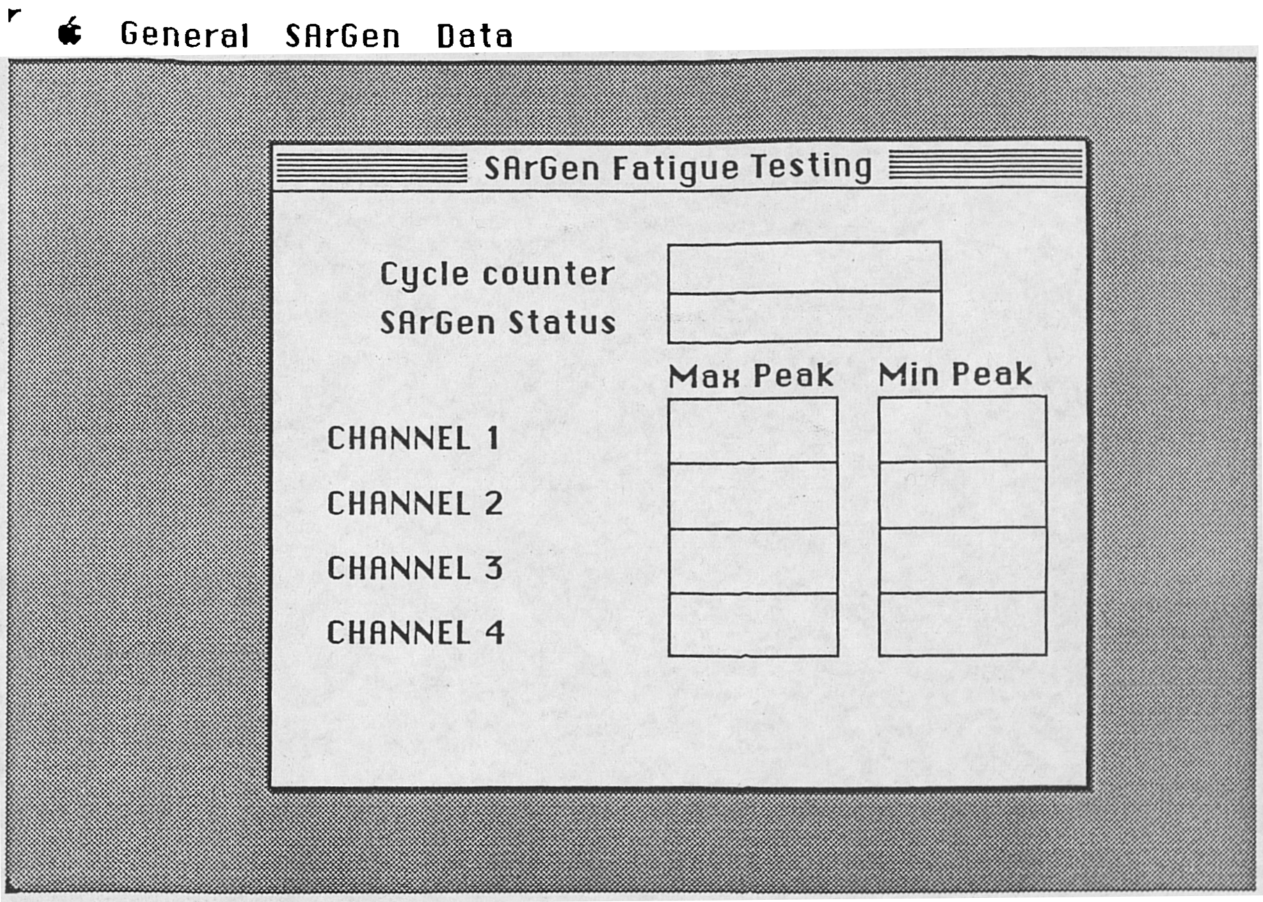


Figure (6.10) The display screen on the Macintosh computer on startup.

Channel 2

Channel 3

Channel 4

(c) **SArGen** : A menu for direct communications with SArGen.

XON : Send XON Character.

XOFF : Send XOFF Character.

Calling SArGen : To test communications with SArGen; SArGen replies with an XON which the screen will indicate with "OK".

DA Off : Turns off digital to analogue signal generation.

DA Constant Amp Wave : Selects sine wave generation. The user will be prompted for maximum peak, minimum peak and signal frequency.

DA Variable Amp Wave : Selects a variable amplitude test. The user will be prompted for a text file which contains the amplitude data.

Zero SGA : This facility is used to zero the 3 channel Strain Gauge Amplifier card. The output from channel 1 to 3 of the SGA must be connected to channels 2 to 4 of the ADC card.

Jack Control : A scroll bar is used to manually control the hydraulic jack. The control mode (position or load) is dependent on the mode the fatigue machine is in.

Begin Test : To initiate a fatigue test. A file name and the level for recording of a change in peak level will be requested if needed for data storage.

Stop Test : To terminate a current test.

Send Integer : This facility is for special situations where the user wishes to send specific commands to SArGen.

(d) **Data** :

A moving average routine which averages 5 peaks (maximum and minimum) continuously can be selected for each channel. Where the peak changes by greater than 1.2% of the fullscale, the routine starts afresh its moving average. The following selects the channels to which the smoothing routine is applied.

Smooth Channel 1

Smooth Channel 2

Smooth Channel 3

Smooth Channel 4

The next two options offer to save peak data or to control the fatigue machine. These options are included for complex loading as the data recorded in a test can be extremely large.

Sample and Save

Sample only

The default is for **Sample and Save.**

Program Structure

The programs for fatigue and static testing was developed using compiled PASCAL. The version for static testing is a subset of that for fatigue testing. The flow chart for the program is given in Appendix B. The Macintosh computer receives commands from the operator through the screen as what is termed *events*. Data from SArGen is received asynchronously via the RS422 serial port. An RS422 serial port driver was developed and a listing is found in Appendix E. When data is transmitted by SArGen, the Macintosh processor is interrupted to the driver to receive the data which is stored in a temporary buffer. The processor than returns to the main program which polls the buffer continuously to check and read the data. The main routine also checks for any user initiated events diverting and branching to the appropriate routines for processing. The cycle counter has also to be continuously maintained on screen and the load level checked at programmed intervals. Should variable amplitude testing have been selected, a check has also to be made if the next block level is to be initiated.

Chapter 7

Experimental Methods

7.1 Introduction

The main thrust of this research is to investigate the flexural fatigue properties of wood under the condition of constant peak load as compared to previous work published in the literature which is primarily in constant deflection. Constant deflection suffers from the fact that the load level drops with cycles due to the increase in compliance and the development of creep strains or perhaps, more appropriately, fatigue strains. The experimental program therefore centres around the use of a servo-hydraulic fatigue machine which can accurately apply a constant load to the specimens.

Early in the program, it was realised that more information on the changes occurring in the specimen while being fatigued was necessary to obtain a better picture of possible fatigue failure mechanisms in wood. It was therefore felt that two approaches were necessary: (a) to follow the fatigue strains and stiffness changes during fatigue, and, (b) to make a fresh study of wood microstructure, using electron and optical microscopes.

To facilitate the first approach, a new computer system was developed. This is described in detail in Chapter 6. The system enabled changes in applied peak loads and the resulting peak deflections to be accurately monitored cycle by cycle. This allowed the detection of the drift in deflection due to fatigue strains and allowed the calculation of the secant modulus of the material. With the new system, complex block loading fatigue was also possible and in view of the need to establish design procedures for the new generation of wooden WEC blades, an exploratory study of block fatigue loading was attempted.

Time prevented a detailed study of microstructural damage to be made using electron and optical microscopes. With assistance from Dr. J. M. Dinwoodie of the Building Research Establishment, a promising approach to the study of fatigue failure mechanisms was developed. 20μm thick microtomed sections of wood were examined with the optical microscope.

The needs of the Wind Energy Industry in this study of the fatigue failure of wood have been considered. As a result, two main types of wood were chosen for this work - Khaya ivorensis, a tropical African hardwood and Sitka spruce from the UK. Khaya has been used in the form of sliced 4mm thick veneers laminated with epoxy resin for WEC blade manufacture. This was therefore the principle material tested. Later on, rotary cut Khaya was used for the second generation blades. This new material was therefore incorporated into the testing for comparative purposes. Because Khaya is a hardwood it has a relatively complex structure. Sitka spruce, a softwood, was therefore selected for studies of possible ways in which fatigue damage develops. This also provided a means of comparison between the fatigue properties of hardwoods and softwoods.

The experimental program therefore fell into three categories to obtain and observe the following:

(a) The stress versus cycles to failure or fatigue life of wood.

(b) The development of fatigue damage as monitored by changes in fatigue strains and stiffness.

(c) Anatomical changes during fatigue as observed using electron and optical microscopy.

7.2 Sample Preparation

7.2.1 Khaya Ivorensis Laminates

Khaya ivorensis was supplied as 4mm thick veneers in two forms, sliced and rotary cut. The sliced veneers were used in the HWP-300 WEC blades and supplies were from the consignment used for the manufacture of the blades. Rotary cut Khaya was introduced later for the second generation blades and subsequently incorporated into the test program. The veneers were visually examined for defects following the quality control procedure used by the blade manufacturers. Three types of defects were found which resulted in the rejection of veneers. These were longitudinal splits and transverse cracks, shakes and brittle heart. These defects would reduce the strength of the laminate and brittle heart also greatly increases its density.

The veneers were laminated together using a room temperature cure epoxy resin purchased from Structural Polymers Ltd. This resin system consists of, by volume, 100 parts of SP110 epoxy resin and 20 parts of SP210 hardener. Because a vacuum bagging technique was used in blade manufacture, a similar method was adopted. This meant that the resin needed a filler to provide a more viscous resin characteristic during vacuum bagging. The filler used was 20 parts of SP glass microfibre, also from Structural Polymers, added to the resin mix.

The vacuum bagging manufacturing technique, widely used in the boat building industry and in the making of composite structures, is an extremely simple and effective method for laminating the veneers. The set up used is illustrated in figure (7.1). The resin mix with filler was first applied to the surfaces of the veneers using approximately 300 gm/m^2. The veneers, about 0.5m wide by 1m to 1.3m long, were then sandwiched together, 4-ply thick, and placed into a polythene bag with a wooden base. Netting was placed above the veneers and the bag sealed. A vacuum was then applied removing any air bubbles between the laminations. The atmospheric pressure around the bag also pressed the veneers together giving a laminate of very high integrity. Figure (7.2) shows an electron micrograph of a section of the laminate showing that the resin has penetrated some of the vessels in the veneers. The vacuum was applied for over 15

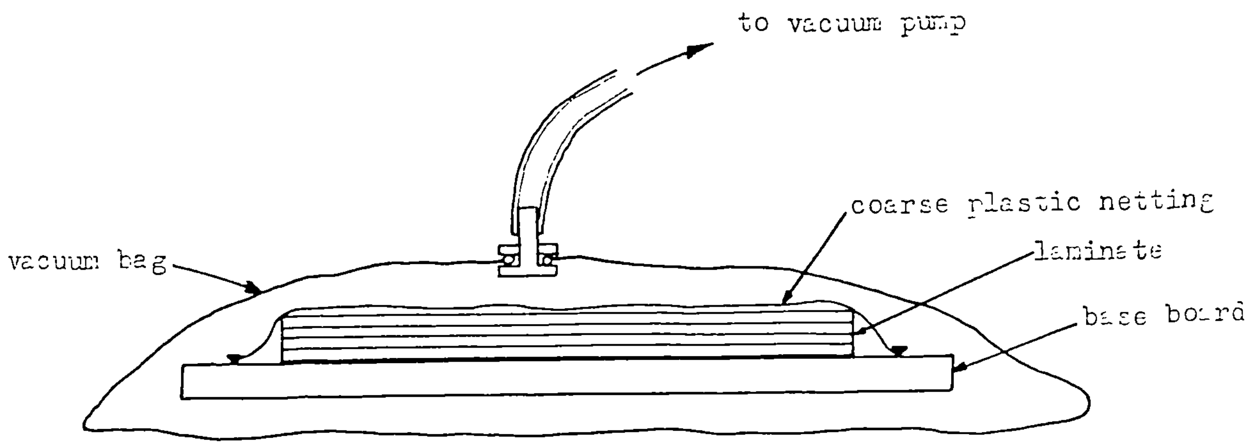


Figure (7.1) Illustration showing the simple technique of vacuum bagging for the manufacture of Khaya Laminates.

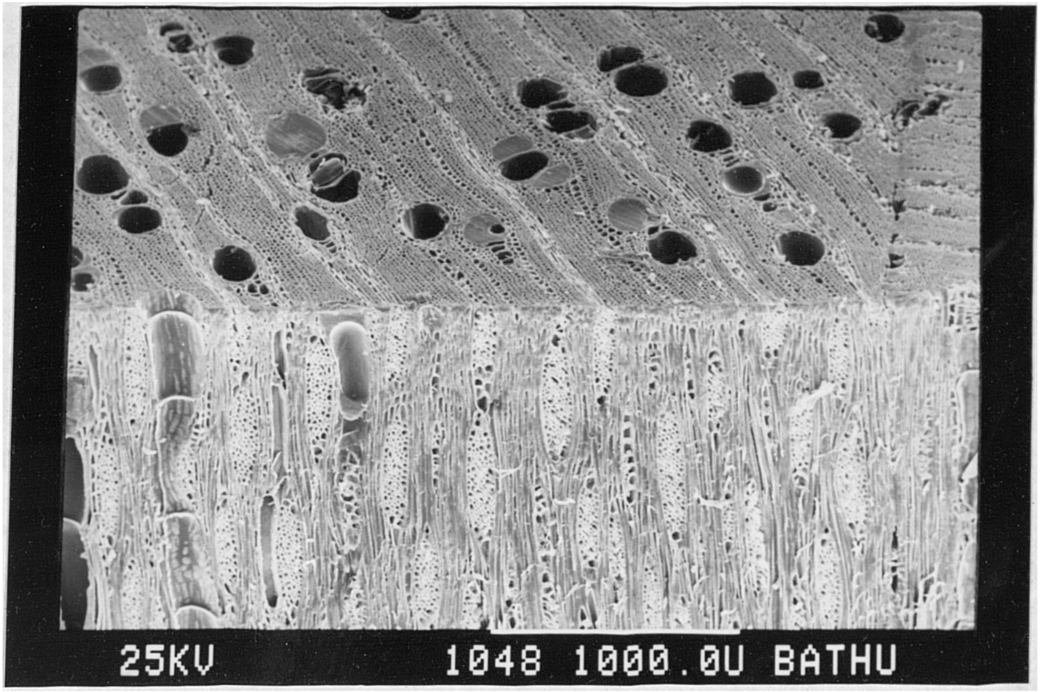


Figure (7.2) Transverse and tangential-longitudinal sections of a Khaya laminate showing a glue line on the far right.

hours before laminate removal. Specimens were then cut to the size of 30mm by 270mm with the longitudinal direction along the length of the specimen. Surfaces were then sanded using a belt sander to remove excess resin before the samples were placed into conditioning cabinets. Specimens were not measured and weighed until just before testing. A total of 14 laminates were manufactured 13 of which were labelled A to N omitting I. Each specimen cut from the laminate was then sequentially numbered, most plates yielding around 35 to 40 specimens. The 14th laminate was a 0.4m long laminate and the specimens cut were labelled KA to KH.

7.2.2 Sitka Spruce and Permali Compressed Laminates

A total of four beams of United Kingdom Sitka spruce were carefully selected and purchased from a local lumber yard. They were chosen specially for straightness of grain and a large radius of curvature of the growth rings. Specimens were cut ensuring that the longitudinal, tangential and radial directions were parallel to the planes of the specimen as shown in figure (7.3). The specimens selected for testing were clear being

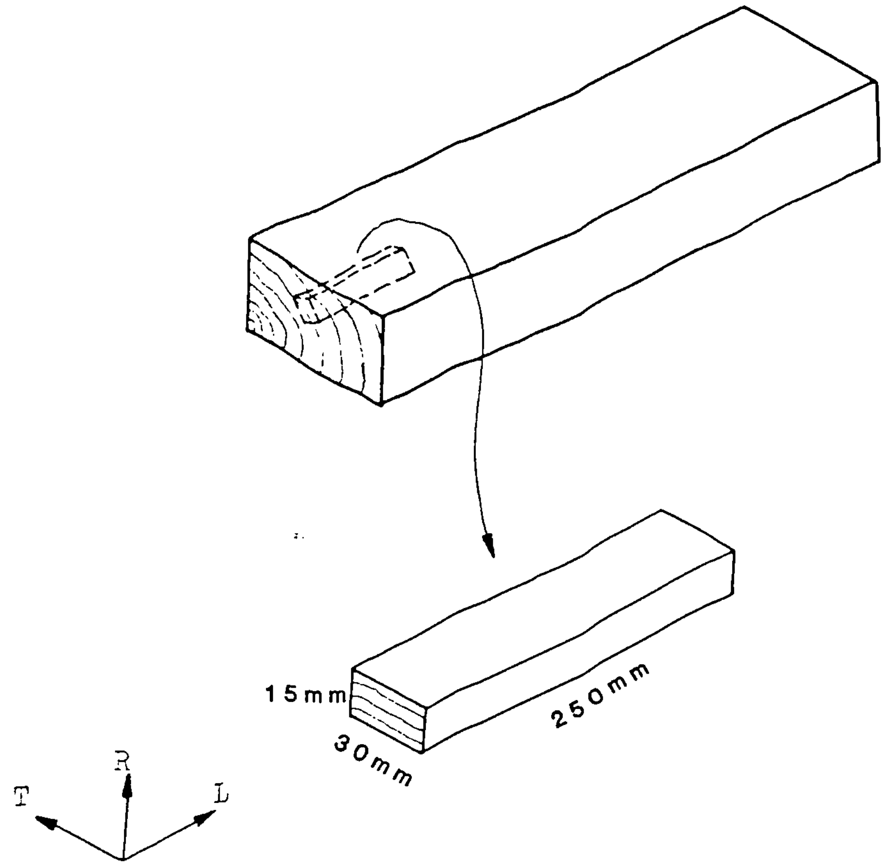


Figure (7.3) Dimensions and orientation of the Sitka spruce specimens.

straight grained and free from knots, resin bands and compression wood. Each specimen was carefully labelled so that its source and location in the beam could be traced. The cut specimens were then placed into conditioning cabinets for at least one month before being measured and weighed for testing.

Compressed Beech laminates were also used initially in this study. These laminates were obtained from Permali Gloucester Limited. They were in two forms; unidirectional and cross (0/90) laminated. The material was 12 plies of Beech veneer thick, laminated using a phenolic resin and compressed with a force of 15 MN/m^2. This type of material is often used in aircraft propellers and in subsonic wind tunnels. The laminates supplied were both about 6.35mm thick and were cut into lengths of 150mm and widths of 20mm. The laminates were cut with the outer plies in the direction of the length of the specimen.

7.3 Environmental Conditioning

With wood properties being strongly influenced by moisture content, careful conditioning is necessary. Three different environments were used. The first was the air-dry condition. Specimens were simply left in the laboratory atmosphere which had a relative humidity of 40%. Most of the specimens tested were conditioned at 65% RH. Glass tanks with a saturated salt solution of sodium nitrite were used to maintain this humidity. To obtain an even higher humidity, a salt spray cabinet was used with distilled water only. This created an environment of 98% RH. The cabinet was carefully adjusted to ensure a low water collection rate to avoid excess condensation of water onto the specimens. The salt spray cabinet design followed the ASTM B117-73 description.

Sliced Khaya conditioned in the three environments attained moisture contents of approximately 5%, 11% and 35%, the last being above the fiber saturation point. Sitka spruce and rotary cut Khaya was conditioned at 65% RH only which also gave a moisture content of about 12-13% and 11% respectively. Compressed Beech wood specimens were not specially conditioned and were tested as received. Their moisture content was not sensitive to fluctuations in the humidity of the laboratory and were at about 8%.

Specimens were conditioned for a minimum of three weeks in each environment. Their moisture content was measured using the oven drying method as described in the ASTM standards. This simply involved measuring the weight before and after drying in a oven with a temperature of between 101°C and 103°C. At various time intervals, a resistance type moisture meter was also used to monitor the moisture content of specimens to ensure that the moisture content was as stated to within 1%.

7.4 Fatigue Test and Monitoring Equipment

The fatigue tests were conducted on a 5kN capacity servo-hydraulic machine supplied by Dartec Ltd. It had a 5kN fatigue rated load cell mounted on a hydraulic jack with ±50mm of travel.The load cell was calibrated to give ±10 Volts fullscale. The jack itself was mounted on a stiff load frame with a manually adjusted crosshead. This is illustrated in figure (7.4). Hydraulic supply was provided by a ≈10 litres/min pump which was aircooled and needed only a standard 13amp single phase supply. The Dartec electronics was separated into two parts, the closed loop servo valve amplifier which was mounted on the frame and a general purpose control console. The control console consisted of various sections. The function generator was capable of generating a ramp function and sinusoidal, square and triangular waveforms up to a frequency of 100 Hz although for the testing of wood, 30 Hz would be the practical limit of the machine. Other sections include a digital volt meter with peak detectors for monitoring the peak loads, a cycle counter, trips to shut the pump off when the specimen fails, and

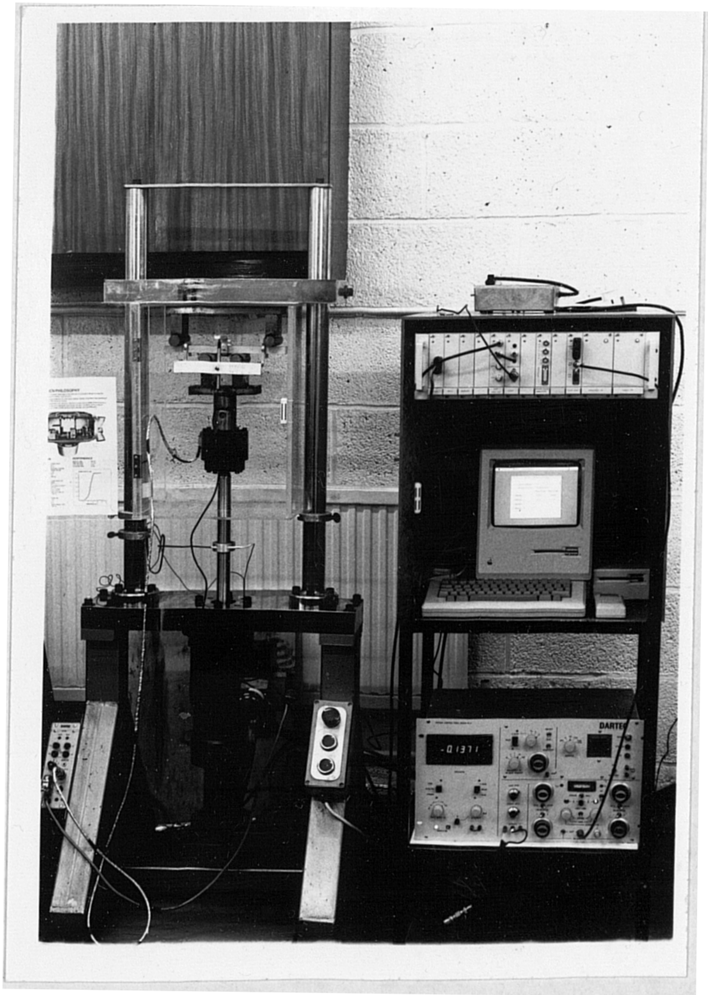


Figure (7.4) Photograph of the Dartec fatigue machine.

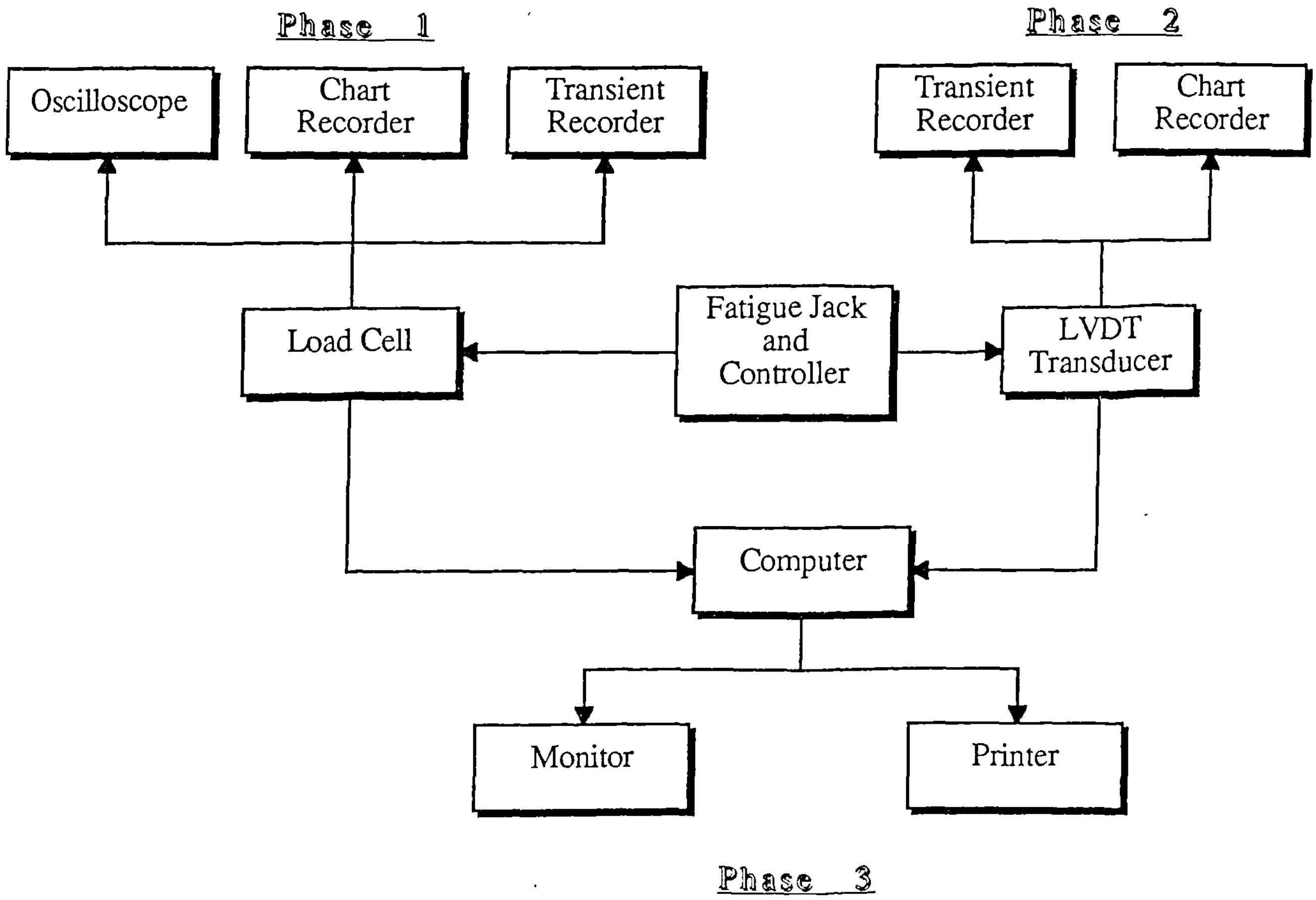


Figure (7.5) Sequence of monitoring equipment added to fatigue machine. SArGen was developed in Phase 3 which made Phase 1 and 2 redundant.

two sets of dials for controlling the command signal to obtain the appropriate peak loads.

Additional monitoring equipment was added in three phases as illustrated in figure (7.5). The first was a transient recorder of 8 bits (or 1 in 256) accuracy to measure the peak failure loads during high speed ramp tests. A second transient recorder was then added to measure deflections from an Linearly Variable Differential Transformer (LVDT) transducer mounted on the specimen (see section 7.5). This set up enabled preliminary measurement of the development of fatigue strains. This setup however has drawbacks. Firstly, the accuracy of the system was poor especially since the drift in the signal due to fatigue strains meant that the sensitivity setting had to be low. A greater problem however was associated with the manual nature of measuring. Generally, the first few hundred cycles were lost due to the need to manually dial up the loads until the peak loads required were reached. For short term tests, this loss of early data would be a significant. Also being manual, the cycles leading up to failure would generally be missed, likewise intermediate events such as the development of cracks. A clear picture of the development of fatigue strains and stiffness changes would therefore not be obtained. The need therefore existed for the third phase which would utilize a computer to control and monitor the fatigue machine. The system would also provide additional features such as complex loading facilities and automatic correction of drifts in the peak loads. This system referred to as SArGen is described fully in Chapter 6. The calibrations of load cell and LVDT transducer with SArGen gave ±100% fullscale for ±5KN or ±8mm respectively, although the transducer was only physically capable of deflecting through ±4mm.

A perspex cabinet was also constructed to fit the fatigue frame to prevent the drying out of specimens during a fatigue test. This drying out can be significant for longer term tests and the cabinet with a beaker of saturated sodium nitrite solution was used to prevent this from occurring. For tests at 98% RH, the fog from the salt spray cabinet was ducted over to the cabinet ensuring that the specimen remained moist.

7.5 4-Point bend Rig

All fatigue tests on Khaya and Sitka spruce specimens were conducted in flexure using a four point bend rig with the supports made from one inch diameter rollers spaced at 70mm apart. This gave a total outer support span of 210mm. The flexural strength, σ_F, is calculated using the formula:

$$\sigma_F = \frac{PL}{bd^2} \qquad .....7.1$$

where P is the applied load, L is the distance between the outer rollers and b and d are the specimen width and depth respectively.

Problems were encountered however with this standard arrangement as, in a fatigue test, the two central rollers had a tendency to dig into the specimen creating a damage zone. This effect did not appear to seriously affect fatigue lives but affected the measurement of deflections. To minimize this, polyethylene pads with a surface curvature of about two inches radius were used spreading out the area of contact and reducing any damage due to the relative movement between

the specimen and rollers. The problem was not totally removed but the damage with the pads was much less and its effect fell within acceptable limits.

For reversed flexural fatigue, the rig had to be greatly modified. The basic four point frame was used but additional rollers were added so that there were now eight rollers in four pairs. These pairs of rollers were attached to each other by means of a steel ring which provided a hinge for movement. The arrangement is illustrated in figure (7.6). The arrangement is by no means ideal. Firstly, as the specimen was fatigued, the rollers inevitably dug into the specimen resulting in premature failure. The use of polyethylene pads as before was not possible as it would add too much thickness

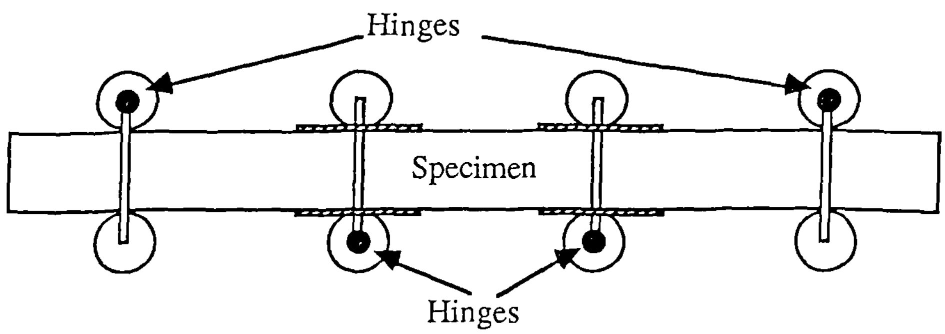


Figure (7.6) Arrangement used for reversed four point bending fatigue. GRP pads were glued to reduce lateral damage at the central load points.

between the rollers. Therefore an alternative method had to be considered and it was found that GRP pads glued to the specimen was the most satisfactory. Woven glass in epoxy resin was used which had the advantage that it was harder than wood, so did not crush easily. Also, since it had a modulus not much greater than wood, it reinforced the area of greatest stress without inducing excessive shear stress between the pads and wood, unlike aluminum pads. The use of these glued on pads however, has the disadvantage of affecting the flexural rigidity of the specimen and the deflections measured would not correspond to the stiffness of wood and can only be used for comparative purposes.

Another weakness of the arrangement is that it does not fully address the manner in which the rollers move as the specimen is loaded in each direction. Ideally, the hinge should be frictionless and located at the line of the neutral axis of the specimen. This would enable both rollers to rotate with the surface of the specimen as illustrated in figure (7.7). The rollers should also be free to rotate to allow for some small expansion and contraction of the specimen when bent. But the ideal arrangement

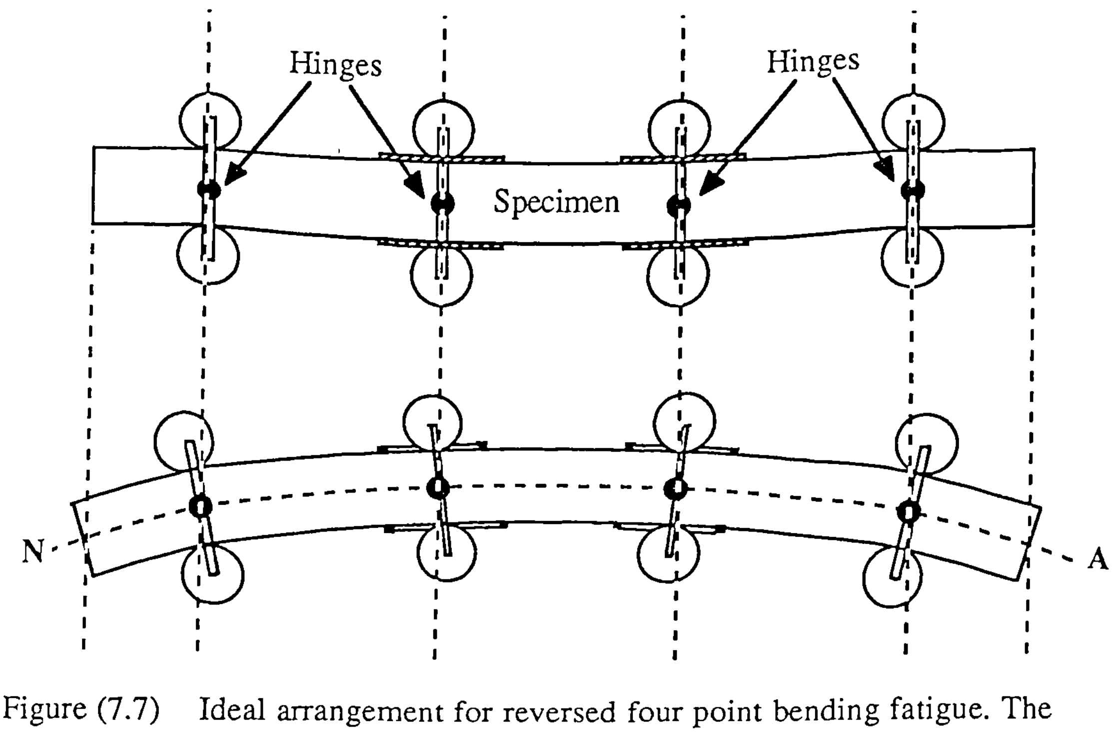


Figure (7.7) Ideal arrangement for reversed four point bending fatigue. The hinge should be frictionless and located at the neutral axis (N - A) of the specimen.

would still have the problem of rollers digging into the specimen, hence it was felt that time expended to redesign and construct a new rig was not justifiable.

Deflections were measured using a LVDT transducer mounted on the specimen by means of an aluminum frame as shown in figure (7.8). Pointed screws were used to hold the frame at the neutral axis of the specimen. The two outer locating points were

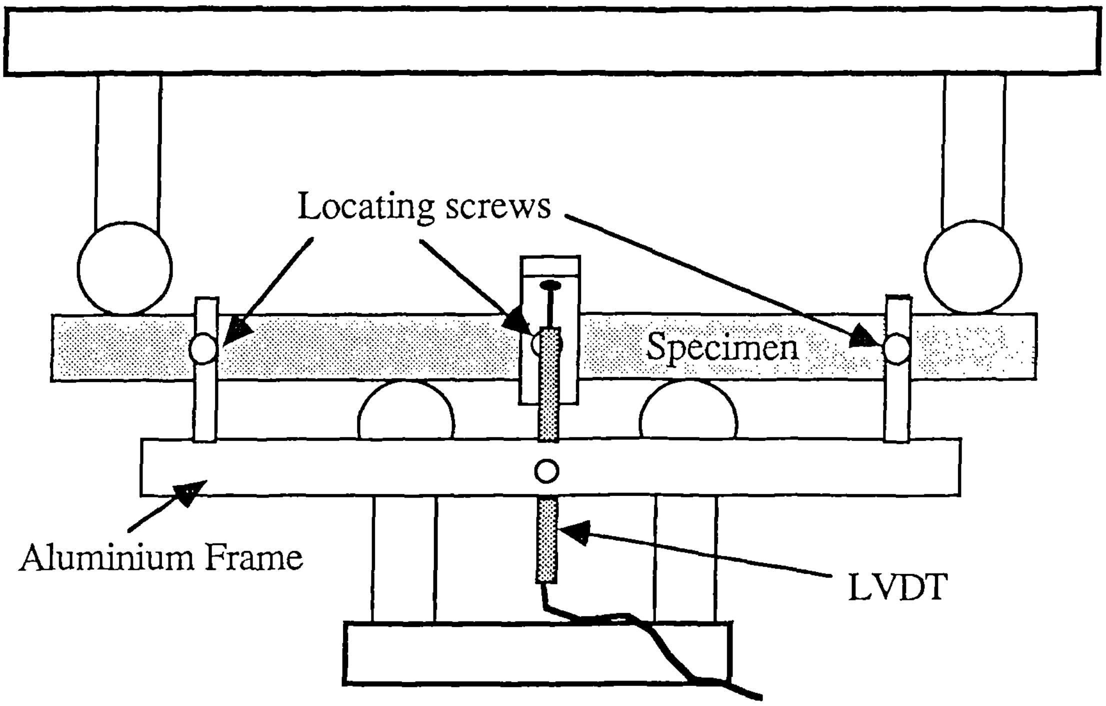


Figure (7.8) Illustration of the rail used to measure centre-point deflections of the specimen.

170mm apart with one at the centre of the specimen. The LVDT transducer therefore measured the relative movement between the two outer and the central points. The deflection measured was therefore non-standard. This was necessary as it would increase the sensitivity of the deflection measured. Also the form of the loading jig imposed considerable restrictions on the possible frame design. The system worked well although the locating points would need improvements for tests exceeding 100,000 cycles as there was a tendency for them to become slack due to the vibrations.

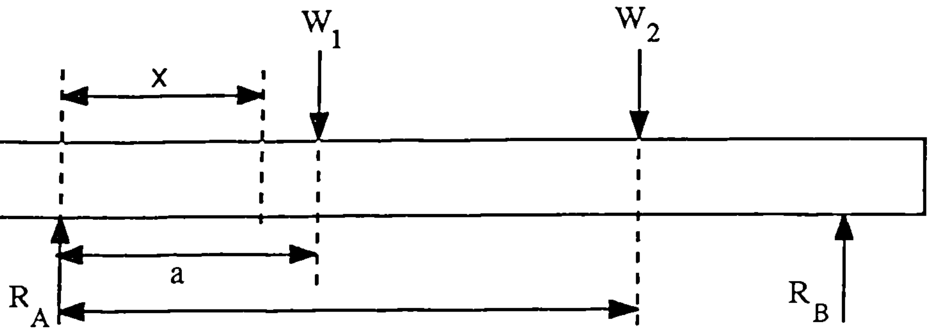


Figure (7.9) Schematic of the terms used in equation 7.2.

The LVDT transducer itself was capable of measuring deflections of ±4mm. It was calibrated to give ±5 Volts fullscale.

Based on this arrangement, no standard formula for modulus calculation exists. The formula has therefore to be derived. The general formula for the deflection, δ, at a distance x from R_A, in a four point test may be given as (Urrey, 1953)

$$EI\delta = -R_A\frac{x^3}{6} + W_1\frac{(x - a)^3}{6} + W_2\frac{(x - b)^3}{6} + Cx + D \quad .....7.2$$

where the terms used are as shown in figure (7.9) and the expressions in the brackets are ignored if negative. The constants C and D may be determined from the conditions at the outer two rollers. For the configuration of the test,

$$D = 0,$$

$$C = \frac{PL^2}{18} \ .$$

Equation 7.2 may be rewritten as

$$EI\delta = -\frac{Px^3}{12} + \frac{P(x - a)^3}{12} + \frac{P(x - b)^3}{12} + \frac{PL^2x}{36} \quad .....7.3$$

where the expressions are again ignored if negative. The deflection measured in the test, δ_T, is the relative movement of the central point with respect to the outer points. Therefore, the equation for the modulus may be expressed as

$$EI\delta_T = EI\delta_1 - EI\delta_2 \quad .....7.4$$

where δ_1 is the deflection for x=105 mm (the center point deflection) and δ_2 is the deflection for x=20 mm. Since L=210 mm, it can be shown that

$$E = \frac{1392250P}{bd^3\delta_T} \qquad .....7.5$$

It must be noted that shear has not been included in the derivation and that its contribution is not insignificant. Care should therefore be exercised when comparing measured values based on equation 7.5 with flexural moduli quoted in the literature. The ASTM standard for modulus calculations based on the three point bend test also ignores the shear contribution. However the smaller shear area of the test used would result in a higher modulus.

Permali compressed Beech wood specimens were tested on the fatigue machine using a four point bend rig with the outer rollers spaced at 85mm apart and the two inner rollers 28.5mm apart. No deflection measurements were made with these tests.

7.6 Microscopy

Sections and fracture surfaces of fatigued and ramp tested specimens were examined under the Scanning Electron Microscopes to study fracture morphology. The work was carried out on two models, JEOL 35C and JEOL T20. Fracture surfaces were first coated with a thin layer (≈15nm) of gold-palladium in a sputter coater using an accelerating voltage of 20kV for 10 mins. Photomicrographs were taken of the fracture surfaces.

Optical microscopy was performed using the technique developed by C.T. Keith (1968) and J.M. Dinwoodie (1966, 1968) to study compression damage in wood. This technique involves taking a 5mm by 5mm by 15mm section of wood and boiling it for 2-4 hours until softened. The section can then be microtomed into 20μm thick sections using a base sledge microtome. Great care was taken to orient the microtome blade to 1.5° to the sample axis in order to avoid inducing compression damage into the microtomed section. This is a common problem as described by

Dinwoodie (1966). Sections were then mounted on microscope slides and examined using polarized light microscopy. Grateful thanks is due to Dr. Dinwoodie for introducing this technique to the author and preparing the slides.

7.7 The Experimental Program

7.7.1 Static Flexural and Compression Tests

The main aim of this work was to characterize the properties of the Khaya laminates, compressed Beech laminates and Sitka spruce. About 20% of the specimens were ramp tested to obtain their flexural strengths using the ramp function of the fatigue machine and the same four-point bending rig as the fatigue tests. The tests were carried out at a constant rate of loading of 40 KN/sec which corresponded to a rate of stress application (RSA) of about 1000 MPa/sec for Khaya and Sitka spruce, and 3900 MPa/sec for compressed Beech specimens.

A number of compression tests parallel to the grain were also made. The specimens for these tests were cut from standard flexural specimens into dimensions 15mm by 30mm by 90mm. These tests were performed using a 100KN, screw driven Instron, model 1195, at a crosshead speed of 1mm/min. A total of 6 sliced Khaya specimens and 15 Sitka spruce specimens conditioned at 65% RH were tested.

7.7.2 Rate of Stress Application Effects

Since wood is sensitive to the RSA of the test, rate effect was examined in a series of tests carried out on the 5% and 11% moisture content sliced Khaya specimen. The RSAs used were 1, 14, 150, 1115, 3230 MPa/sec. giving a range of over three decades. The specimens were all taken from two laminates, one for each moisture content to reduce the spread of results within each family of tests. A series of tests were also performed on 0/90 compressed Beech specimens with RSAs of 5, 50, 600, 4200, and 145600 MPa/sec.

7.7.3 Fatigue Testing

The fatigue tests carried out on the specimens followed various themes. These are listed as follows.

(a) **Comparison of various species and types of wood.**

With five very different types of wood and wood composites, a comparative study was performed by establishing S-N curves for each type. They were all tested at an R ratio of 0.1. Test frequency however was not fixed. The unidirectional compressed Beech specimens were tested at a fixed frequency of 20Hz but the 0/90° specimens were tested at a constant RSA of 2600 MPa/sec. Sitka spruce specimens could not be tested at that rate due to the limitations of the fatigue machine and the rate was therefore reduced to 1600 MPa/sec. Tests on the sliced and rotary cut Khaya laminates were at a slower rate of 1000MPa/sec.

(b) **Moisture effects.**

The three moisture conditions of the sliced Khaya specimens, 5%, 11% and 35%, were compared by fatigue tests at a R ratio of 0.1. The 5% moisture specimens were tested at a fixed frequency of 15 Hz while the other two tests were conducted at an RSA of 1000MPa/sec.

(c) **R ratio effects.**

The R ratio is an important test variable. Five different R ratios were used to generate fatigue curves. Two R ratios, -1 and -0.5, involved stress reversals. The other three R ratios were at 0.1, 0.3 and 0.5. In all the tests a constant RSA of approximately 1000 MPa/sec was used.

(d) **Block loading.**

Within the time scale of the research project, it was impossible to perform a thorough investigation into the problem of complex loading. A wide variety of tests would be necessary. Therefore, an exploratory study in to the possible problems associated with complex loading fatigue of wood was made. Sliced Khaya specimens conditioned at 65% RH were used in all the tests. Until the development of SArGen, block loading had to be done manually. Initially an attempt was made to reduce the burden of manual operation by conducting a two level block test using R=ratios of 0.1 and 0.5. Stress reversals were avoided in view of the problems in the loading rig.

From the S-N curves established earlier, a stress level for the two R ratios were chosen to give a mean failure life of around 5 million cycles as estimated using Miner's Rule. Therefore, for R = 0.1, the stress level of 75% of flexural strength was chosen and for R = 0.5, 85%. Specimens were first fatigued at R = 0.1 for $2*10^5$ cycles followed by R = 0.5 till failure. The sequence was then reversed with tests at R =0.5 for 10^6 cycles followed by R = 0.1 till failure. Two other programs with shorter blocks of two levels but repeated until failure were attempted but it was found to be tedious to conduct manually and only a few tests were performed. These were at two stress levels with R=0.1. Two specimens were tested at 63.2% for 100,000 cycles followed by 73.7% for 10,000 cycles, the sequence repeated till failure. Another two specimens were tested at 60% for 10,000 cycles and 70% for 10,000 cycles.

Following the development of SArGen, more tests were conducted but these samples did not all fail. Changes in peak loads and deflections were recorded during these tests and unfortunately due to the size of data files created, the tests automatically terminated with less then 250,000 cycles tested. There were also difficulties with over loads during the transitions from one load level to another. The software for SArGen was later improved but the limited time available meant only a few tests could be performed.

7.7.4 Fatigue Testing with SArGen

Tests with SArGen were carried out initially on sliced Khaya conditioned at 65% RH. These tests were conducted at an R ratio of 0.1 with the aim of looking at the changes in stiffness and fatigue strains from the start of the tests till failure. Most tests were therefore conducted between the stress levels of 85% to 70% of flexural strength. Subsequently, stiffness and strain changes at low stress levels were also felt to be of interest hence tests were conducted without taking them to failure, up to 150,000 cycles.

A more comprehensive examination of the effect of stress level on the changes in stiffness and fatigue strains were carried out on Sitka spruce conditioned at 65% RH. These tests were carried out at R = 0.1 with tests conducted till the data recorded by SArGen was no longer meaningful due to movement of the LVDT frame caused by vibration. A number of tests on rotary cut Khaya were also made at 75% of flexural strength with R = 0.1.

The effect of stress reversals was also examined with a series of tests at different load levels and at R = -1. Tests were conducted on both sliced Khaya and Sitka spruce conditioned at 65% RH.

7.7.5 Microscopy

The work on microscopy is relatively limited. Examination of sliced Khaya specimens failed in ramp tests and fatigue tests were made using the SEM for comparison of microstructural differences. Using the optical microscope technique to examine compression creases, the development of damage in fatigue of Sitka spruce was studied. Closely matched specimens from the same block were used. They were fatigued at 75% of flexural strength at R = 0.1 to 100, 500, 1000, 10,000 and 100,000 cycles. Each specimen were then microtomed at the region of maximum compressive stress and examined for structural changes.

Chapter 8

Results and Discussion of Static and Fatigue Tests

8.1 Density and Static Mechanical Properties

8.1.1 Results

The properties of the materials tested are summarised in table (8.1). In all cases, the flexural strengths were measured although for sliced Khaya and Sitka spruce, the main species investigated, the properties were more thoroughly characterized. A total of 86 specimens of sliced Khaya at 11% MC and 27 specimens of Sitka spruce were statically tested. The density results included the densities of the fatigue specimens. Early tests on sliced Khaya at 5% MC and the Permali compressed Beech laminates did not include density measurements and no data was available. In addition to the data in table (8.1), the flexural modulus of Sitka spruce was also measured using the high sampling rate available with SArGen. The modulus, based on 14 specimens was found to be 11.9 GPa with a standard deviation of 1.5. The compression strength were measured using the Instron 1195 and was tested at a relatively low speed of 0.1 mm/min or approximately 0.05 MPa/sec, more than four decades slower than the stress rate of the flexural tests.

Table 8.1 The density and basic static mechanical properties of species tested.

	Moisture	Flex. Str.(MPa)			Density (kg/m^3)			Comp. Strength(MPa)		
Material	Content	Mean	Std. Dev	No.	Mean	Std. Dev	No.	Mean	Std. Dev	No.
Sliced Khaya	5%	106.74	7.23	18						
	11%	95.58	12.61	86	572.16	50.21	170	43.68	2.66	6
	>35%	67.46	3.37	6	985.7	49.9	19			
Rotary Khaya	11%	109.04	10.85	5	540.42	16.08	19			
Permali UD	8%	255.0	8.6	4						
Permali 0/90	8%	227.88	13.24	7						
Sitka spruce	12%	91.32	4.64	27	416.5	30.2	35	37.66	5.31	15

Compared with the data for solid wood published in Lavers (1983), the flexural properties of both the Khaya laminates and Sitka spruce shown in table (8.1) are much higher. For solid Khaya ivorensis at 12% MC, Lavers reported a flexural strength of only 78 MPa with a density of 497 kg/m^3. While the lamination process with the added resin would account for the 13% difference in densities, this would not be expected to make a major effect on the mean strength. The measured flexural strength was found to be 22% higher. Sitka Spruce at 12% MC from the U.K. was reported to have a flexural strength of only 74 MPa and a modulus of 8.1 GPa with a density of 384 kg/m^3. The density of the Sitka tested was only 8.5% higher and cannot account for the 23% difference in strength and 46.9% difference in modulus. The higher flexural properties measured compared to that reported by Lavers may be due to the difference in test method, a 4 pt. test being used rather than a 3 pt. test. However, as seen in the results on rate effects (section 8.1.3), most of the difference must be due to the high stress rate used. It was significant that the compression strength results measured under similar test conditions (except for the difference in size) were slightly lower than those reported by Lavers. Given the statistical scatter, however, the difference was not significant and it can be concluded that the material tested did not have exceptional properties.

8.1.2 Statistical Variation in Strength and Density properties.

As the aim of the fatigue program on sliced Khaya ivorensis was to test material as used on the HWP-300 Wind Energy Converter blades, the veneers for tests were sourced from the same consignment used for blade manufacture with similar quality control procedures. Since only veneers with obvious defects such as brittle heart, cracks and shakes were rejected, a high variability in properties would be expected. This was due to the fact that the veneers, although sourced from the same consignment, would be from different trees. Therefore a variation in density, chemical composition and cell wall structure would occur from veneer to veneer all contributing to scatter of properties. Even within a veneer or tree, properties will vary due to changes in conditions and rate of growth. To characterize this variability, a large number of tests were carried out. A closer analysis of the sliced Khaya and Sitka spruce results was therefore possible.

About 20% of specimens from each laminate of sliced Khaya manufactured were statically tested. Results for the tests and the densities of the specimens from each laminate is summarised in table (8.2). Considering all specimens together, it can be seen that the scatter of bend strengths is quite large, ranging from 60.15 to 123.2 MPa with the mean of 95.58 MPa occurring approximately at the middle. A large scatter in results also exists for the densities which range from 488.1 to 717.1 kg/m^3 with a mean of 563.4 kg/m^3. Using a simple statistical test for comparison of two normal distributions with known variances, it was found that there exists a difference in strength and densities between laminates E, F and G and laminates J, N and KA at the 95% confidence level. Some laminates are therefore significantly stronger and denser then others. However, taken as a whole, the standard deviation remains quite small and is of the order of that of individual laminated boards.

Table 8.2 Flexural strength and density of sliced Khaya ivorensis laminates

Laminate	No. of	Flexural Strength (MPa)			Density (kg/m^3)		
	Samples	Mean	Std. Dev	Range	Mean	Std. Dev	Range
C	5	96.93	7.01	18.29	-	-	-
D	4	95.68	1.18	2.3	-	-	-
E	4	108.10	13.49	28.10	698.4	17.8	34.4
F	5	112.73	8.72	21.11	608.9	44.8	107.8
G	5	106.27	12.10	32.54	585.3	9.4	26.1
H	11	96.78	11.49	39.66	566.3	23.0	77.3
J	5	88.62	7.10	17.47	527.2	3.9	10.8
K	8	94.97	8.31	22.73	547.9	16.2	45.9
L	7	91.41	12.26	28.58	551.8	31.6	91.4
M	12	93.80	15.00	57.53	544.7	35.2	110.2
N	12	89.64	14.54	43.06	561.5	15.2	47.2
KA	8	89.59	7.29	20.38	528.5	15.9	50.5
Total	**86**	**95.58**	**12.61**	**63.05**	**563.4**	**44.4**	**229.0**

The distribution of the flexural strengths can also be seen in figure (8.1) as a cumulative failure probability plot based on a median ranking of data. The fitted normal distribution function of the data is also plotted with the data. The figure clearly shows that the distribution closely fits that of a normal distribution. The good fit reinforces the use of the total sample of specimens as the basis of estimating the strength of the sliced Khaya laminates. The cumulative probability plot of the density is shown in figure

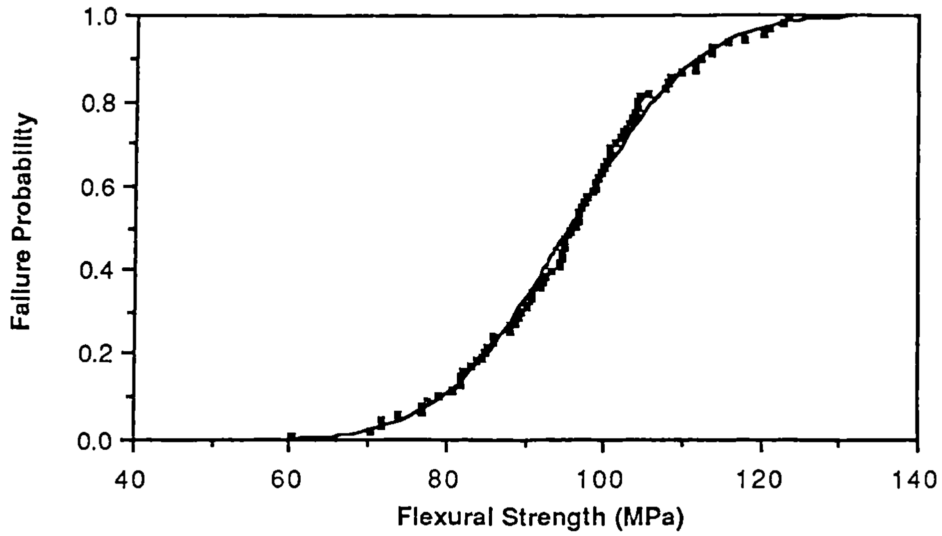


Figure (8.1). Cumulative probability plot for the static flexural strength of sliced khaya laminates. The curve shows the normal distribution function fitted to the data.

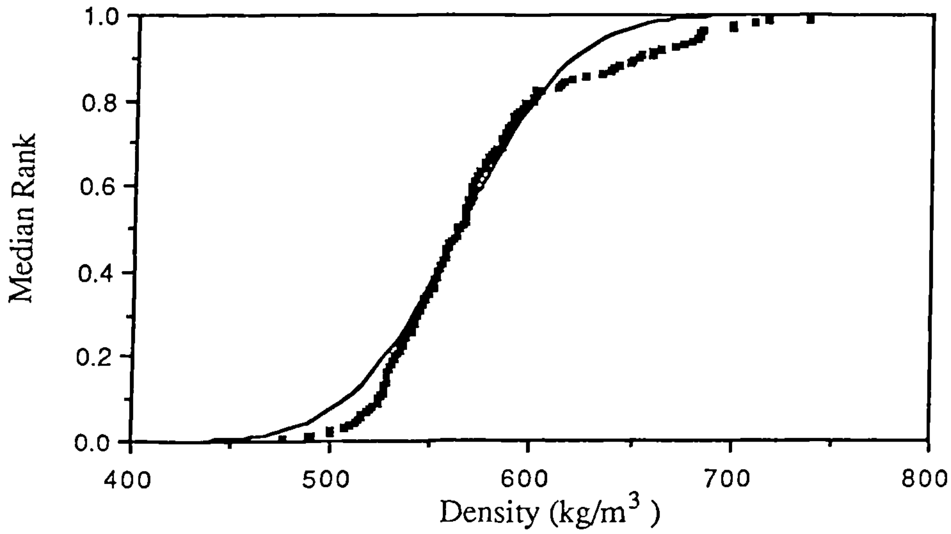


Figure (8.2). Cumulative probability plot for the density of sliced khaya laminates. The curve shows the normal distribution function fitted to the data.

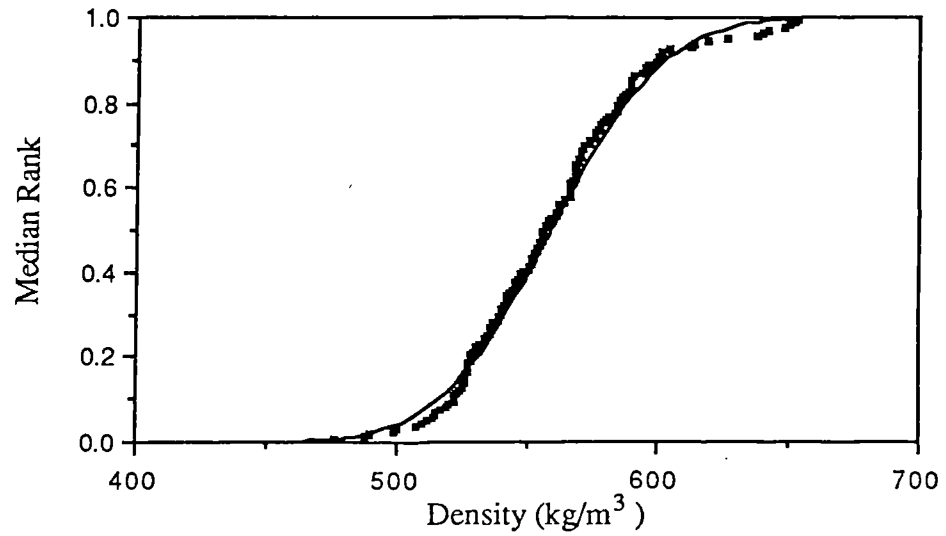


Figure (8.3). Cumulative probability plot for the density of sliced khaya laminates with laminate E omitted. The data shows an improved fit to the normal distribution function.

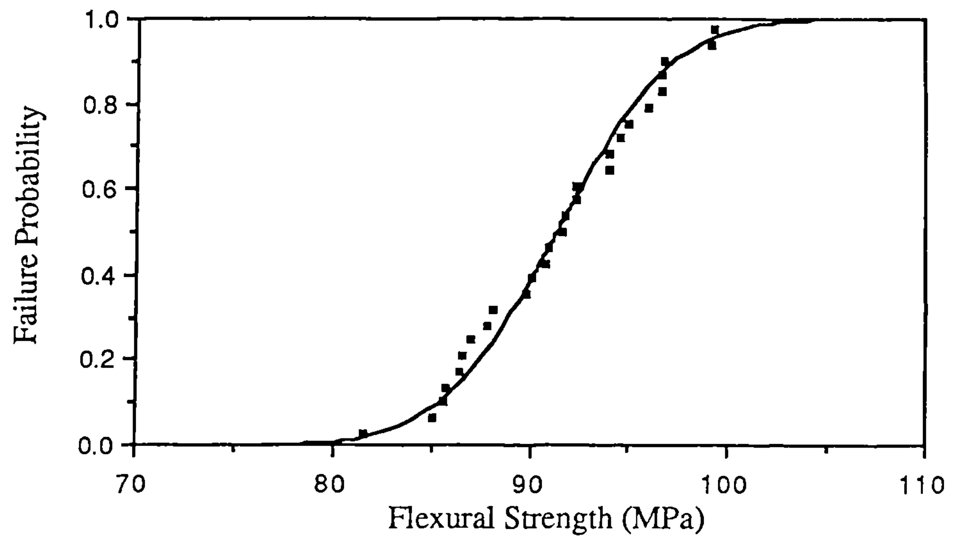


Figure (8.4). Cumulative probability plot for the flexural strength of Sitka spruce with the normal distribution function fitted to the data.

(8.2). Unlike the strength plots however, the data does not closely follow a normal distribution. A closer look at the results reveal however that a bias in density was created by an exceptionally high density laminate labelled E. Laminate E was 90kg/m^3

greater in density than the next highest density laminate. When the data was plotted omitting this laminate, the density distribution follows closely a normal distribution as seen in figure (8.3). Laminate E is therefore an exceptionally dense laminate although this does not appear to affect the flexural strengths. The laminate was visibly darker reflecting the high extractive content of heartwood.

The flexural strengths distribution of Sitka spruce similarly followed a normal variate as seen in figure (8.4). The density probability plot however does not follow the normal function. Figure (8.5) shows that like Khaya, it is at the high density tail of the distribution that the deviation occurs. That the distribution in strength remains normal despite the skewness in the density distribution suggests that a non-strength affecting factor, such as extractive content, was influencing the density of the specimens.

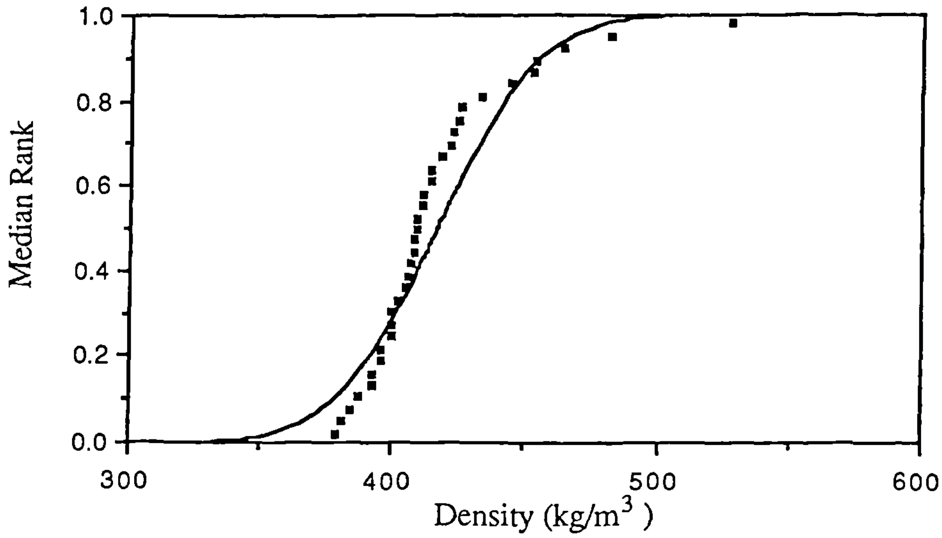


Figure (8.5). Cumulative probability plot for the density of Sitka spruce with the normal distribution function fitted to the data.

8.1.3 Effect of Stress Rate

The effect of stress rate was studied with 5% MC and 11% MC sliced Khaya and 0/90 Permali compressed Beech laminates. At 5% MC, the results, as shown in figure (8.6), indicate no increase in strength over a three decade change in stress rate.

Any effect, if present was totally masked by the scatter of results. Figure (8.7) however, shows that for 11% MC, there was a significant increase in strength. The data is based on tests using one laminate and there was a very small scatter of results with a strength increase of over 10% from 10^0 to 10^3 MPa/sec. It is therefore important

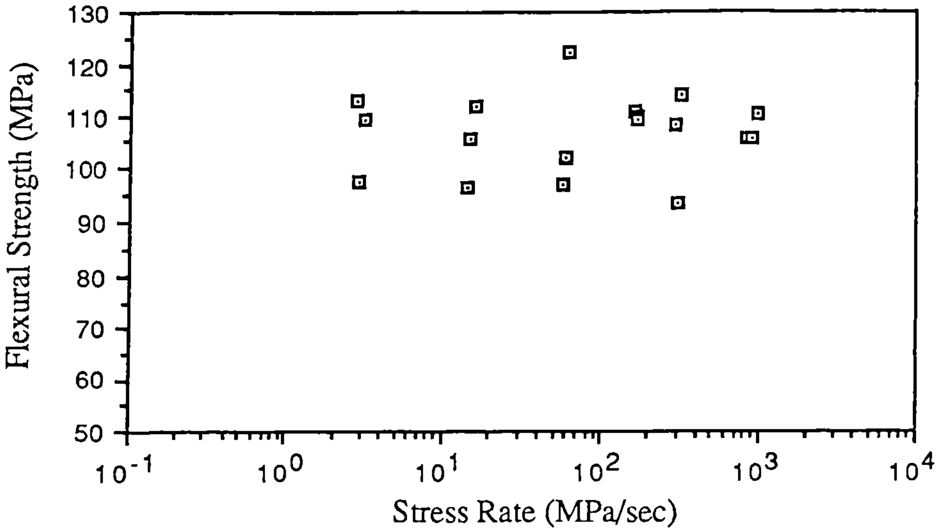


Figure (8.6). Effect of stress rate on the flexural strength of 5% MC sliced khaya laminates.

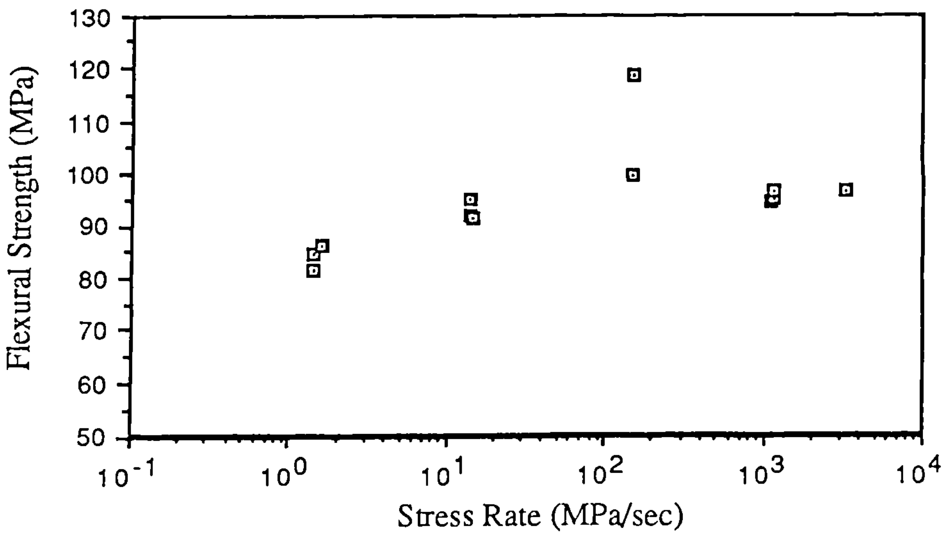


Figure (8.7). Effect of stress rate on the flexural strength of 11% MC sliced khaya laminates.

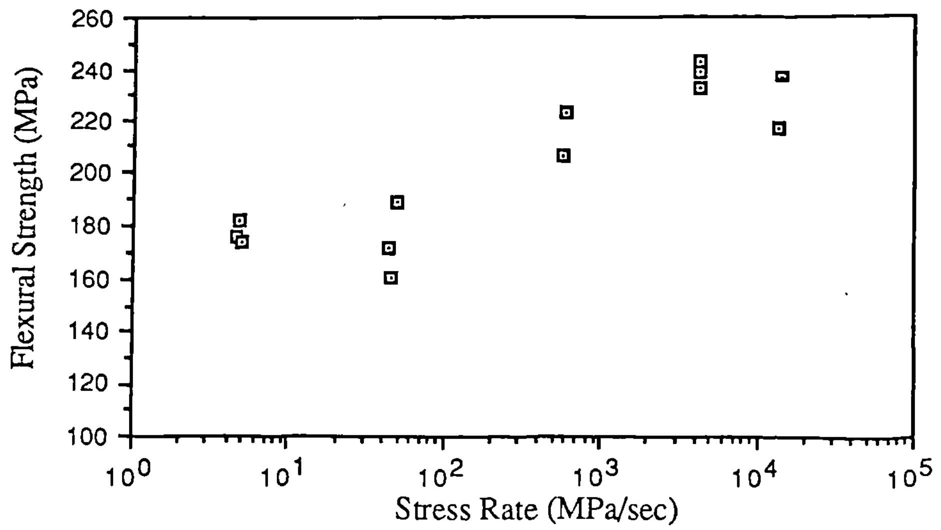


Figure (8.8). Effect of stress rate on the flexural strength of 0/90 Permali Compressed Beech laminates

that the estimation of flexural strengths for fatigue tests be based on tests at the appropriate stress rate. Figure (8.7) also shows that above 10^2 MPa/sec, the strength increase is not significant supporting the result of Nadeau et. al. (1982).

For 0/90 Permali laminates, the stress rate also greatly affects the flexural strength as shown in figure (8.8). This sensitivity to stress rate can be explained by the opening of flaws in the 90° laminae, following the slow crack growth theory of Nadeau et. al. (1982).

8.1.4 Relation between Strength, Modulus and Density

The relation between strength, modulus and density are well established as reviewed in Chapter 2. However it is of interest here to examine these relationships for Khaya and Sitka spruce specimens to determine if they are sufficiently reliable for reducing the scatter of the results in the fatigue tests. For sliced Khaya at 11% MC, figure (8.9) shows that a relationship does exist between flexural strength and density. The correlation of the data is however quite low (0.46) due to the fact that most of the data is concentrated around the average density. As noted earlier, laminate E also created a bias in the results and it may be concluded that for sliced Khaya, any attempt to use density for correction of estimated static strength is futile.

Figures (8.10) and (8.11) shows the relation of flexural strength to density and modulus respectively for Sitka spruce. Again the correlations of the regression lines

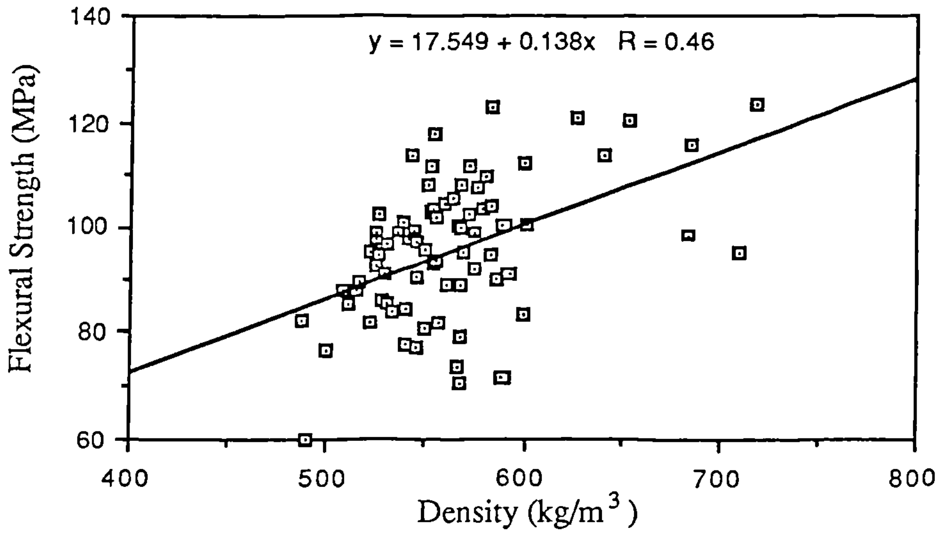


Figure (8.9). Flexural strength versus density plot of 11% MC sliced Khaya laminates.

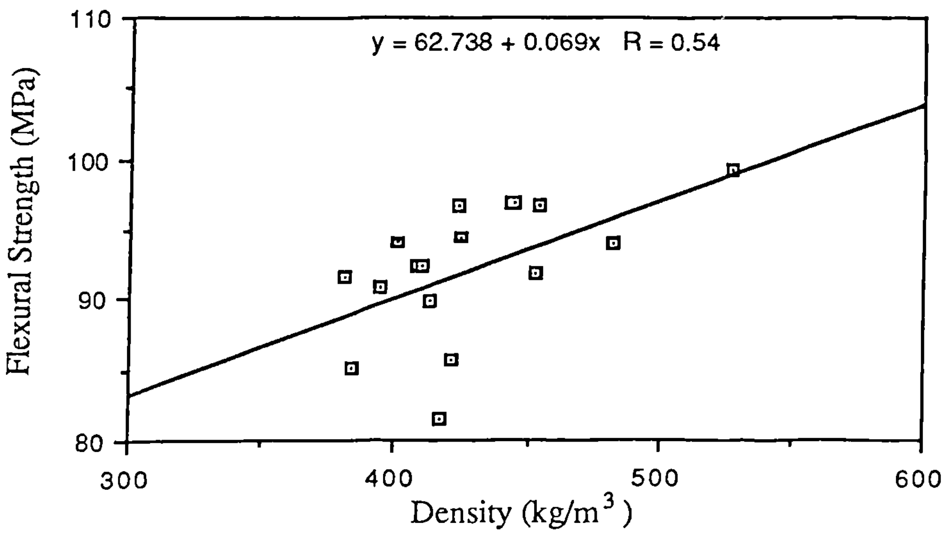


Figure (8.10). Flexural strength versus density plot for Sitka spruce.

are not high. However, when the compression strengths are plotted against density, figure (8.12), the correlation is much greater and the regression line extrapolates to nearly zero strength at zero density. It therefore suggests that where compression strength is to be estimated, the regression equation may be used from the measured specimen density.

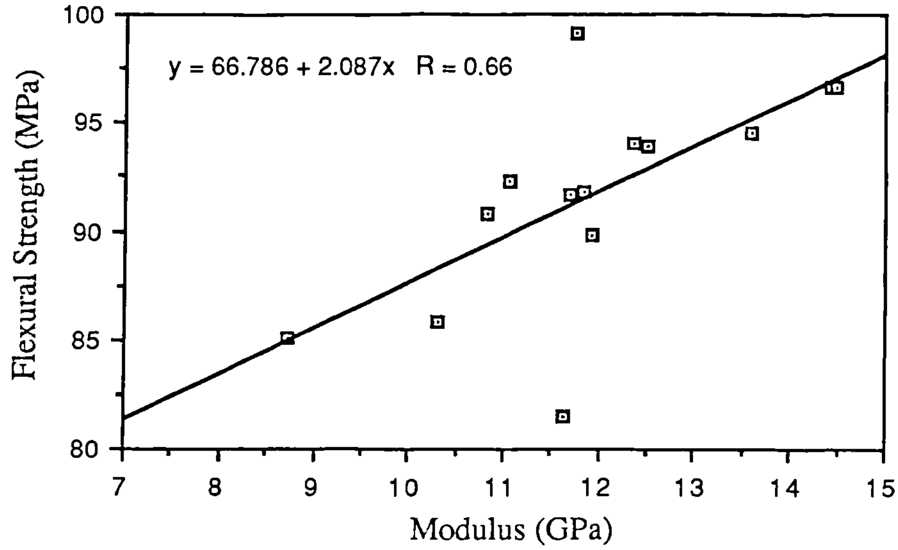


Figure (8.11). Flexural strength versus modulus plot for Sitka spruce.

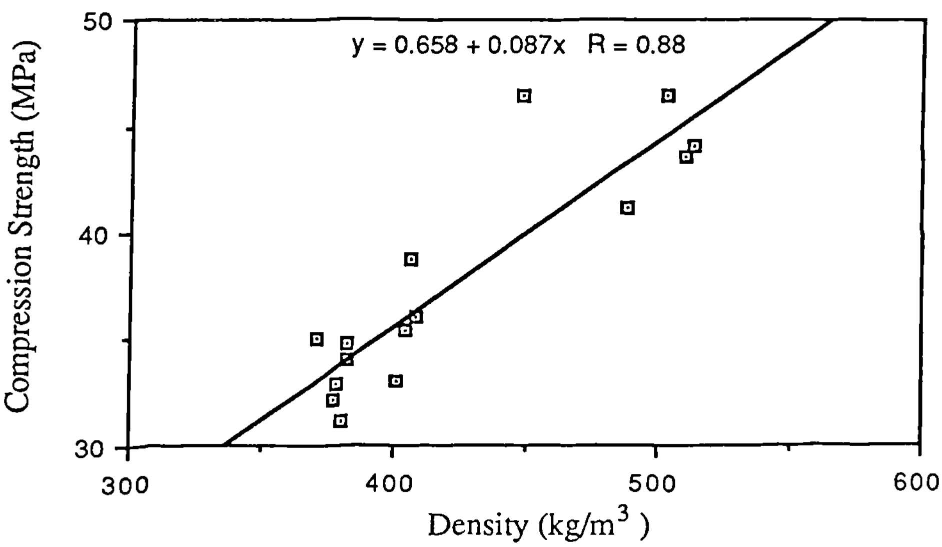


Figure (8.12). Compression strength versus density plot for Sitka spruce.

8.2 Fatigue Life of Various Types of Wood at R = 0.1

8.2.1 S-N Fatigue curves at R = 0.1

The fatigue curves for the five different types of wood and wood laminates are shown in figures (8.13) to (8.17) as plots of maximum applied peak stress against the logarithm of the number of cycles (S-N curve). The fatigue curves for Sitka spruce, rotary cut Khaya and Permali 0/90 laminate support the log-linear model for the fatigue of wood.All their respective linear regression lines extrapolate to values very close to the measured static strength. For sliced Khaya (figure(8.14)) however, the line extrapolates to a value nearly 9MPa less than that measured. This however is still within one standard deviation of the scatter of measured values and with the small gradient of the line, it is unlikely that the fatigue behaviour of sliced Khaya is significantly different from the other three.

The unidirectional Permali compressed Beech laminates however, show a strong non-linear behaviour for low cycle fatigue. It would be unlikely for the fatigue

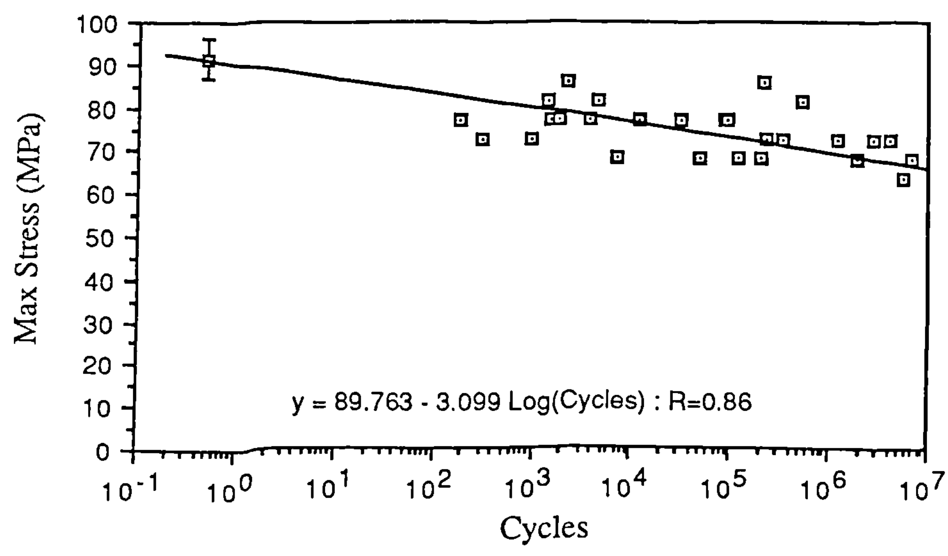


Figure (8.13). S-N fatigue curve for Sitka spruce with regression line through data. The static strength and its error bars indicating one standard deviation is also included in the plot.

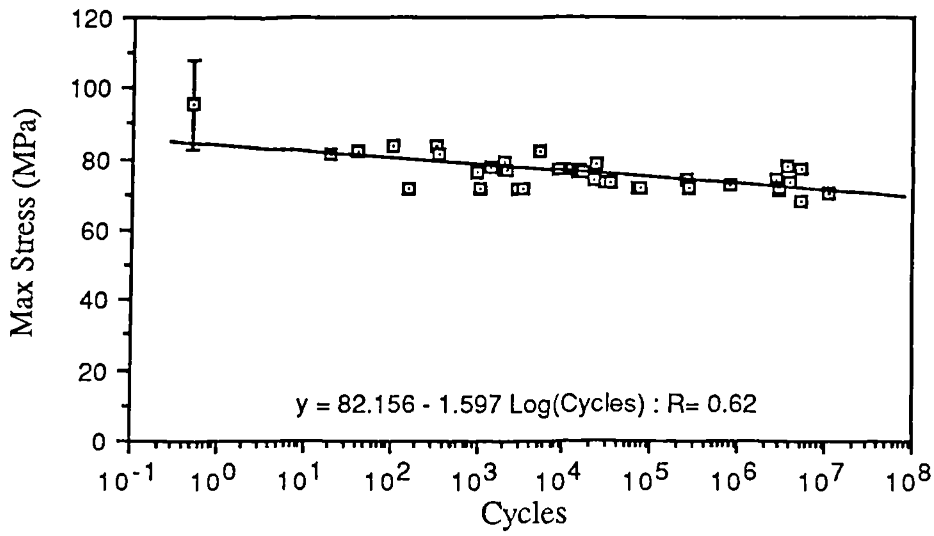


Figure (8.14). S-N fatigue curve for 11% MC sliced Khaya laminates with regression line through data but excluding the static strengths. The static strength and its error bars indicating one standard deviation is also included in the plot.

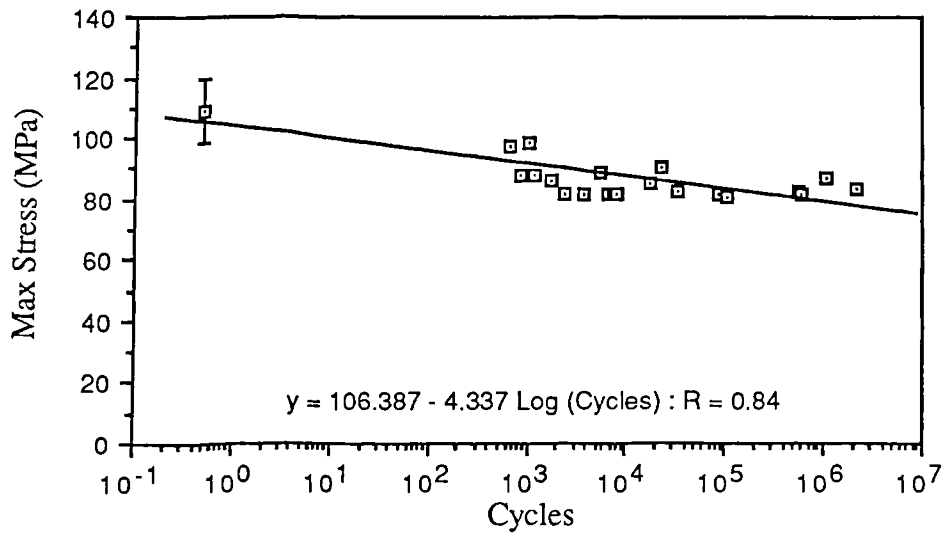


Figure (8.15). S-N fatigue curve for rotary cut Khaya with regression line through data. The static strength and its error bars indicating one standard deviation is also included in the plot.

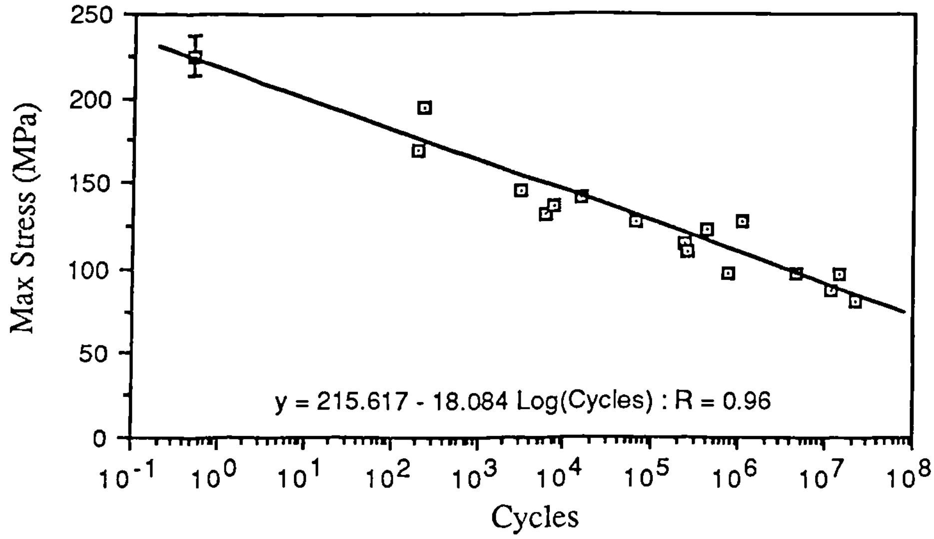


Figure (8.16). S-N fatigue curve for 0/90 Permali compressed Beech laminates with regression line through data. The static strength and its error bars indicating one standard deviation is also included in the plot.

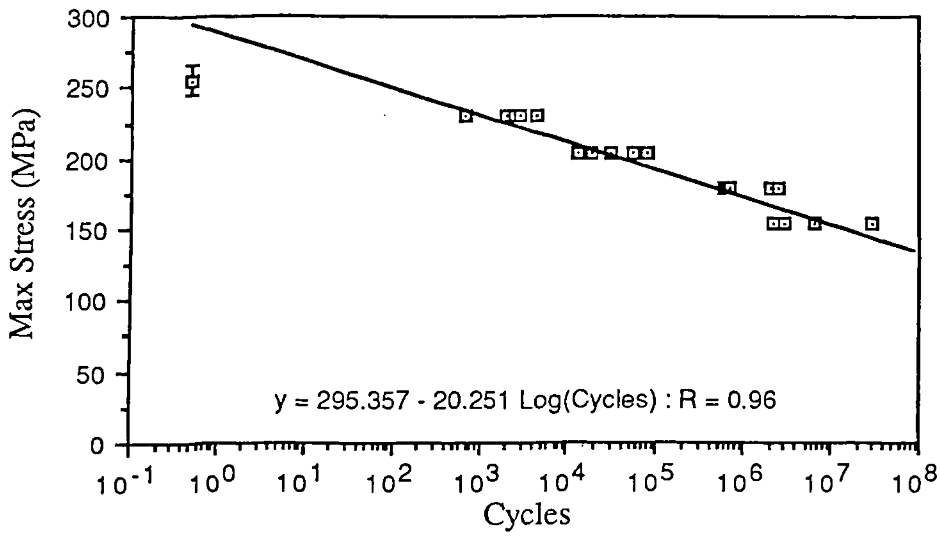


Figure (8.17). S-N fatigue curve for unidirectional Permali compressed Beech laminates with regression line through data. The static strength and its error bars indicating one standard deviation is also included in the plot.

strength to be greater than the static strength hence the curve must flatten towards the static strength for fatigue lives of less than 1000 cycles. A comparison of the fatigue curves plotted together in figure (8.18) and in their normalized form in figure (8.19) suggests that the Sitka spruce and Khaya laminate fatigue curves differ greatly to the Permali compressed beech laminate fatigue curves. While the curves for Sitka spruce and Khaya laminates appear to have similar gradients when normalized (especially if the line for sliced Khaya is made to regress through its static strength), those for the compressed laminates have greater and almost equal gradients. This would indicate that the two compressed Beech laminates have a similar fatigue behaviour distinct from the other three. This is not surprising since the structure of the compressed laminates is very different from the uncompressed Khaya and solid Sitka. The cellular structure of wood is destroyed in compressed laminates to give a more homogeneous material less susceptible to compression damage. Furthermore, since the tests are in flexure, the unidirectional and 0/90 laminates behave quite similarly with the highly stressed outer plies of the laminates being in the same direction. The 90° plies of the 0/90 laminate do not contribute significantly to the strength and serve more as a shear stress transfer medium. Also where tensile cracks occur in the 90° plies, they act as stress

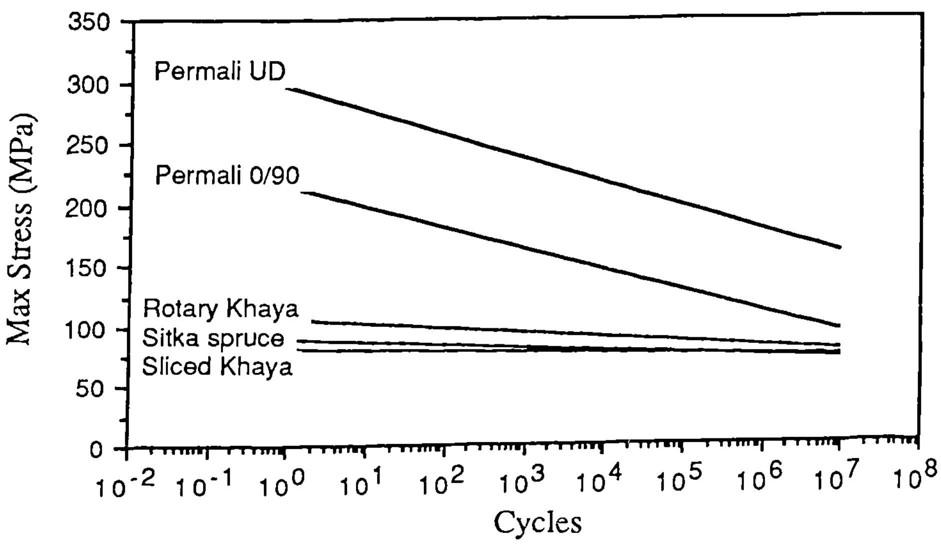


Figure (8.18). A combined plot of the regression lines of figures (8.13) to (8.17).

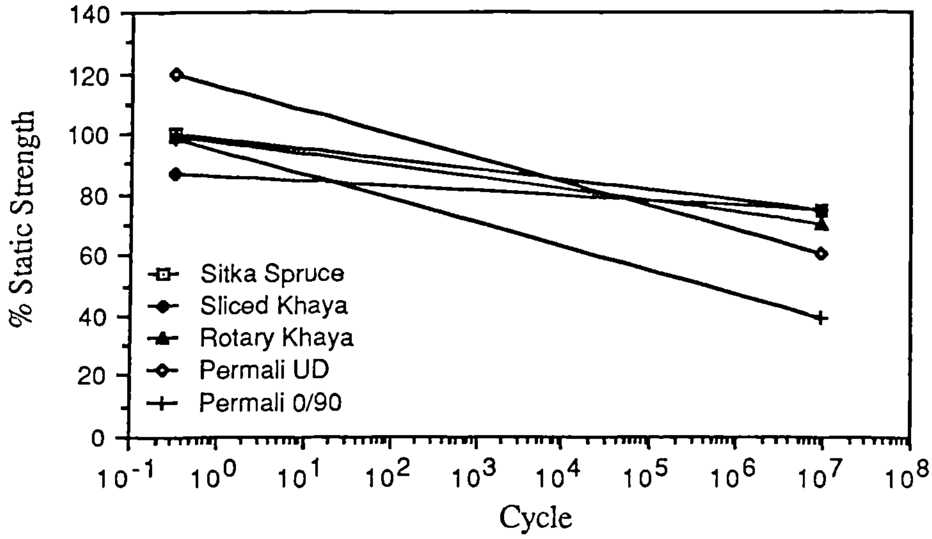


Figure (8.19) A combined plot of the regression lines in figures (8.13) to (8.17) with maximum stress expressed as a percentage of static strength.

concentrators. However, as the cracks would be against plies in the direction resistant to crack growth their role would be more severe in fatigue in the form of slow crack growth. Fatigue crack growth studies on the 0/90 laminates (Ansell and Tsai, 1984) have shown this slow fatigue crack growth.

8.2.2 Assessment of the Scatter of Fatigue Data

It should be noted that the scatter in results as shown in the fatigue graphs is quite high. With the larger data set of 11% MC sliced Khaya, further analysis of the nature of this scatter was made. First, an improvement was obtained when the stresses were normalized using their respective laminate strength as listed in table (8.2). This new plot was assessed using a log-linear regression analysis with a constant of 100% assumed. This ensured that the line regressed through the static strength of the slice Khaya. This is shown in figure (8.20). The error of the data from the model, called residuals, was stored as the scatter in fatigue strength. The residuals can then be analysed for comparison with the scatter of the static strength.The regression model is given by the equation,

$$\% \text{ Static Strength} = 98.8 - 4.038 \log(\text{Cycles}) \quad .....8.1$$

The residuals to this regression model had a standard deviation of only 4.77% of static strength. Figure (8.21) shows the cumulative probability plot and its fit to the normal

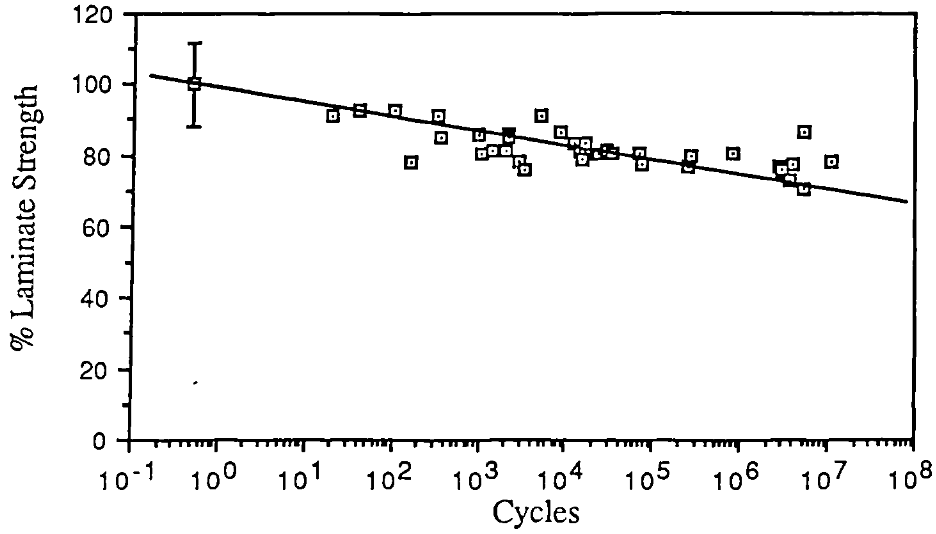


Figure (8.20). Fatigue plot for 11% sliced Khaya with fatigue strength normalized with the laminate strength of the specimen. The log-linear regression model used in the statistical analysis is also shown.

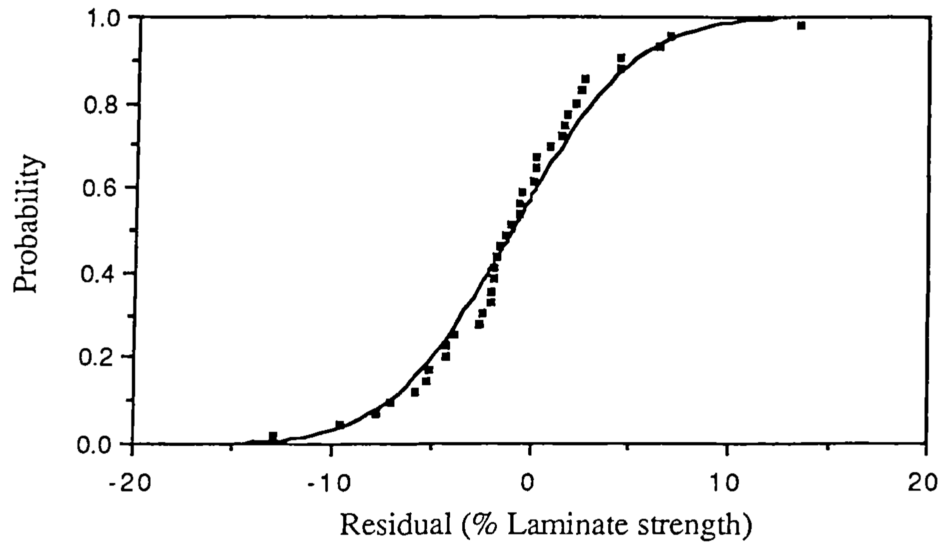


Figure (8.21). Cumulative probability plot for the residuals in fatigue strength based on data normalized with laminate strength.

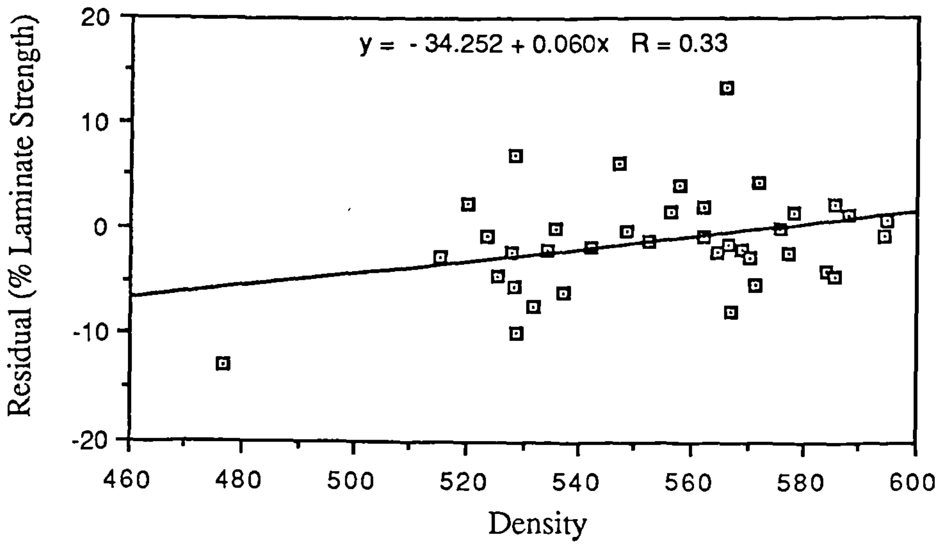


Figure (8.22). Relation between the residuals of fatigue strength with density.

distribution function. The fit is reasonable and the small scatter in results suggests that the log-linear model with a constant assumed was reasonable. Also like the static data, there was a similar correlation between the residuals and density as figure (8.22) shows. Further similar analysis of rotary cut Khaya and Sitka spruce confirmed this model.

8.3 Effect of Moisture on Fatigue Life

Table (8.1) shows clearly that with the increase in moisture content, the flexural strength of sliced Khaya is greatly reduced. This is also well established in the literature but in fatigue, it is important to consider if the reduction in fatigue life is proportional to the reduction in static strength. The data for the three moisture contents of sliced Khaya assessed (5%, 11% and >35%) is plotted in its normalized static strength form in figure (8.23). Moisture can clearly be seen to have a strong effect on fatigue strength and the reduction in strength cannot be factored based on the static properties. Plotting the gradients against the moisture contents shows a very good linear relationship. Figure (8.24) shows this with the regression equation included. It should be noted that there is some error involved in the moisture determination particularly for the 35% moisture content results. Nonetheless, the trend is clear and extrapolating the line towards zero moisture level gives a gradient of -2.5, suggesting that fatigue still occurs and is not negligible. Also since 35% MC is about the fibre saturation point of wood, the trend

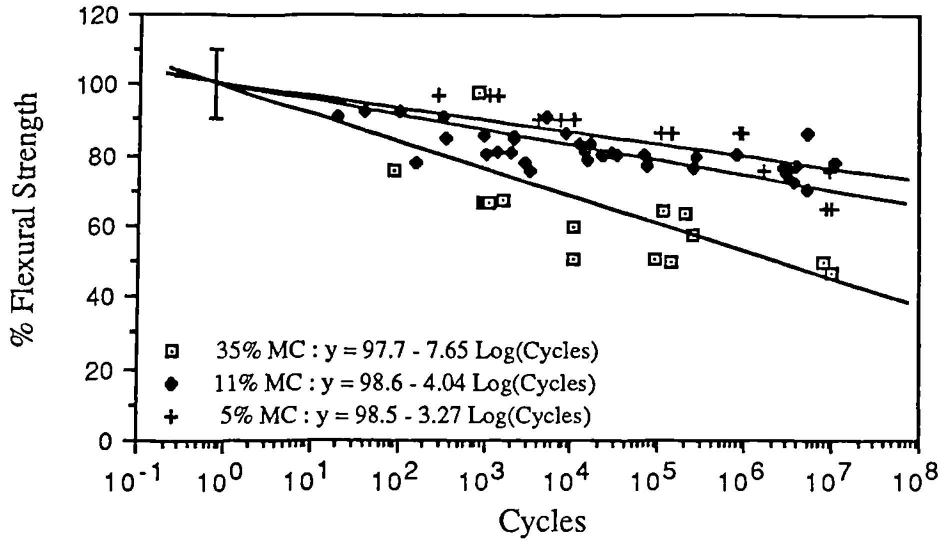


Figure (8.23) The effect of moisture content on sliced Khaya laminates fatigued at an R ratio of 0.1. The maximum peak stresses is expressed as a percentage of static flexural strengths.

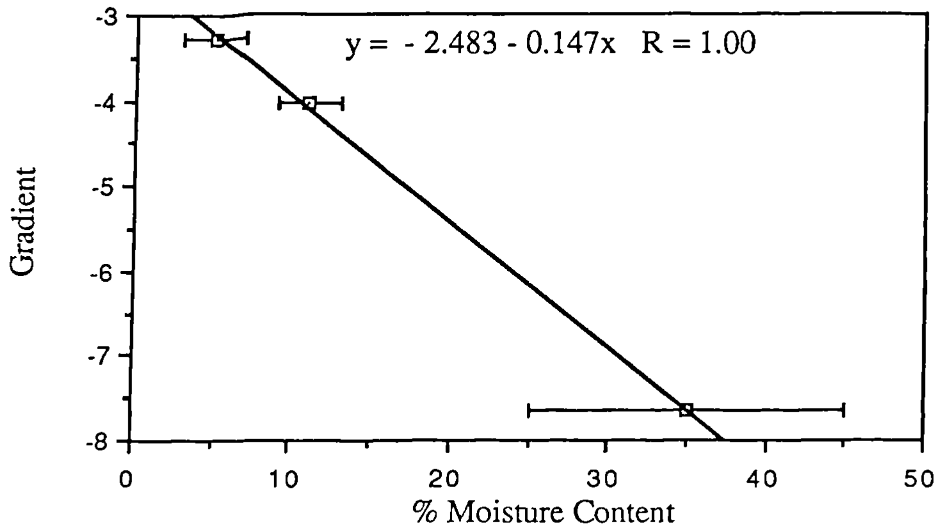


Figure (8.24) Relation between the gradient of the regression lines in figure (8.23) and the moisture content.

beyond 35% may not follow the line but, like the effect of moisture on compression strength, may level off. A further element of the effect of moisture can be seen in figure (8.25) where the fatigue data is plotted in absolute stress terms. The curves show clearly that moisture has not only affected the static strength but also the fatigue process. The higher the moisture content, the greater the gradient of the line indicating a higher rate of fatigue damage accumulation.

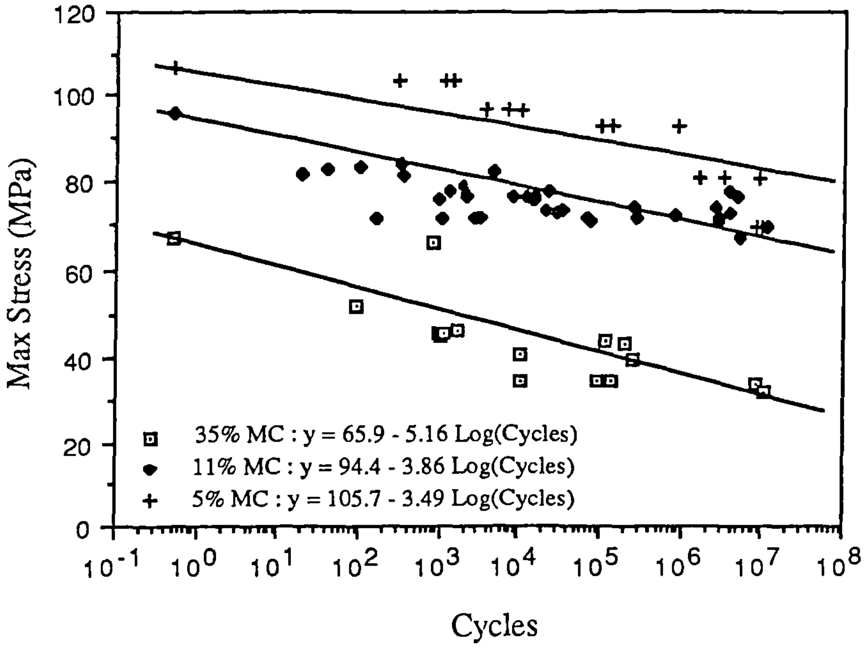


Figure (8.25) Data in figure (8.23) replotted as absolute peak stresses.

8.4 Effect of R ratio

The R ratio as noted in chapter 6, defines the ratio of the minimum to the maximum applied stress. It includes within its value therefore an indication of the stress range and through its sign, any stress reversals. Therefore in a discussion of the effect of R ratio, three important variables are being incorporated,

(a) the absolute stress level,

(b) the stress range,

(c) the combination of tensile and compressive stresses.

The three variables may be considered together in terms of the energy released with each cycle. Low negative R ratios would result in higher energy dissipation rates. This is a consequence of the different stress range and therefore the work done to the system. Compressive damage would be expected to result in a different energy release rate compared to tensile damage.

The results of the effect of R ratio are shown in figure (8.26). As expected the lower the R ratio, the greater the fatigue damage rate with more energy dissipated per cycle and enhanced structural damage rates. The equations of the regression lines drawn for each R ratio follow,

R ratio = 0.5 : $y = 99.5 - 1.66\ Log_{10}(Cycles)$ 8.2

R ratio = 0.3 : $y = 99.2 - 2.74\ Log_{10}(Cycles)$ 8.3

R ratio = 0.1 : $y = 98.8 - 4.04\ Log_{10}(Cycles)$ 8.4

R ratio = -0.5 : $y = 97.9 - 7.13\ Log_{10}(Cycles)$ 8.5

R ratio = -1.0 : $y = 97.5 - 8.40\ Log_{10}(Cycles)$ 8.6

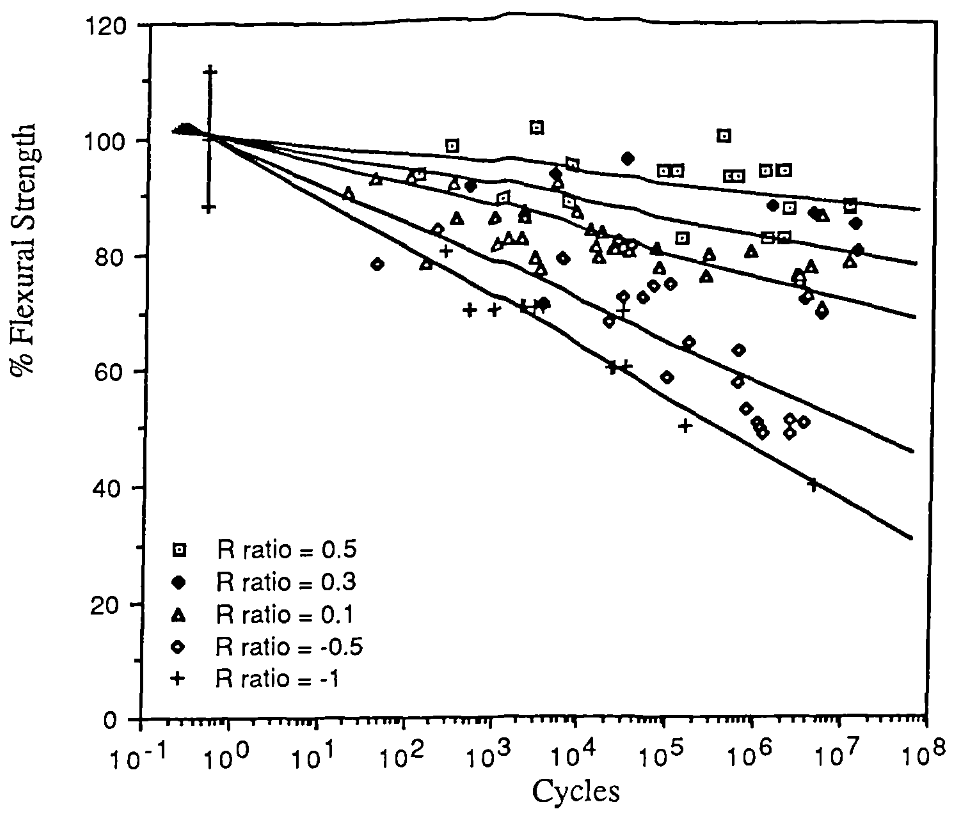


Figure (8.26) S-N characteristics of sliced Khaya laminates conditioned at 65% RH and fatigued in flexure at R ratios of 0.5, 0.3, 0.1, -0.5, -1.

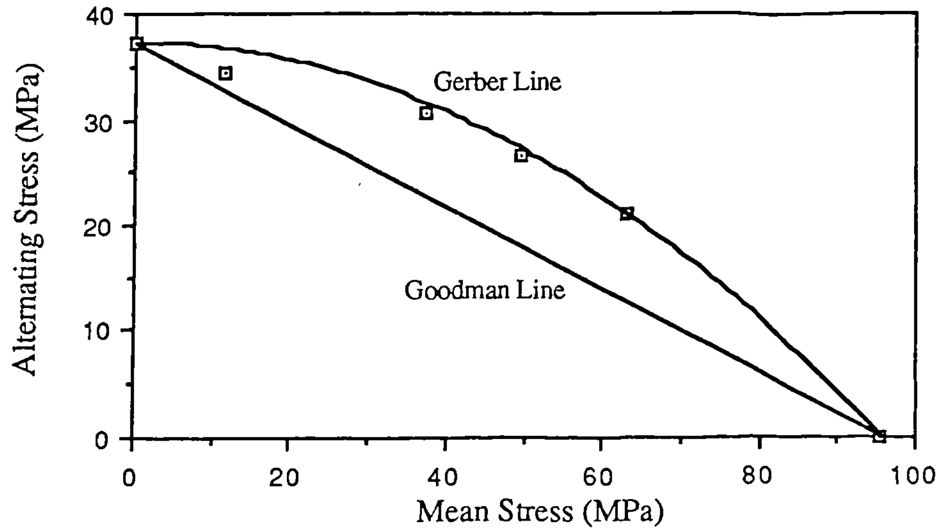


Figure (8.27) Goodman and Gerber constant life lines for sliced Khaya laminates for a life of 10^7 cycles.

A method of representing the fatigue data is by means of a Constant Life diagram or Goodman diagram. Figure (8.27) shows this for a fatigue life of 10^7 cycles. A straight line relating alternating stress σ_{alt}, and mean stress σ_{mean}, is given by the Goodman line

$$\sigma_{alt} = \sigma_e \left[1 - \frac{\sigma_{mean}}{\sigma_{ult}} \right] \qquad .....8.7$$

where, σ_e is the alternating stress at R ratio of -1 and σ_{ult} is the ultimate flexural strength. A better fit of the data may be obtained using a parabolic relationship proposed by Gerber.

$$\sigma_{alt} = \sigma_e \left[1 - \left(\frac{\sigma_{mean}}{\sigma_{ult}} \right)^2 \right] \qquad .....8.8$$

The nonlinear relationship suggests that for negative R ratios, the alternating stress has more influence than the mean stress. This is significant as wood is weaker in compression than tension hence the damage rate in compression would be greater than in tension and as a consequence, with increasing negative R ratios, the compressive component of the fatigue cycle will play an increasing role in damage accumulation.

8.5 Block Loading

Initially, the different block loading programs investigated gave a relatively poor set of results. Early tests with blocks of 10^5 cycles at 65% of static strength followed by 10^4 at 75% of static strength with R = 0.1, were difficult to conduct and were discontinued. The block program was therefore simplified to a two-stage fatigue test. The first stage was at a load level of either 70%, 75% or 80% of static strength with an R ratio of 0.1 lasting 200,000 cycles. This was followed by the second stage with a load level of 85% and an R ratio of 0.5 lasting till failure. The result for this test is seen in figure (8.28) with the x-axis representing the total life of the specimen. The vertical line next to the y-axis represents the 200,000 cycle mark and specimens which did not survive the first stage would fall to the left of the line.The load level of the first block can be seen to affect the life of the second block. Specimens which survived the first block at 80% had a smaller likelihood of surviving longer than those at 70% or 75%.

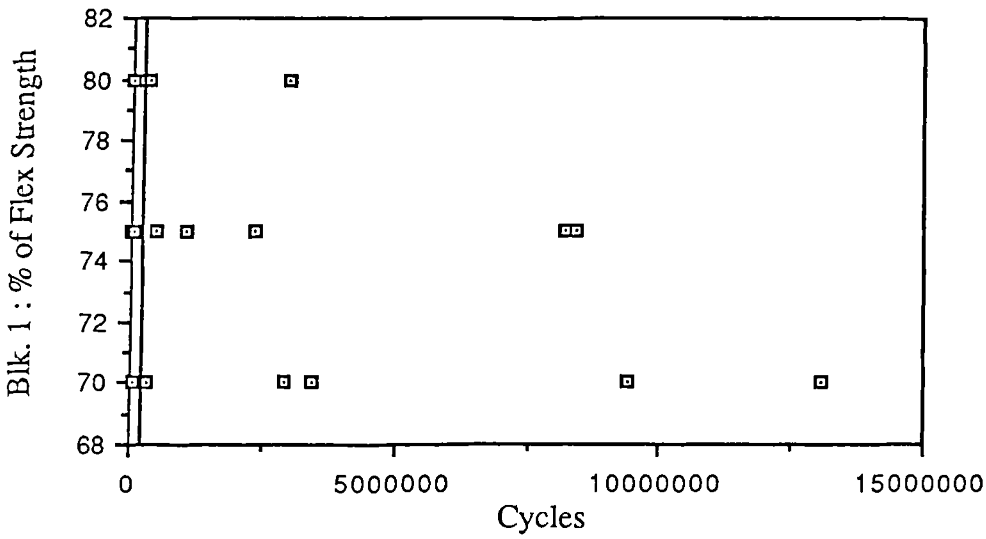


Figure (8.28) The effect of the first stage load level of a two stage fatigue program on the total fatigue life of the specimen.

Based on equation (8.1) for the fatigue curve at R=0.1, the three load levels for the first stage, 70%, 75% and 80%, would have survived for 26.9 * 10^6, 1.55 * 10^6 and 90,000 cycles respectively. Also the fatigue life at 85% of static and R=0.5, according to the regression analysis, is predicted to be very high being of the order of 10^9 cycles. The results show clearly that the life of the second stage is significantly reduced by the first stage. The effect however does not correlate with the severity of the first stage since at 70%, 200,000 cycles corresponds to less than 0.75% of the fatigue life. It should be noted however that for R=0.5, the low gradient combined with a scatter in strength of at least 10%, the fatigue life can be expected to vary from at least 10^5 to 10^{13} cycles. This large scatter would cover the range of results for the 70% and 75% tests although the total number of cycles would still be relatively low.

Another set of results were obtained with a load level of 85% at R=0.5 for the first stage lasting 10^6 cycles, followed by 75% at R=0.1 to failure. Three specimens out of 10 specimens did not complete the first stage although they survived for more than 200,000 cycles. For the seven which did, they survived for an average of a further 760,000 cycles (assuming a log-normal distribution) with a standard deviation of over one decade. The data ranged from 2230 to 15,690,000 cycles. This is within the expected scatter of results for fatigue at R=0.1 hence the effect of the first stage here is negligible. Considering the extremely small damage contribution of the first stage (<0.1%), this is perhaps not surprising. Considered with the previous set of results however, it suggests that there is a strong sequence effect. A low R ratio followed by a high one is more severe. It suggests that the damage contribution at a low R ratio is very high for the initial cycles compared to that of the higher R ratio. This would explain the apparent short fatigue life for the second stage at R=0.5 following fatigue at 70% and 75% with R=0.1.

It can therefore be concluded that there is a clear correlation between the damage due to the first stage of the load program and the subsequent fatigue life in the second stage. Also, there is a strong suggestion of sequence effects because the fatigue life in the second stage can be reduced more greatly by the first stage than estimated from Miner's rule calculations.

Chapter 9

Changes In Fatigue Deflections And Dynamic Modulus

9.1 Introduction

Fatigue deflections are defined as the changes in the maximum and minimum deflections during a fatigue test. They are similar to creep deflections but instead of being time dependent, they are cycle dependent. A typical curve for fatigue deflection is shown in figure (9.1) for rotary cut Khaya tested at an R-ratio of 0.1. The test was in flexure and therefore since the load was applied in the compressive direction, the deflection was increasing in the negative direction. This convention was used throughout the tests. The upper curve therefore represents the change in the minimum deflection peak while the lower represents the maximum deflection peak corresponding to the minimum and maximum applied loads. The true deflection of the specimen would be oscillating between the minimum and maximum peaks.

The form of the curves are very similar to that found for creep tests. It can be similarly considered as having three stages: a primary, secondary and tertiary stage. The primary stage shows a rapid increase in levels of strain followed by a secondary stage where the increase is at a constant rate. The final failure is preceded by an rapid

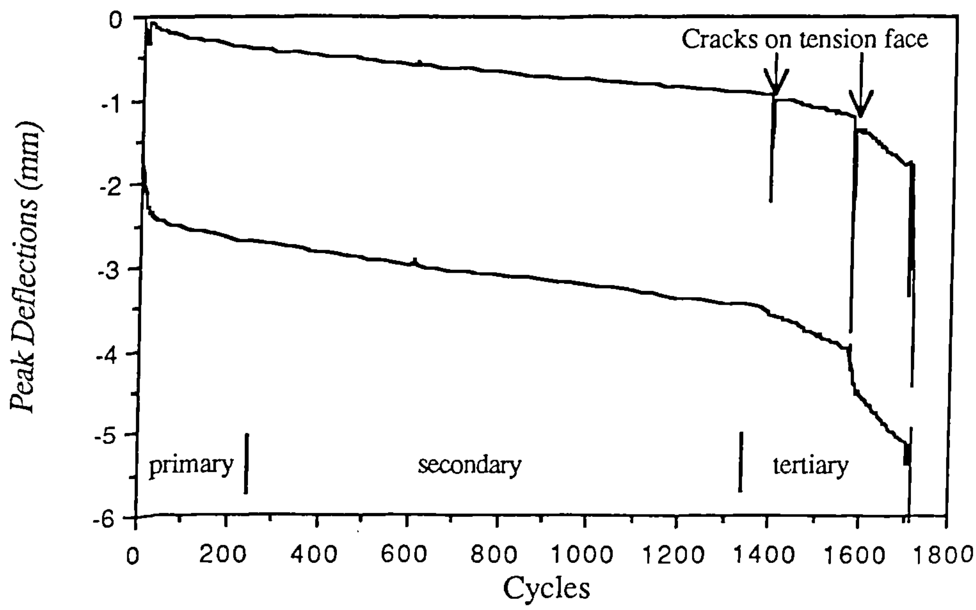


Figure (9.1). The minimum and maximum fatigue deflection curves for a rotary cut Khaya specimen at 78% of flexural strength.

increase in the strains. As in the creep of wood, the extent of each stage is variable and the tertiary stage may not be found before catastrophic failure. The specimen in figure (9.1) illustrates the development of failure of some specimens in the tertiary stage with cracks developing in the tension face prior to final catastrophic failure. The transition between the stages is however not definite. Each stage is not distinguishable as a physical phenomenon. They are used simply to separate the trends and therefore the transition from one stage to another is determined quite arbitrarily depending on the model used for each stage of the curve.

The dynamic modulus is based on the peak strains measured and the peak stresses applied to the specimen. Equation 7.5 was used to calculate the modulus. This will not be expected to be equivalent to the static modulus as measured in standard tests. This is due to rate effects, since the fatigue tests were conducted at stress rates of between 500-1000 MPa/sec while standard flexural tests recommend stress rates of about 6 MPa/sec (a constant strain rate of 0.0015 m/m/min is usually specified). The higher stress rates would result in a higher modulus. A typical effect of fatigue cycles on the dynamic modulus is shown in figure (9.2), based on the same specimen as

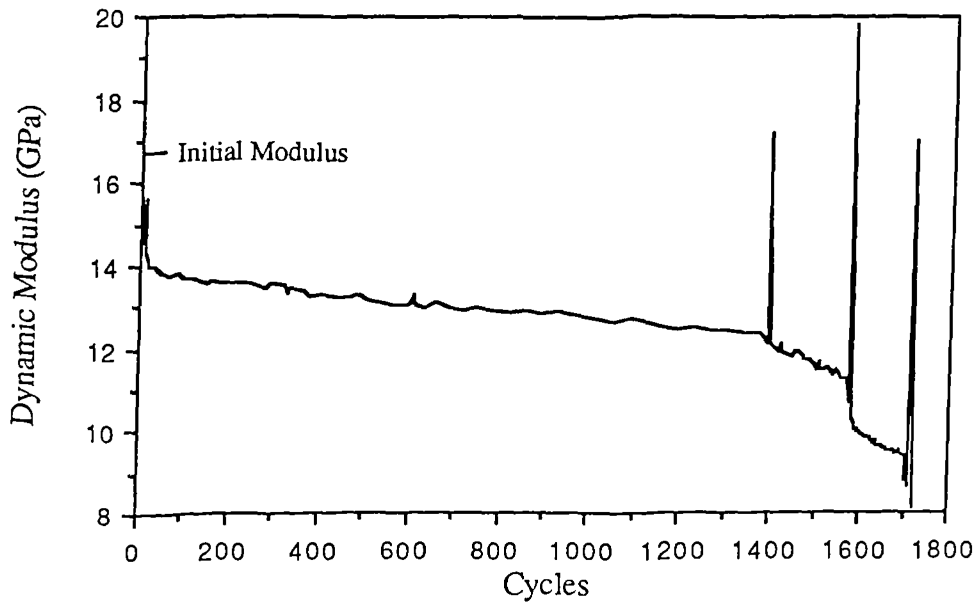


Figure (9.2). The change in dynamic modulus with fatigue cycles for a rotary cut Khaya specimen at 78% of flexural strength.

above. Again, a trend similar to the three stages of fatigue deflection is found. The modulus decreases rapidly in the primary stage followed by an approximately linear secondary stage prior to the final tertiary stage and failure.

In a fatigue test where the R-ratio is positive, ie. no stress reversals, and the minimum load is close to zero, the changes in the minimum and maximum deflections largely reflects the delayed elastic and nonrecoverable components of strain in the specimen (see figure (9.1). The difference between the minimum and maximum deflections is due to the elastic component of strain which reflects the modulus of elasticity. Therefore, with positive R-ratios, it would be easier to consider just the minimum fatigue deflection to follow the delayed elastic and nonrecoverable components. The change in modulus may then be used to reflect changes in the elastic contribution. In so doing, two means of monitoring the changes in the specimen during fatigue become available. However, if the minimum load is not close to zero (such as for an R-ratio of 0.5), then the change in modulus would also contribute to some of the changes in the minimum deflection although this would be small. The higher average stress the specimen is at would result in a smaller dynamic contribution and increased time dependent effect.

If stress reversals occur, ie. at negative R-ratios, the maximum and minimum peak strains may not change in the same direction. Where the mean stress is close to zero, there would be only a small component of delayed elastic and nonrecoverable strain. The use of fatigue deflections would not therefore be directly relevant. In fact with R = -1, any changes in the fatigue deflection would be more a result of changes in modulus.

9.2 Analysis of Fatigue Deflections

9.2.1 Effect of Load Level at R-ratio of 0.1

A series of fatigue tests for a wide range of load levels at R = 0.1 was conducted for Sitka spruce and sliced Khaya laminates. The result to 10,000 cycles is shown in figure (9.3) for Sitka spruce and figure (9.4) for sliced Khaya. It can be seen

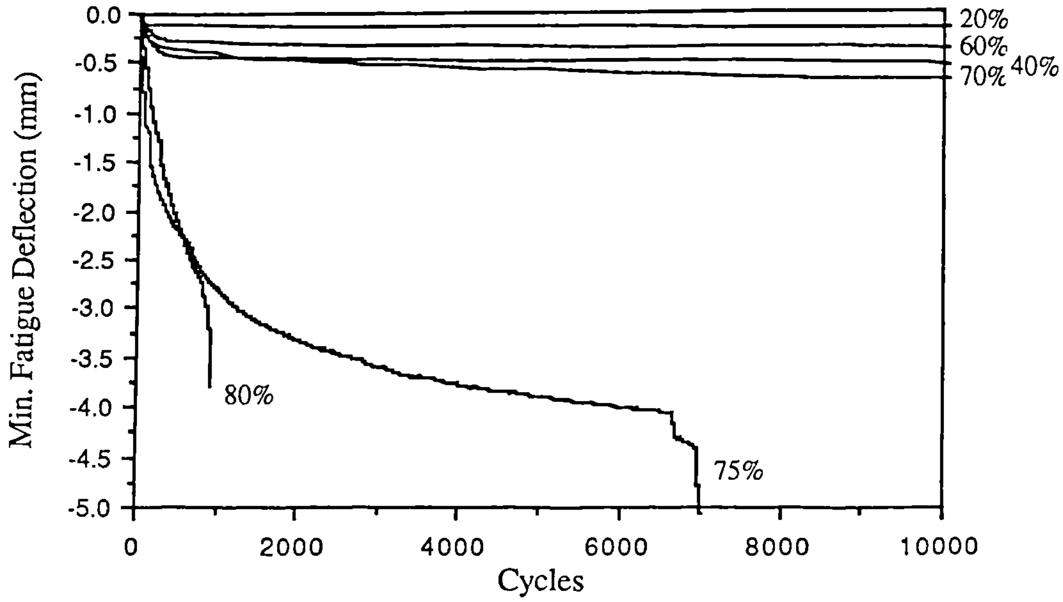


Figure (9.3). Fatigue deflection curves for Sitka spruce fatigued at different levels of load expressed as a percentage of static flexural strength.

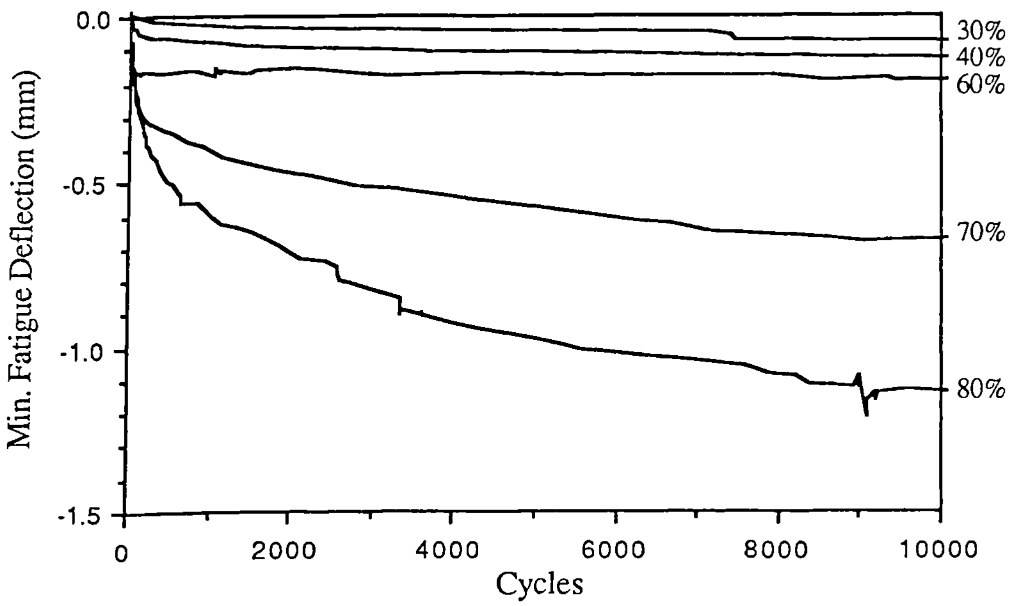


Figure (9.4). Fatigue deflection curves for sliced Khaya laminates fatigued at different levels of load expressed as a percentage of static strength.

that at higher load levels, the amount of fatigue deflection is much higher. In figure (9.3), the specimens at 75% and 80% failed within 10,000 cycles and they show a very large increase in fatigue deflection. The specimens at 70% and lower however all survived the fatigue test beyond 1 million cycles. The large difference in fatigue deflections is probably reflected by the difference in fatigue life. It can also be seen that the fatigue deflection at 40% was in fact marginally greater than that at 60%. This could be due to the scatter in static strengths with the consequence that the quoted load levels are wrong to the extent that the true applied load level are actually similar.

The variability in the specimens may also be considered as variability in fatigue lives. Figure (9.5) shows the fatigue deflections of four Sitka spruce specimens fatigued at a nominal load level of 80% of static flexural strength. The result shows an increased rate of change in the deflection for specimens with a lower fatigue life. The final deflection close to failure is however variable although 4mm appears to be typical for most tests. Therefore, any direct prediction of fatigue life from the rate of change of deflections would not be possible but a slow rate of change in deflection is a strong indicator of a longer life specimen.

All the data collected for Sitka spruce to 100,000 cycles is shown in figure (9.6) as a plot of the change in deflection at a cycle level over a wide range of load levels. The data points are plotted in different symbols for 100, 1000, 10000 and 100000

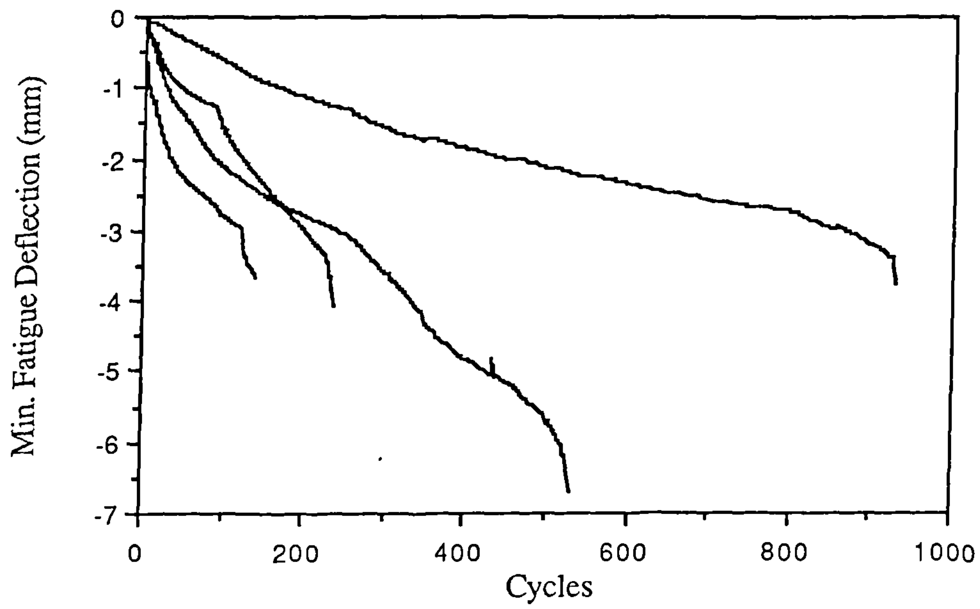


Figure (9.5) Fatigue deflection curves for four Sitka spruce specimens fatigued at 80% of static flexural strength.

cycles. For the results that are above 65% of the flexural strength, a rapid rise in the fatigue deflection occurs. This is significant as it corresponds to the change in creep compliance with load as seen in figure (3.8). Figure (9.7) shows the result for sliced Khaya plotted in a similar way. A similar transition or knee is found in the graph at

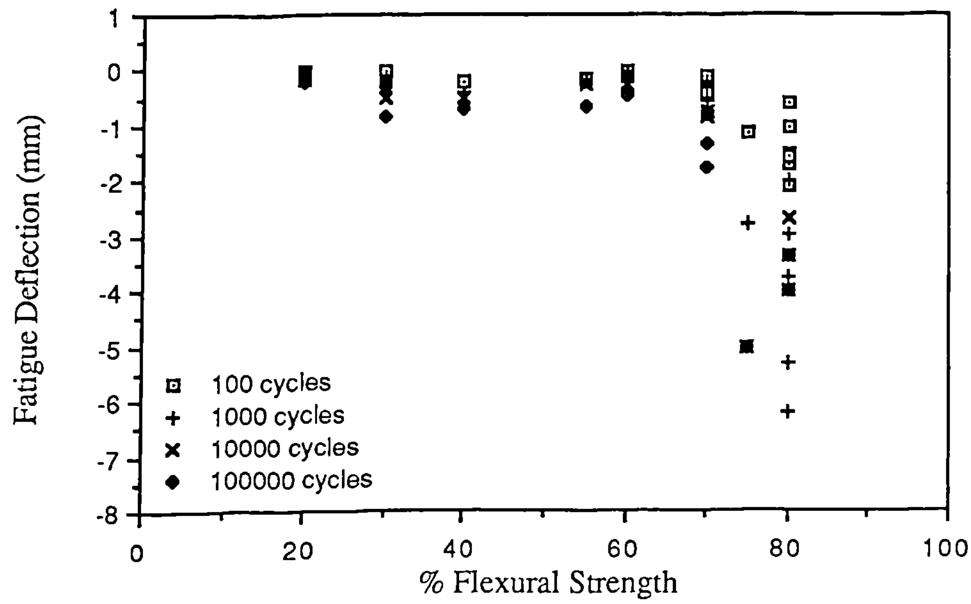


Figure (9.6) The effect of load level on the fatigue deflection at 100, 1000, 10000 and 100000 cycles of Sitka spruce specimens.

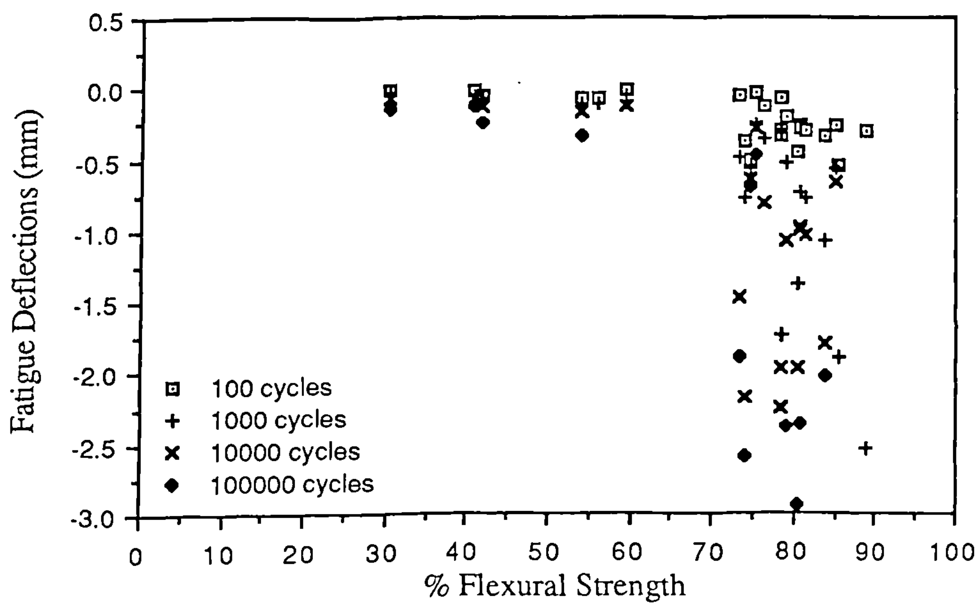


Figure (9.7) The effect of load level on the fatigue deflection at 100, 1000, 10000 and 100000 cycles of sliced Khaya specimens.

around 65% of the flexural strength. Attempts to curve fit this result were made using an inverse function which may be expressed as

$$\frac{1}{F_d} = k + cS \qquad .....9.1$$

where F_d is the fatigue deflection and S the percentage load or stress level. The constants, k and c was determined using linear regression. The results and the correlation coefficient, R is shown in table (9.1). The curves show a very good fit for Khaya especially at higher fatigue cycles. For clarity, the curves for Khaya at 100 and 100,000 cycles are plotted with the data in figure (9.8). It indicates a small shift to lower load levels of the 'knee' at 100,000 cycles. There are however difficulties in this analysis. The asymptote for the vertical axis appears to occurs at approximately 81% of flexural strength for all fatigue cycles. This cannot be valid and is simply due to the nature of the data, being available only to 80% of flexural strength. Since the fatigue deflection is greater at lower loads for 100,000 cycles, the curve fit is more reliable. Further, the data is more sensitive at higher load levels to the variability in strength. However, the very good fit for 100,000 cycles does suggest that the inverse relation is a reasonable model for the data.

Table (9.1). Constants and Correlation coefficients of equation 9.1 fitted to the data in figures (9.6) and (9.7).

Material	Cycles	k	c	R
Khaya	100	-94.74	1.144	0.79
	1000	-42.46	0.529	0.88
	10000	-21.12	0.260	0.91
	100000	-11.74	0.145	0.94
Sitka	100	-57.24	0.733	0.57
	1000	-28.86	0.373	0.59
	10000	-13.06	0.165	0.63
	100000	-5.197	0.063	0.81

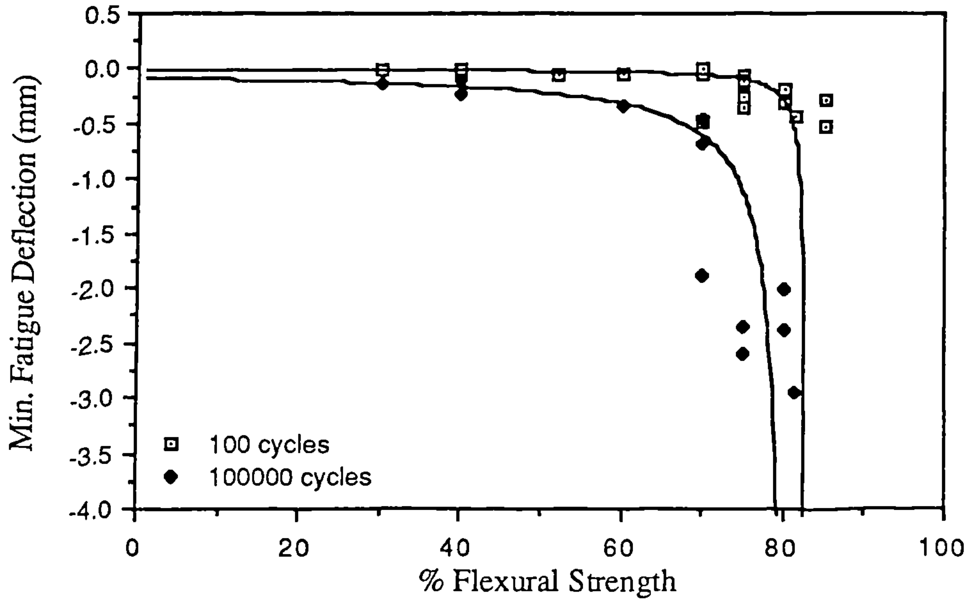


Figure (9.8) Data for sliced Khaya specimens at 100 and 100000 cycles plotted with the fitted curve based on equation (9.1) and coefficients in table (9.1).

9.2.2 Effect of R-ratio

The effect of R-ratio was considered by conducting another series of tests with Sitka spruce and sliced Khaya at an R-ratio of -1. This provided a totally different loading regime compared with R=0.1. As elaborated in the introduction, at R=-1, the zero mean stress would imply that with every cycle, full recovery of the delayed elastic and nonrecoverable strain would occur. Figure (9.9) and (9.10) shows this for Sitka spruce and sliced Khaya respectively with the deflections changing in both directions. The deflections here were plotted as absolute deflections of the specimens rather than fatigue deflections as in earlier plots of R=0.1 data. The deflections appear to change rapidly during the initial cycles followed by a steady increase before the final rapid rise following the three primary, secondary, and tertiary stages found with R=0.1. This can be clearly seen in both plots at the higher load levels. Both the maximum and minimum peaks appear to change in a similar but opposite direction suggesting these changes may be better considered as changes in modulus (see Section 9.3.2). Not all results however gave a symmetrical change in maximum and minimum deflections with cycles. Some specimens appear to sustain greater damage to one face with the effect that most of the change occurred to only one of the peaks. This can be seen occurring

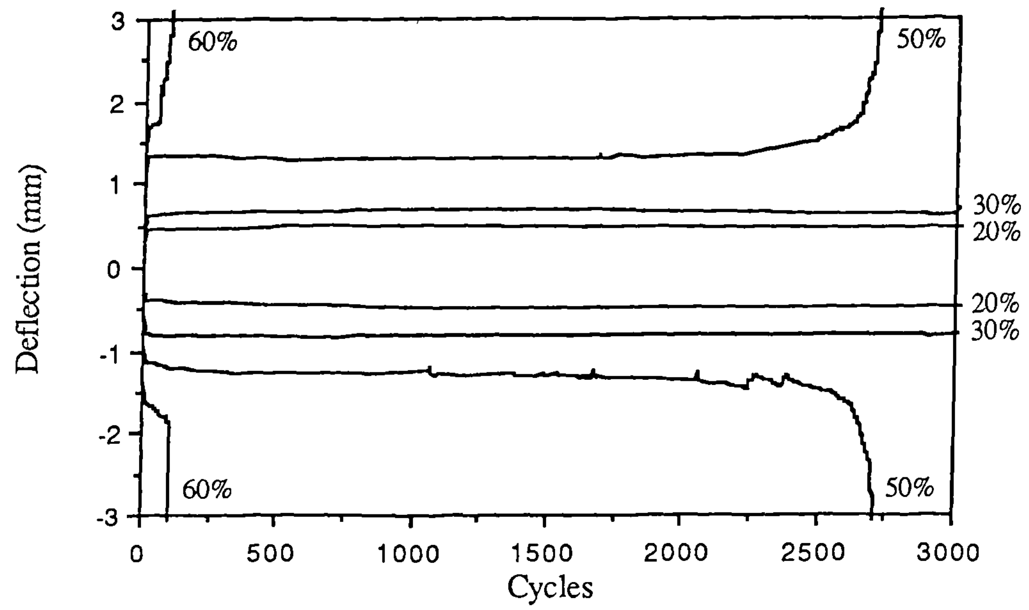


Figure (9.9) Changes in peak deflections of Sitka spruce fatigued at R=-1 and at various load levels.

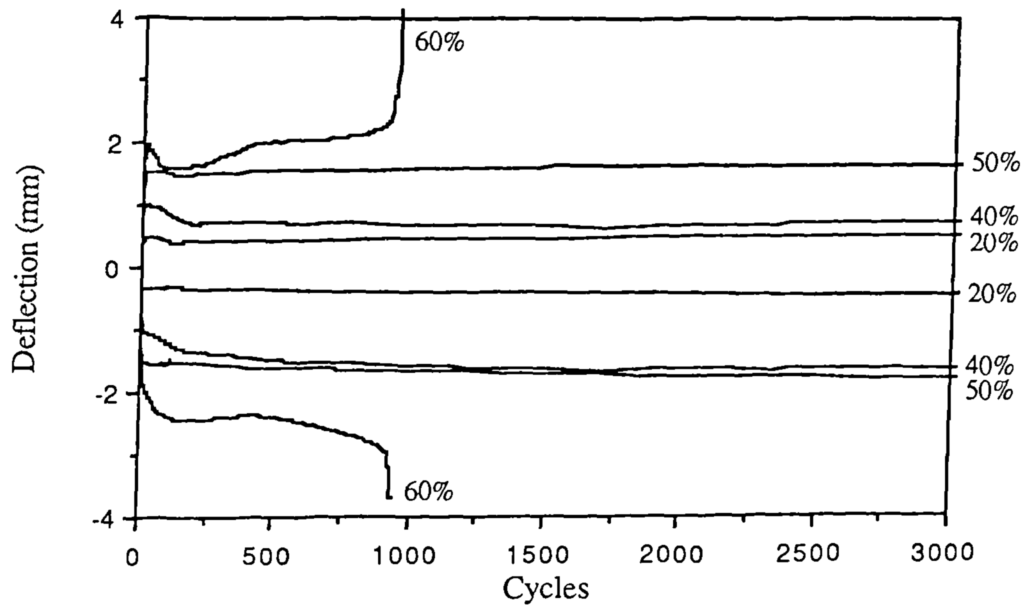


Figure (9.10) Changes in peak deflections of sliced Khaya fatigued at R=-1 and at various load levels.

in figure (9.10) for sliced Khaya at 40% of static strength. The initial cycles resulted in a net decrease in both deflection peaks before stabilizing and proceeding in the manner similar to other specimens.

The load level also greatly affected the changes in deflection. At higher loads, the change is greater as seen in figures (9.9) and (9.10). It is interesting to note that like the R=0.1 results, at the lower load levels, the difference in the change in deflections are all very small. Unfortunately, insufficient reliable data was available to examine if a similar 'knee' was present. The available data however indicate this though not conclusively. Better results were obtained from the modulus measurements and these are examined in Section 9.3.2.

9.3 Analysis of Dynamic Modulus

9.3.1 Effect of Load Level at R-ratio of 0.1

The changes in modulus were studied in the similar manner to that of the deflections. Its relevance however is different as the modulus reflects the elastic response of the wood structure. Any damage such as cell wall buckling, cracking and a

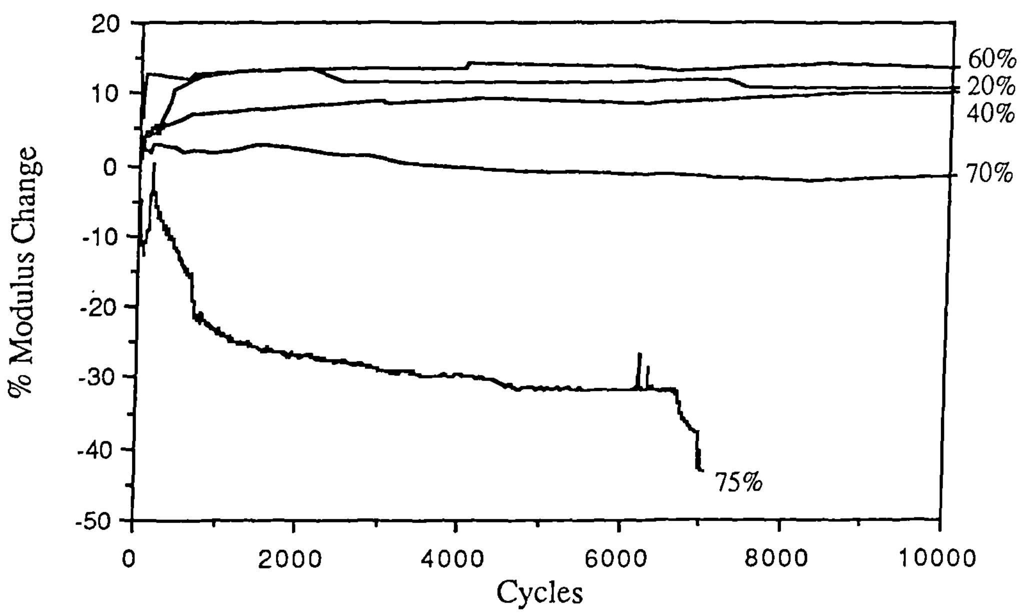


Figure (9.11) The changes in dynamic modulus with cycles for Sitka spruce at different percentage load levels.

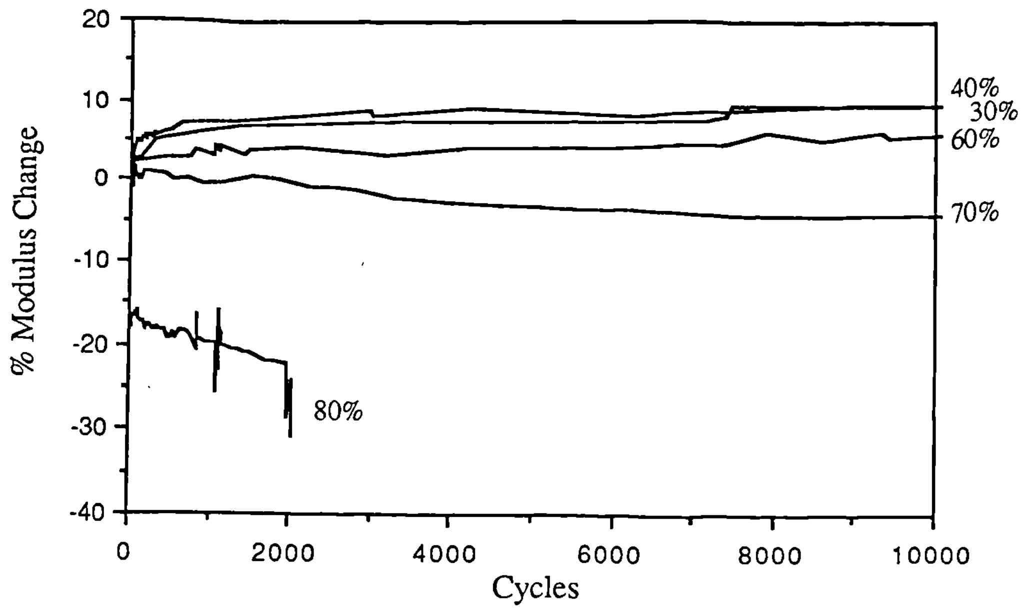


Figure (9.12) The changes in dynamic modulus with cycles for sliced Khaya at different percentage load levels.

general redistribution of stresses in the specimen would alter the elastic response. Figure (9.11) and (9.12) compare the changes in dynamic modulus at different load levels for Sitka spruce and sliced Khaya respectively. Two features stand out in the results. The first is that the modulus changes to a different extent for high and low load levels. This is as expected since the higher load levels would cause greater damage and cell wall buckling in the compression face.

The second rather more startling result is the increase in modulus with cycles for low load level specimen and a decrease in modulus at higher loads. Both for Sitka and Khaya, the results show the greatest increase in modulus at the lowest load level with a smaller increase up to 60%. At 70%, the specimens showed both changes with a small increase before a decrease. Indeed, only in the much higher load levels was the characteristic three stages described in Section 9.1 found. However, at the lower loads, the form of the increase followed the characteristic primary, then secondary stages of change. It was not possible to ascertain if at the low loads, the increase in modulus would be followed by a decrease at much higher fatigue cycles. If fatigue failure was to occur, this downturn would be expected to occur in a similar way to the 70% results.

Plotting the percentage change in dynamic modulus against the fatigue load level at different fatigue cycle level, the transition point between an increase in modulus and a decrease can be more clearly seen. Figure (9.13) and (9.14) show the plots for Sitka spruce and sliced Khaya respectively with the load level expressed as a percentage of

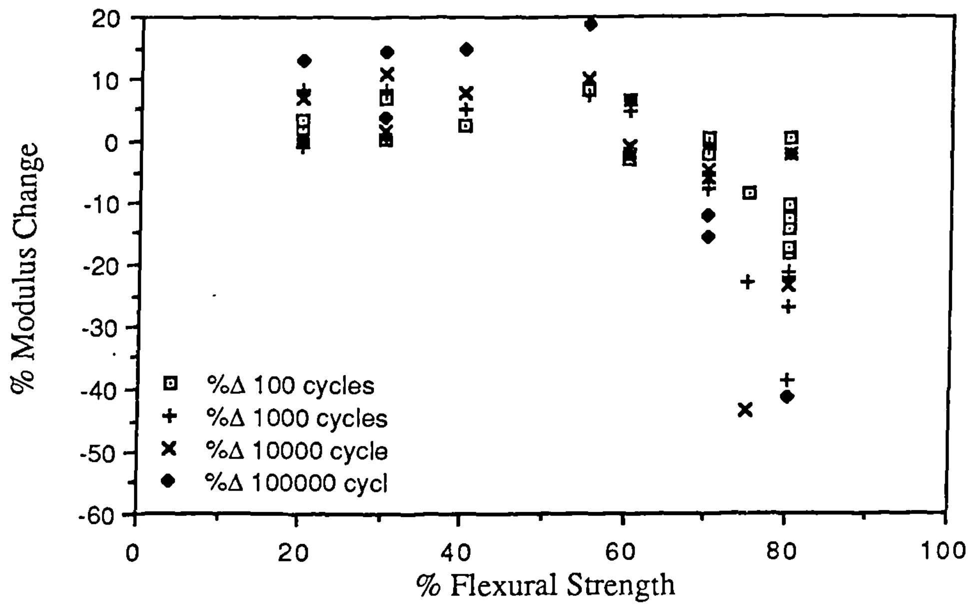


Figure (9.13) The effect of load level on the percentage change in the dynamic modulus of Sitka spruce at 100, 1000, 10000, 100000 cycles.

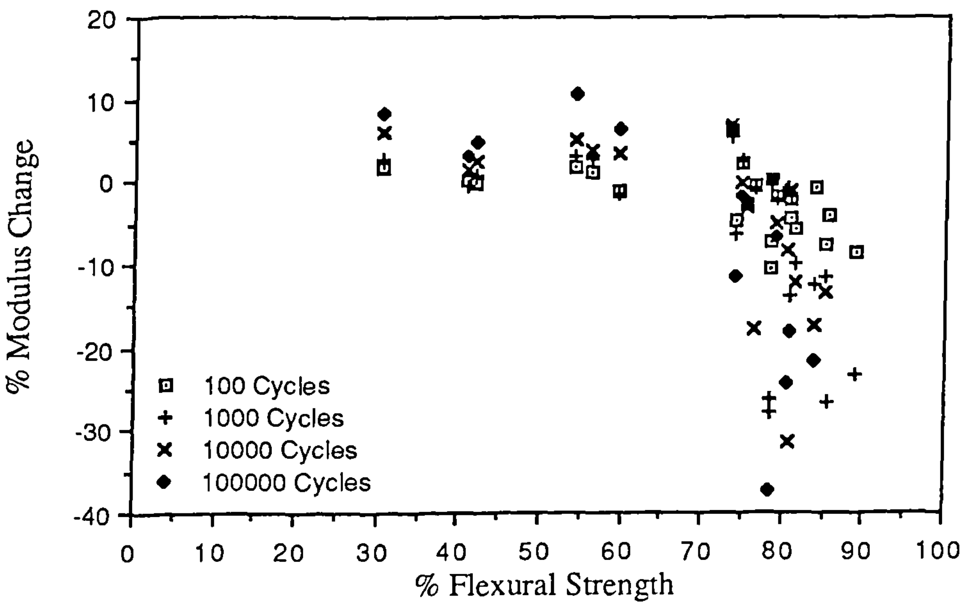


Figure (9.14) The effect of load level on the percentage change in the dynamic modulus of sliced Khaya at 100, 1000, 10000, 100000 cycles.

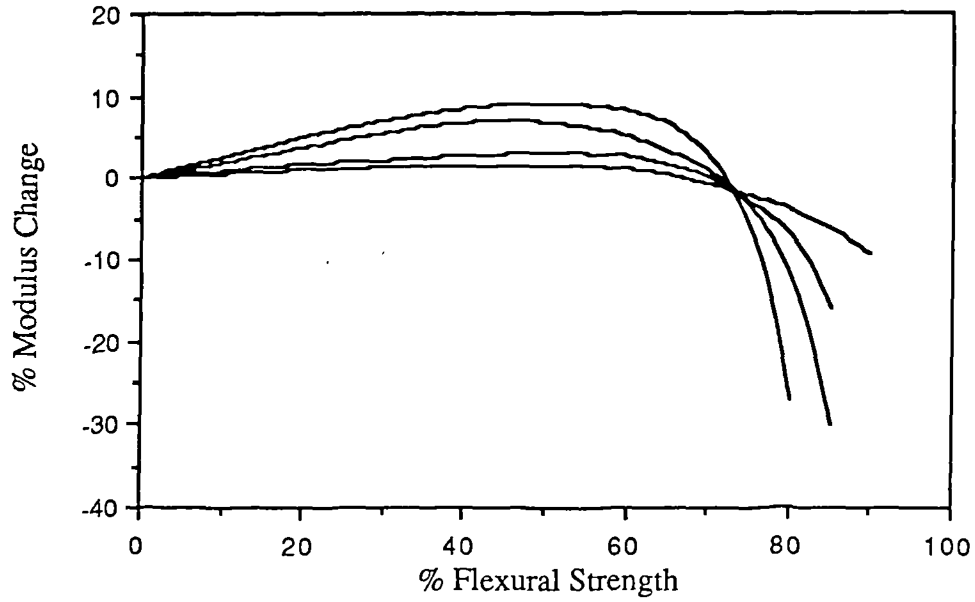


Figure (9.15) Visually fitted lines based on data of sliced Khaya in figure (9.14).

static flexural strength. Despite the scatter in results, the data shows a net increase in modulus up to around 70-75% of static strength. Beyond 75%, the plots shows a vary rapid decrease in modulus. Figure (9.15) shows visually fitted lines through the sliced Khaya data at the various cycle levels. The figure also suggests that around the transition point, the modulus will increase before at higher cycles, decreasing. Although the cycle level is a decade apart, the change in the transition point with cycle is small compared with the intercept with the x-axis. Also the fall in modulus above the transition point is much more rapid at higher fatigue cycles. These observations suggest strongly that the transition point at around 70-75% of the flexural strength is also a transition for the fatigue damage mechanism. Also the increasing steepness of the fall above the transition suggests that below the transition point, fatigue life can be very high and possibly a fatigue limit may exist. Confirmation of this however is required with tests to 10^6 and 10^7 cycles but it may be experimentally very difficult to obtain. The scatter in results is also a major problem which greatly affects the accuracy in determining the changes in the transition point.

9.3.2 Effect of R-ratio

A very different characteristic was found in the change in dynamic modulus when the R-ratio is -1. As noted in section 9.2.2, the opposite movement in the peak deflections must imply that a more rapid decrease in dynamic modulus must occur with cycles. Indeed, unlike for R=0.1, no increase in modulus was found. Figure (9.16) and (9.17) are plots showing the effect of fatigue cycles on modulus at different load levels for Sitka spruce and sliced Khaya respectively. The plots indicate the similarity in the characteristics at low load levels but the decrease in modulus is rapid at higher load levels. No major difference in the characteristics of Sitka and Khaya were observed although Khaya appears to be less affected at the same load level.

The quality of the results at R=-1 was relatively poor due to the difficulties of the test as discussed in chapter 7. Figure (9.18) shows the modulus changes for sliced Khaya up to 10000 cycles at various load levels. Despite the scatter, it appears that no definite "knee" exists as for tests at R=0.1. The modulus decreased steadily with load level with the rate of decrease greater at the higher cycle level. This may be because unlike the tests at R=0.1, where much of the modulus decrease was associated with the

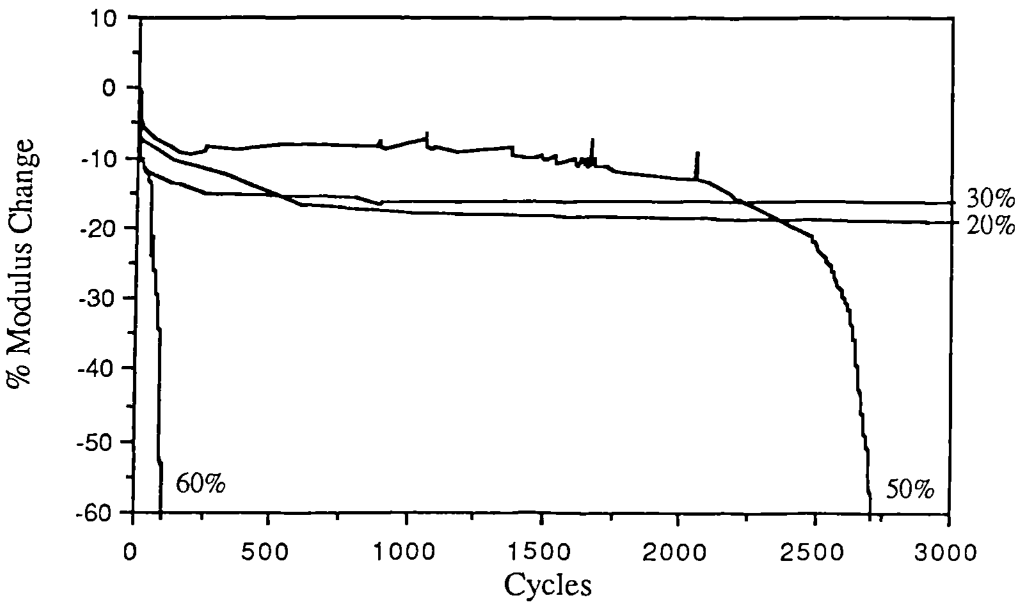


Figure (9.16) Effect of fatigue cycles on dynamic modulus of Sitka spruce at different load levels and a R-ratio of -1.

tertiary stage, a substantial decrease occurred during the primary and secondary stages. It suggests that when under compression, structural damage such as cell wall buckling occurs which, although not extensive at low stresses, reduces the tensile modulus as the stress is reversed.

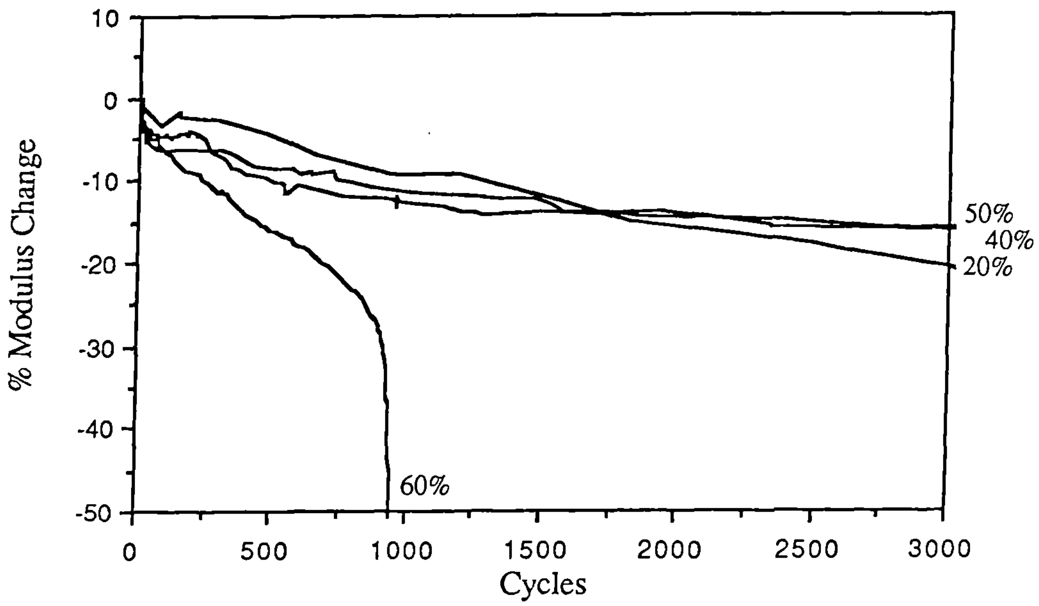


Figure (9.17) Effect of fatigue cycles on dynamic modulus of sliced Khaya at different load levels and a R-ratio of -1.

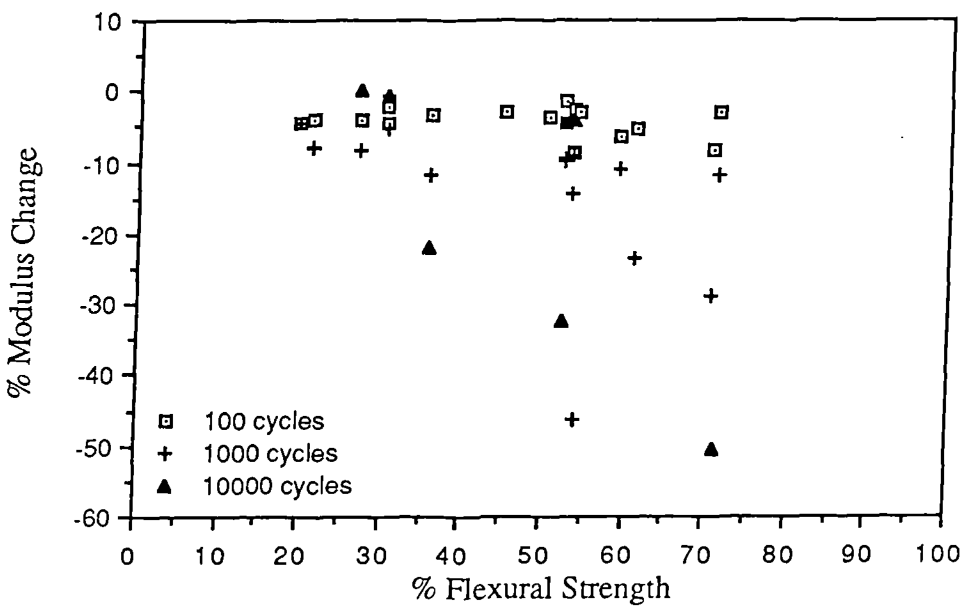


Figure (9.18) Effect of load level on the dynamic modulus of sliced Khaya fatigued at R=-1.

9.4 Miscellaneous

9.4.1 Block Loading

A few tests were conducted using the block loading capability of SArGen. Initial tests suffered from problems of overshoot as the load levels changed from one level to another. This was clearly unacceptable and modifications to the program were made to eliminate this. A successful result was achieved based on a loading program with a maximum stress of 75% of flexural strength and at two R-ratios, 0.1 and 0.5. The peak deflections are shown in two parts in figures (9.19a and b). Failure of this sample occurred at just over 200,000 cycles. As expected with the same maximum load levels, the maximum deflection showed a continuity while the minimum peak changed between the two R-ratios. In the initial primary stage, most of the change in deflection appear to occur during the larger amplitude block. However during the secondary stage, it is noticed that the changes tended to occur in the R=0.5 block immediately following the R=0.1 block. In fact, the small tertiary stage and final failure occurred in a similar manner following the fatigue cycles at R=0.1.

The changes in dynamic modulus surprisingly showed a large difference in value between the two R-ratios. Figure (9.20) shows the change in modulus with cycles. At R=0.5, the modulus appeared to be significantly higher. It increased after the initial block before showing the usual decrease in modulus. It appears that the difference in modulus at the two R-ratios was systematic and may be due to the different parts of the load-deflection curves in the cycles. This however is inconsistent with expectations since at the upper half of the load-deflection curve would show a decrease rather than the increase in modulus found. This is seen in figure (9.21) with the different moduli measured and the expected load-deflection curve from which a decrease in modulus might be expected. In figure (9.20), the maximum peaks were continuous which imply that the difference in modulus is due to the minimum peak being not coincident with the load-deflection line for R=0.1. In fact, extrapolating the minimum deflection peak for R=0.5 to zero load, a positive residual deflection of about

0.5mm remain. Apart from being an experimental error (which may be discounted since errors are more likely increase the modulus at R=0.1 rather than R=0.5) no explaination for this unexpected behavior could be found.

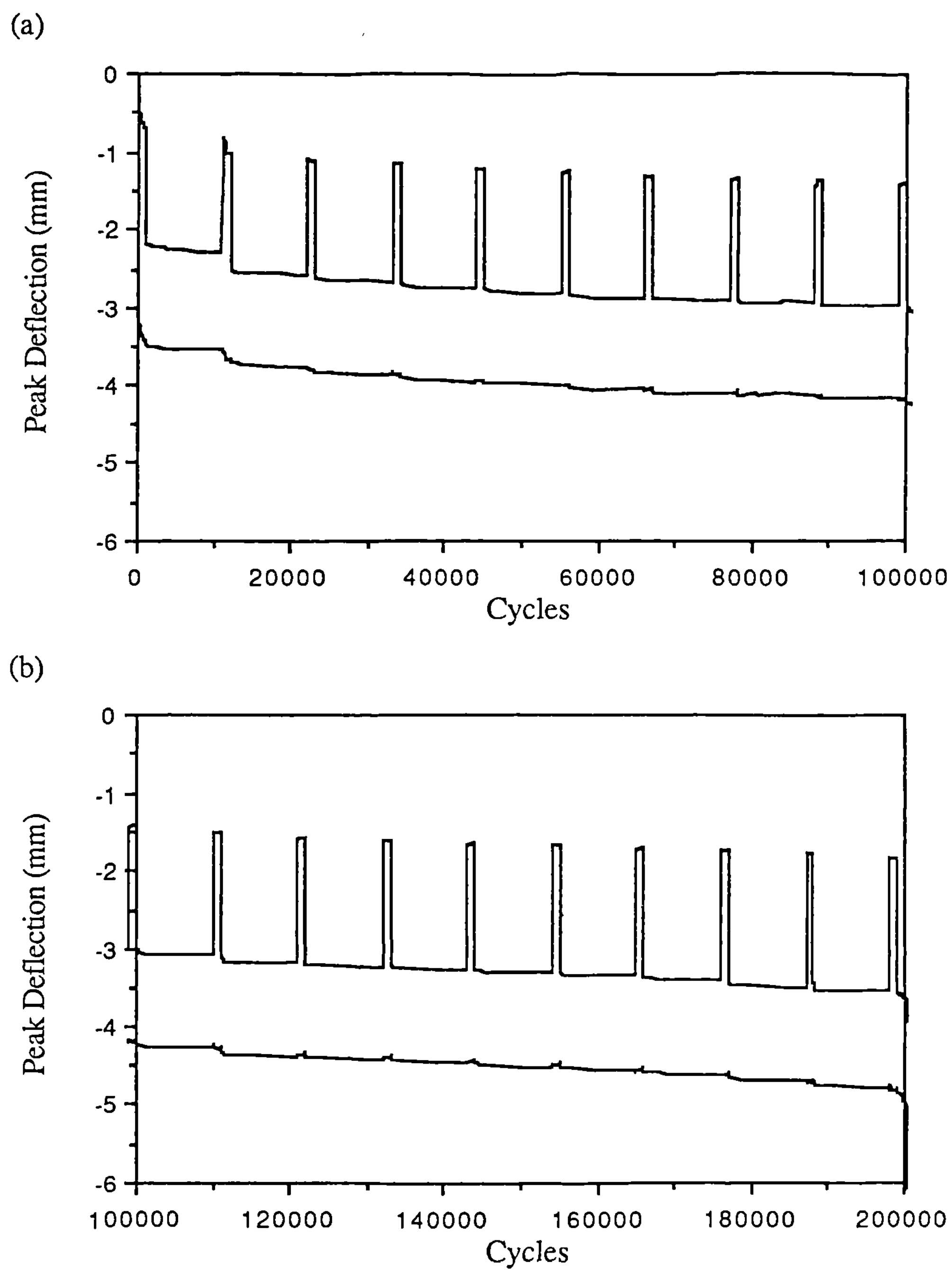


Figure (9.19) The changes in fatigue deflections of sliced Khaya subjected to a block loading program at 75% of flexural strength and at R-ratios of 0.1 (1000 cycles) and 0.5 (10,000 cycles). (a) Fatigue cycles between 0 and 100,000 cycles. (b) Fatigue cycles between 100,000 and 200,000 cycles.

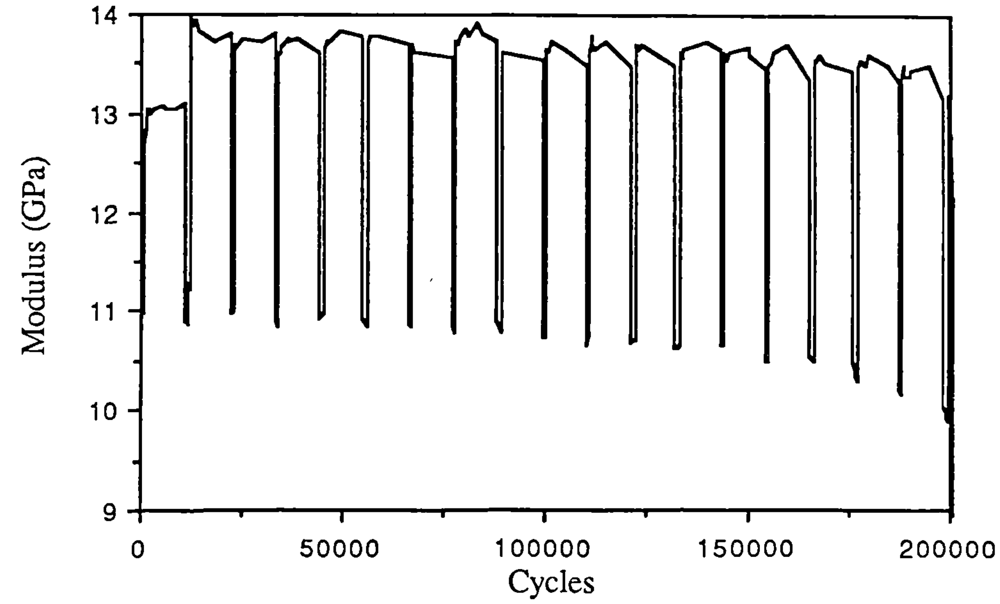


Figure (9.20) The variation of the dynamic modulus of sliced Khaya subjected to a block loading program at 75% of flexural strength and at R-ratios of 0.1 (1000 cycles) and 0.5 (10,000 cycles).

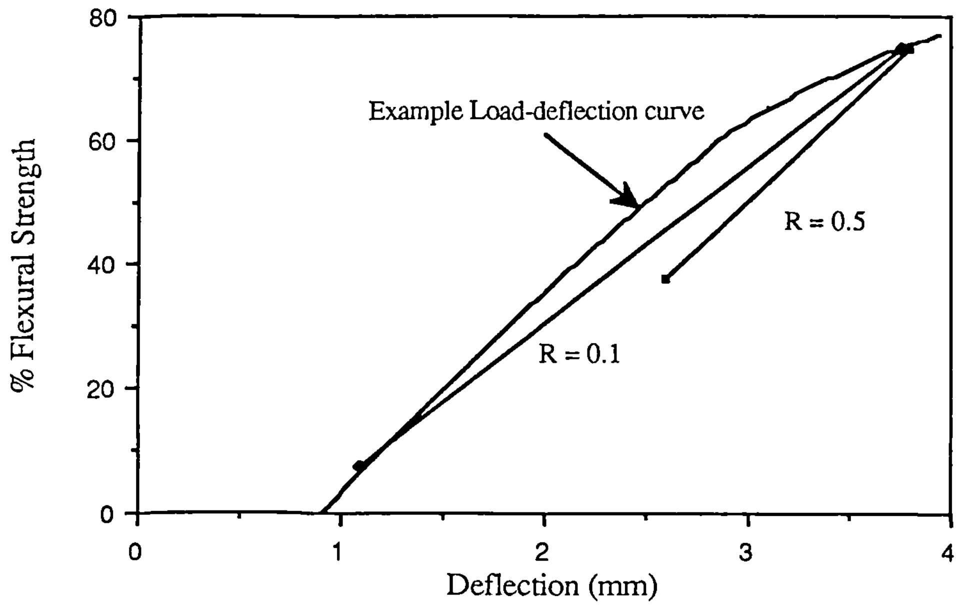


Figure (9.21) Comparison of the measured dynamic modulus at R=0.1 and R=0.5 with an example load deflection curve.

9.4.2 Comparison of Species

The results for Sitka spruce and sliced Khaya in the figures above are now compared since the two woods are structurally very different. However, as the plots show, the fatigue deflections and changes in dynamic modulus are not significantly different between the two species. Indeed the results on rotary cut Khaya were also very similar as seen in figure (9.1) and (9.2). It reinforces the view that different wood species are distinct largely because of the amount of wood tissue present. Other factors do cause some difference but they are not as significant. This may be seen in some difference found in the relative amount of change in modulus (figures (9.13) and (9.14)) although this is difficult to compare due to the different load levels. However, in general, by comparing figure (9.7) and (9.8), the fatigue deflections of Sitka spruce do appear higher. This may be because of the more brittle nature of sliced Khaya laminates with their interfacial resin bonds. The knee however does not appear to be significantly higher. Comparing the changes in modulus, Sitka spruce appears to be affected more greatly. The knee in the result also appears to be slightly lower for Sitka spruce occurring at around 60% rather 70% for sliced Khaya. Differences appear to coincide with the difference in compression strengths. As seen in table (8.1), Sitka has a much lower compressive strength to flexural strength ratio. With much of the damage appearing to occur in the compression face a higher amount of damage would occur at the same percentage load on the compression face with Sitka spruce. This damage would be reflected in the fatigue deflections.

Chapter 10

Macroscopic and Microscopic Study of Fatigue Failure

10.1 Visual Observation of Fatigue Damage

Specimens were observed during fatigue tests for any visual indications of damage or any signs of imminent failure. In general only two types of damage were observed, longitudinal cracks on the tensile face and compression creases. The sequence of damage observed on sliced Khaya at R=0.1 was as follows.

(a) Small compression creases were usually the first signs observed. They occasionally occur above the loading points despite the polyethylene pads. A crease can be seen above the left roller in figure (10.1).

(b) Towards the later half of the specimen's life, a small crack would appear on the tensile surface. For specimens with a short fatigue life (<1,000 cycles), being subjected to a high load, the crack often appears very late and it very quickly develops within a few cycles to failure. Also cracks are very small at low loads but much larger at high loads.

(c) Once the crack is formed it grows along the length of the specimen to form a large splinter. Occasionally, an initial crack might form very early on at the top edge of

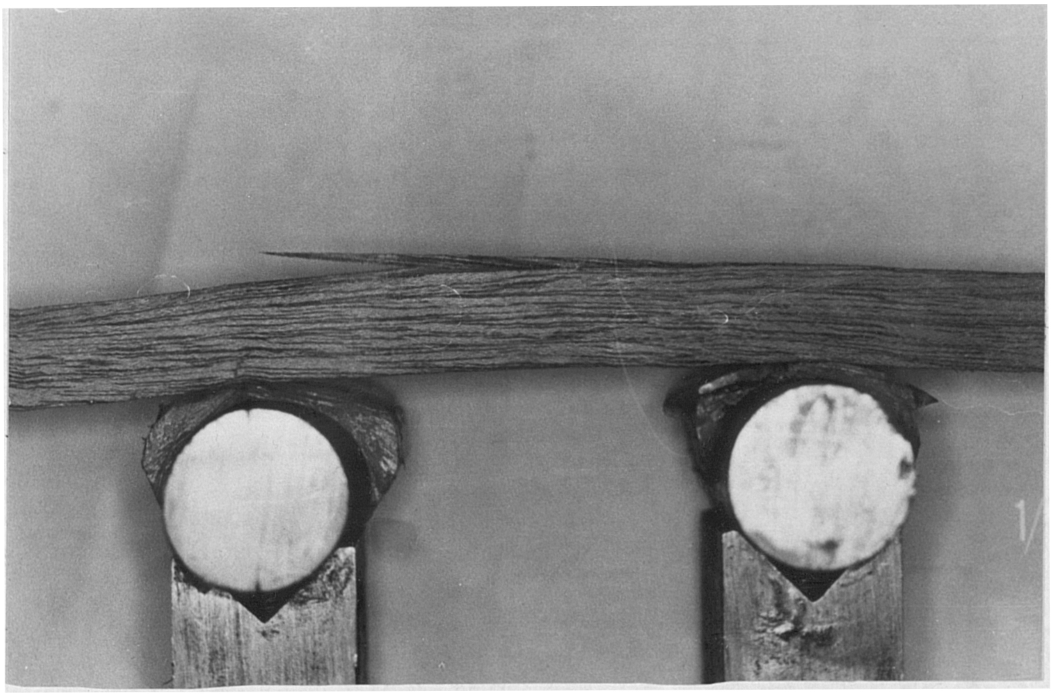


Figure (10.1) Photo of tensile cracks of a sliced Khaya specimen at 3 million cycles.

the specimen but does not prove serious and another crack forms instead and grows. Figure (10.1), taken at 3 million cycles, shows a tensile crack which developed at around 2 million cycle.

(d) There is normally very little indication that failure is imminent. A substantial crack may be present for over a third of the life of the specimen before failure is catastrophic.

Sitka spruce shows a similar behavior although the crack appears to form preferentially along the latewood/earlywood interface. Where an interface is present at the tensile surface, cracks would initiate there. These cracks also do not appear to have a significant effect on the modulus of the specimen if the load level is <70% of the ultimate strength. Compression creases also appear to be more extensive with Sitka specimens compared with slice Khaya.

With stress reversals, the cracks formed in the specimens are very different from those of the unreversed specimens. There is usually a blunt tip to the splinters indicating tensile failures had occurred where compression creases had formed earlier. Also, the formation of a crack rapidly leads to other cracks forming on both sides of the specimen leading to failure.

10.2 Electron Microscopy

The electron microscope has been extensively used by Dobraszcyk (1983) to examine the fracture surfaces of fatigued specimens of Douglas fir. Boatright (1977) studied fatigue crack propagation in the RL and TL planes. Most of the fracture types found were similar to that found with static fracture. A brief study was also made in this work on Sitka spruce and sliced Khaya.

Figure (10.2) to (10.4) shows the fatigue fracture surface from the middle of a Sitka spruce specimen. Figure (10.2) examines the failure surface covering the earlywood and latewood sections. The earlywood with its thin cell walls appears to have failed cleanly in a brittle fashion. The latewood has a more fibrous surface but is

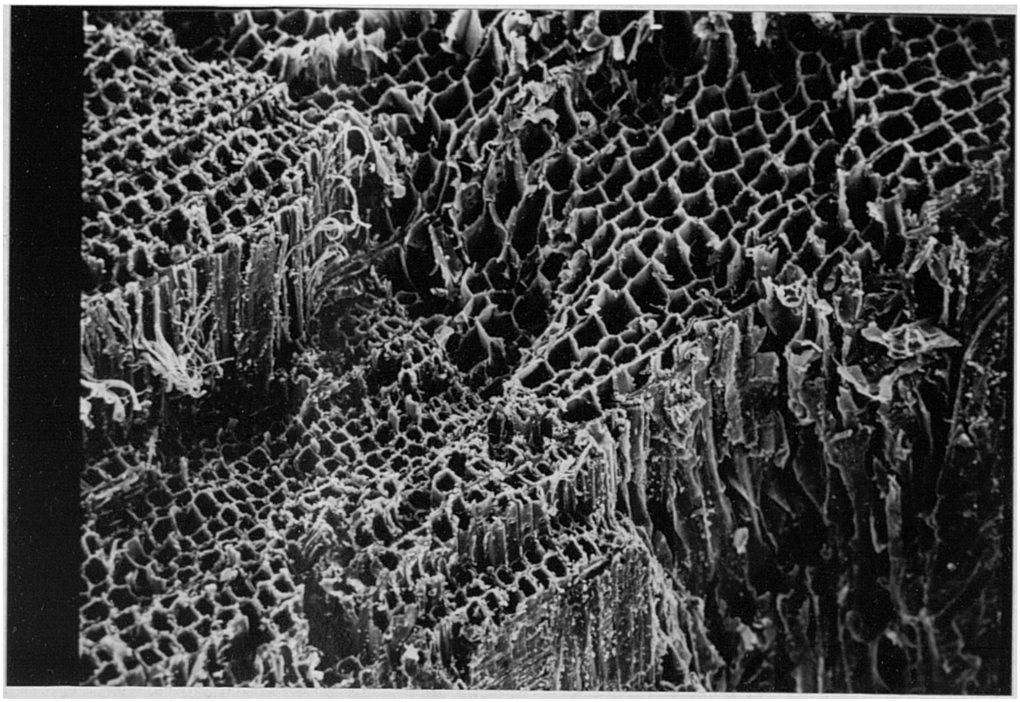


Figure (10.2) Flexural fatigue fracture surface of a Sitka spruce specimen.

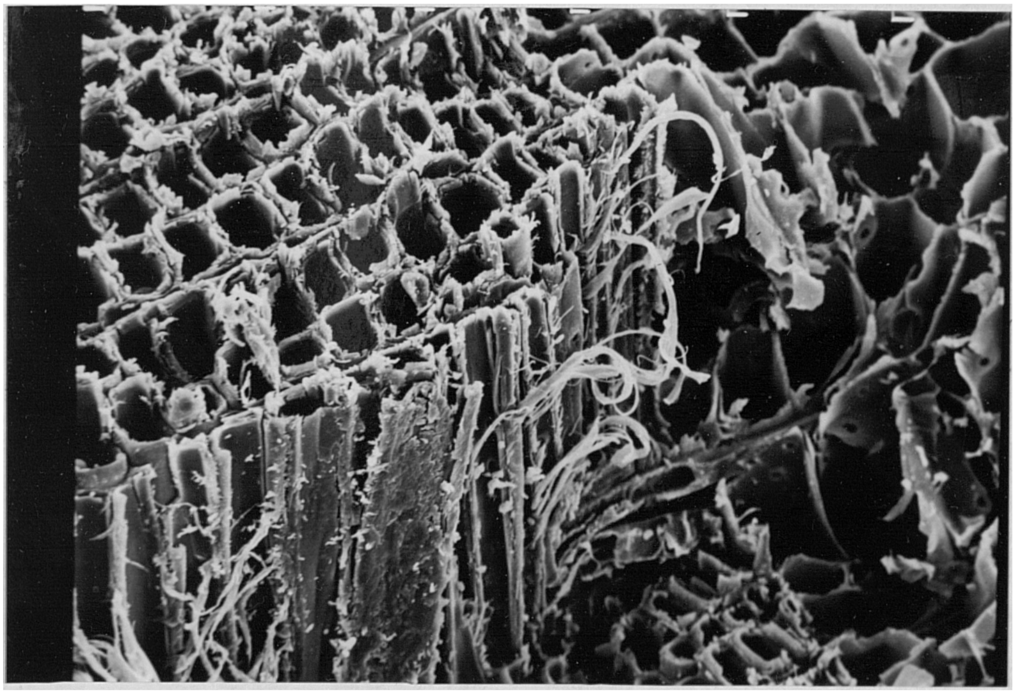


Figure (10.3) Flexural fatigue fracture surface of a Sitka spruce specimen.

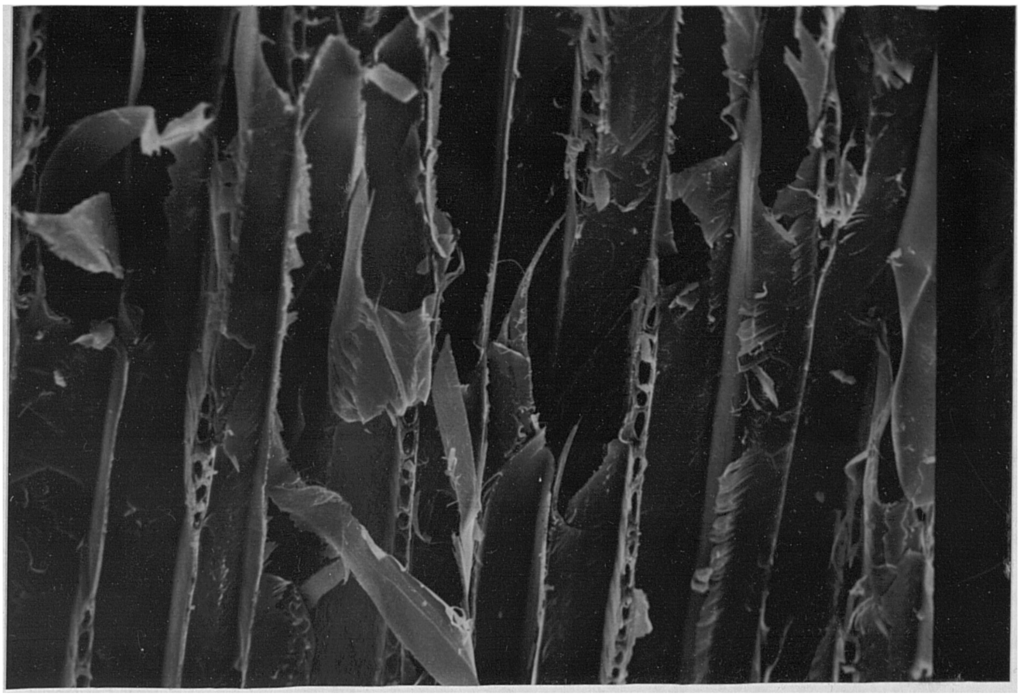


Figure (10.4) LT flexural fatigue fracture surface of Sitka spruce.

still brittle in character indicating that compression failure had probably occurred prior to tensile fracture. Another feature is seen in the fracture surface in the LT plane in Sitka spruce. In the latewood side, failure occurred at the middle lamella while in earlywood, failure was through the cell walls. A higher magnification view of the latewood fracture surface is shown in figure (10.3). Some lateral cell crushing also appears to have occurred with pullout of clumps of microfibrils. This could have occurred during the last few cycles prior to failure, from transverse or impact forces which arise due to the opening and closing of a crack. Figure (10.4) shows a LT section. There is a substantial amount of distortion and separation of the cell wall layers. Delamination of the S3 layer can also be seen on the bottom right of the figure. This appears to be quite common and extensive in fatigue compared with ramp loaded specimens although no quantitative comparison was made.

Figures (10.5) to (10.8) are micrographs of the fracture surface from the tension and compression sides of a specimen tested in fatigue. The microstructure of sliced

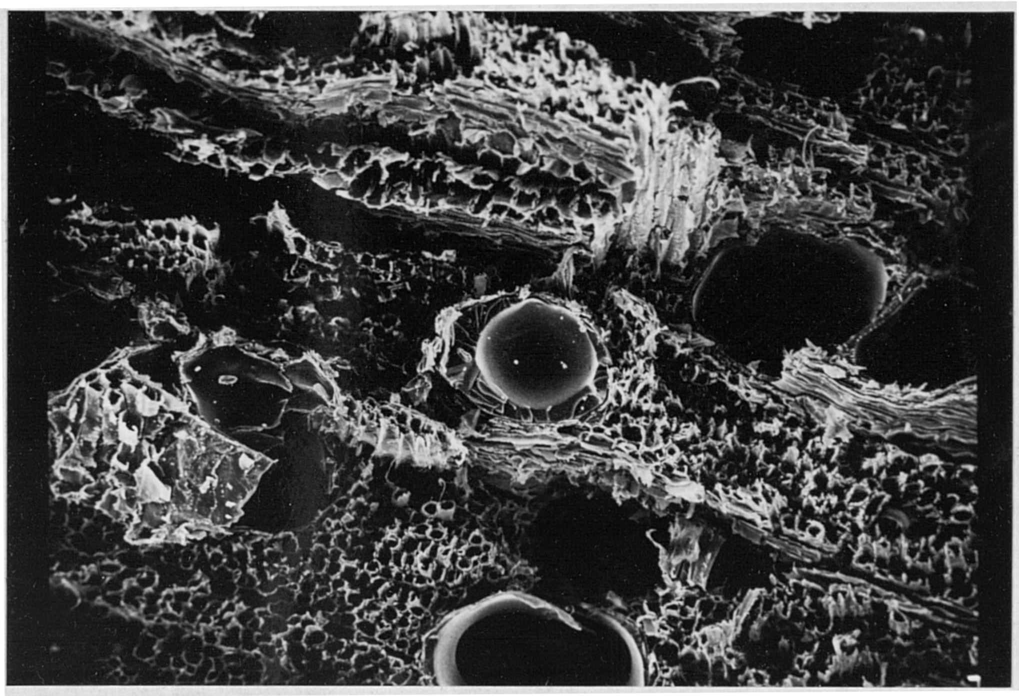


Figure (10.5) Fatigue fracture from the tension side of a sliced Khaya specimen.

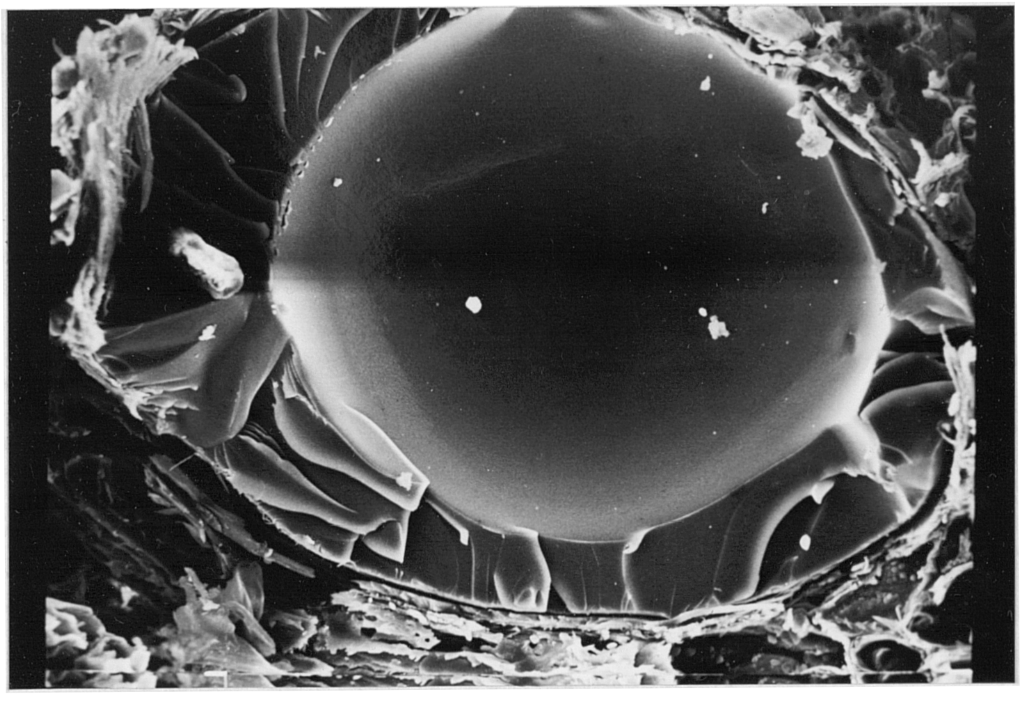


Figure (10.6) Low cycle fatigue fracture of a resin filled vessel.

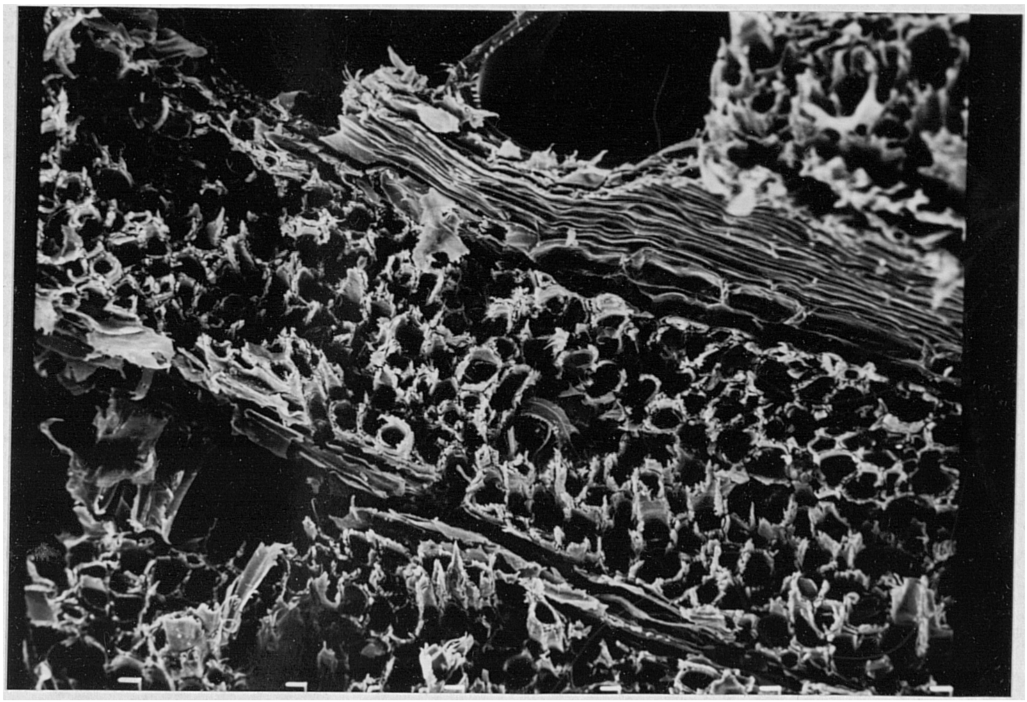


Figure (10.7) Fracture surface from the tension side of a fatigued sliced Khaya specimen.

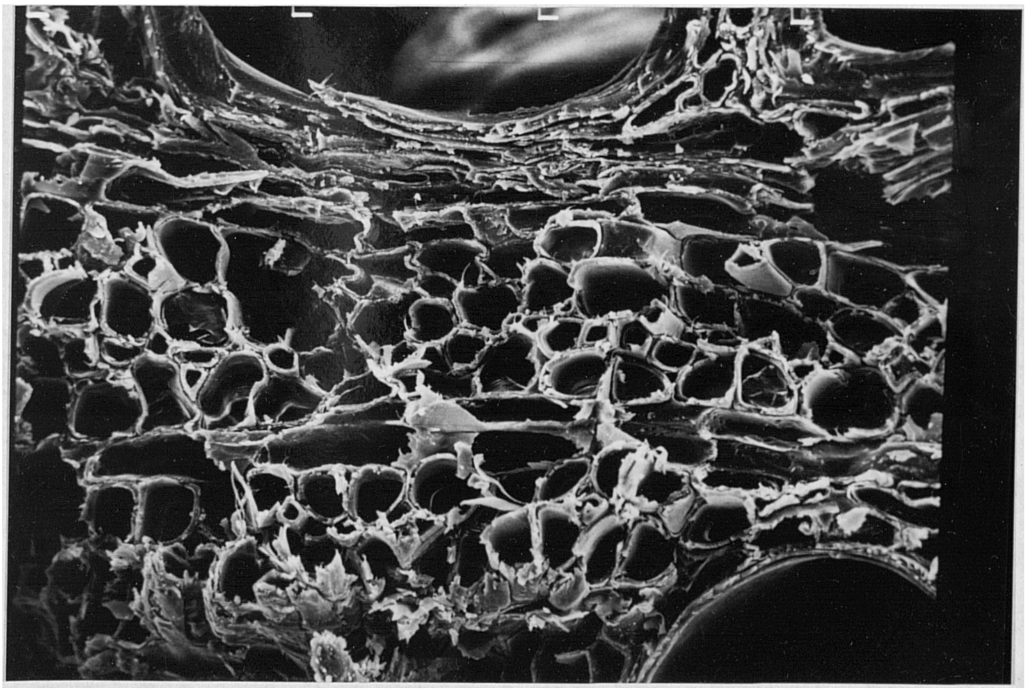


Figure (10.8) Fracture surface from the compression side of a fatigued sliced Khaya specimen.

Khaya, being a hardwood, is not as simple as Sitka with large vessels which when laminated are sometimes filled with resin. Figure (10.5) shows the low cycle fatigue fracture, on the tension side of specimen, of one such vessel filled with resin. A higher magnification view of the resin fracture is seen in figure (10.6). The initiation site appears to be on the lower right propagating around the edge before final fracture. Comparing the fracture surface of the tension side of the specimen as seen in figures (10.7) with the fracture surface on the compression side, seen in figure (10.8), a similarity with the fatigue fracture of Sitka specimens, figures (10.2) and (10.3), can be discerned. The fibrous fracture surface is seen from the tension side but a relatively brittle fracture surface is found from the compression side.

10.3 Study of Damage Accumulation by Optical Microscopy.

Sitka spruce specimens were fatigued at 75% of flexural strength and at $R = 0.1$ for 10^2, 500, 10^3, 10^4 and 10^5 cycles. Changes in the microstructure on the compression side of the specimen was observed using polarized light optical microscopy of microtomed sections. Sitka spruce was chosen for this study because of its simple microscopic structure.

Although at 75% of flexural strength, the fatigue life of Sitka is estimated (by log-linear regression analysis) to be in excess of 10^7 cycles, damage was observed as early as 500 cycles. Figure (10.9) shows a line of compression kinks or slip lines in the cell walls. This is similar to the type of damage observed in compression tested specimens of Dinwoodie (1968) and Keith and Cote (1968). A higher magnification view of the kinks can be seen in figure (10.10). The "X-shaped" kinks appear consistent with the type of deformation described by Wardrop and Dadswell (1947 (see Section 2.4.2).

At higher fatigue cycles, there is a clear trend in the development of these compression kinks. Figures (10.11) to (10.13) show the development of these kinks to a gross macroscopic crease at 10^3, 10^4 and 10^5 cycles. The kinks grows both in depth, towards the centre of the specimen, and also in length along the length of the cell wall.

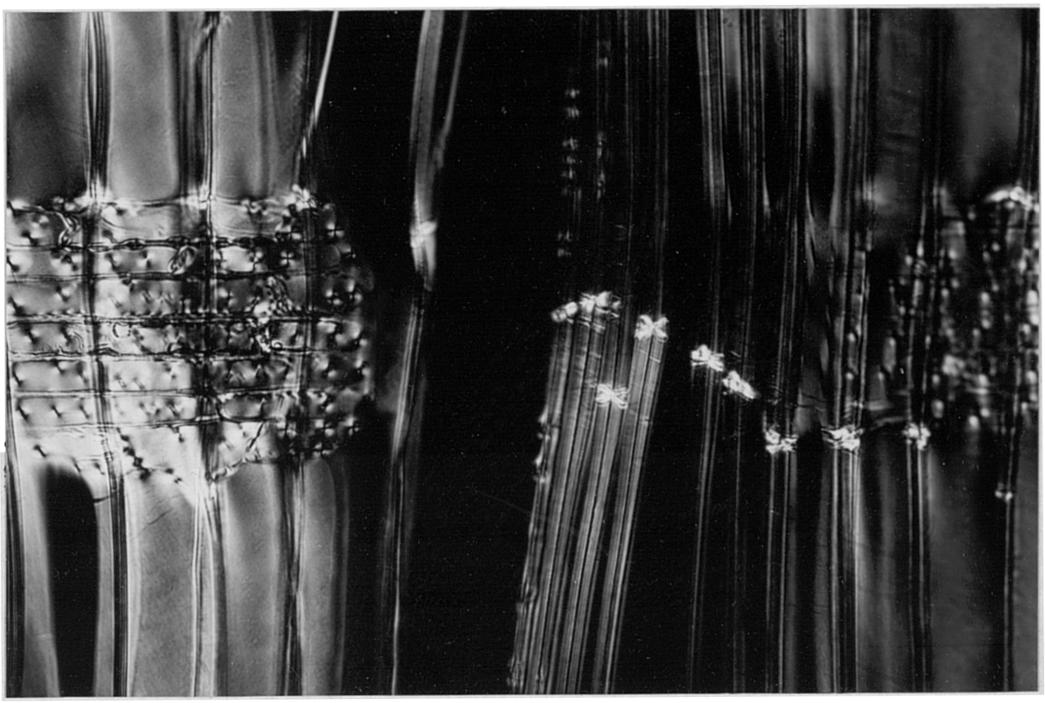


Figure (10.9) Compression kinks on a radial-longitudinal section of Sitka spruce after 500 cycles at 75% of ultimate flexural strength.

Figure (10.10) Higher magnification view of the compression failures from the central portion of figure (10.9).

This suggests progressive lateral damage occurring. Compression failure of one cell wall results in local stress transfer which causes adjacent cells to fail. As kink spreading develops, a crease is eventually formed which may be considered as analogous to a crack. There would be stress concentrations at the tip of the crease causing more cells to fail. At the same time within the crease, damage builds up along the length of the cells as the kinks themselves are compressed. The growth of the crease occurs despite the fact that the average stress towards the centre of the specimen is very low and at the neutral axis, zero. This suggests that the neutral axis is in fact moving from the

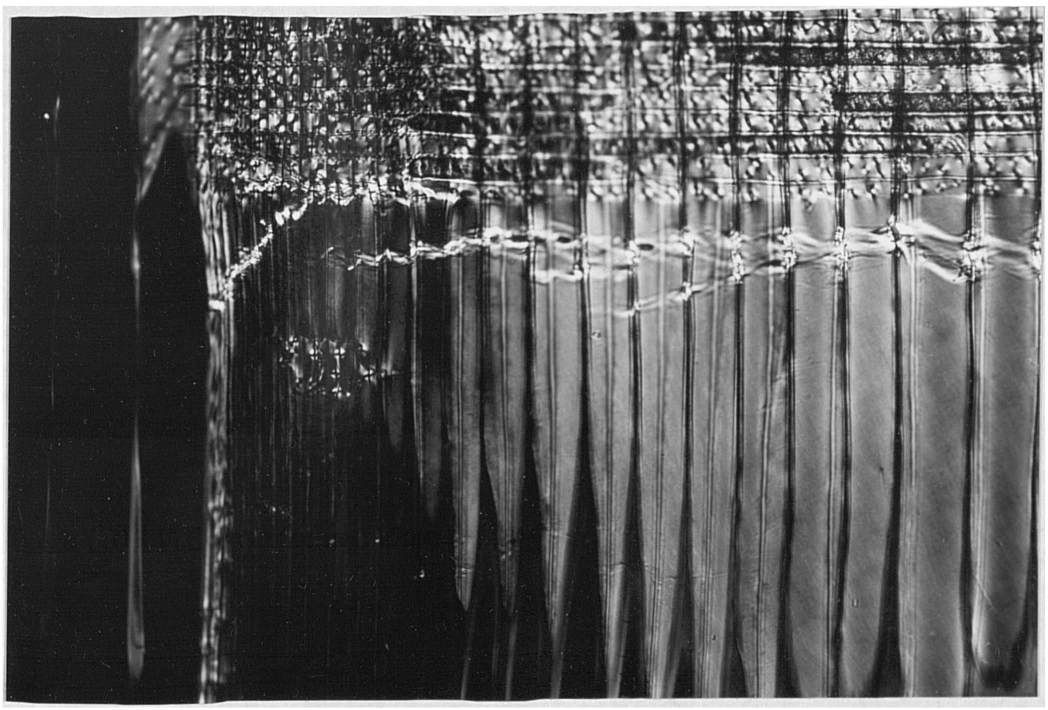


Figure (10.11) Crease formed after 1000 cycles.

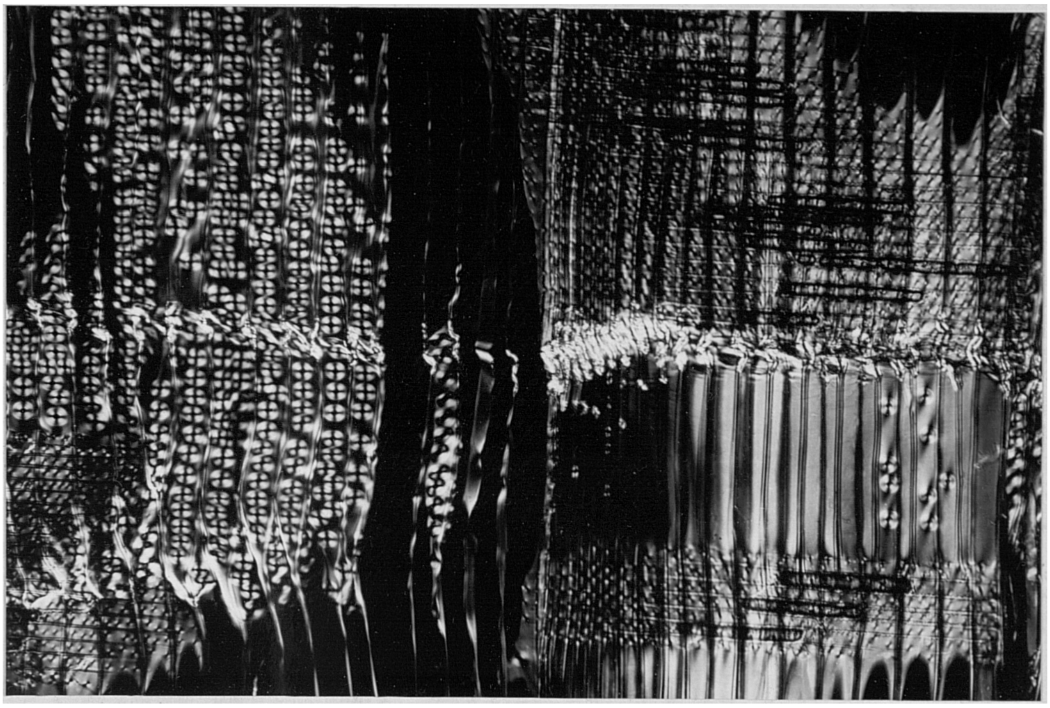


Figure (10.12) Crease formed after 10,000 cycles

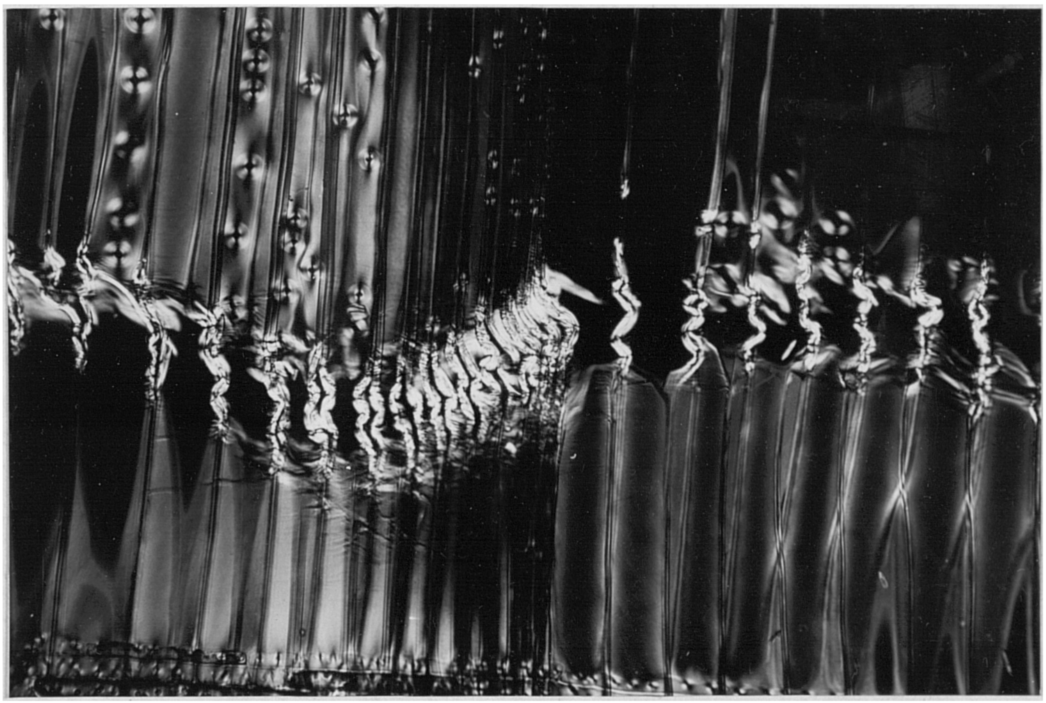


Figure (10.13) Crease, which is visible by naked eye, after 100,000 cycles.

centroidal axis of the specimen towards the tensile face. This increases the area under compression to sustain the compressive stresses as the cellular structure weakens. This results in a stress distribution across the depth of the specimen similar to that seen in figure (2.12).

The progress of the crease is also affected by latewood. In the earlywood, the line of damage progresses perpendicular to the cell walls, but in the latewood, where the cell walls are thicker and closer together, the line forms at an angle of about 30°. This suggests that the dominant type of stress changes from an axial stress on the cell wall to a shear stress. This is not surprising since in the earlywood, the wide separation between the cell walls would result in them acting more independently as columns.

Chapter 11

Discussion

11.1 Fatigue Mechanisms in Wood

There are many possible mechanisms which may operate and cause fatigue failure. In this discussion, a general description is made following the mechanisms at two levels: changes which occur to the cell wall material at the ultrastructural level and changes at the microscopic and macroscopic level. In general, the first would be more speculative as evidence of such mechanisms are difficult to obtain and in fact would be based on extension of current damage models in static and creep tests. The evidence for structural damage to the cell wall with fatigue cycles is much stronger. However, this discussion hopes to lay the foundation for relating mechanisms to the experimental evidence from this and previous work.

At the ultrastructural level, there are two possibilities: chain scission or chain slippage. Chain scission is related to ideas from tensile failure of wood. As discussed briefly in Section 2.4.1, the idea of chain scission as a failure mechanism is better than chain slippage but the stresses to initiate this remain very high. Only during the final failure process where shear stresses cause the separation of the cell wall layers can there be cell wall fracture with chain scission. This must imply therefore, that it is an unlikely process prior to the final stages of fatigue failure. From macroscopic observations of flexural fatigue, figure (10.1) where cracks may form on the tensile surface, this might be visualized as the initiation of cracks followed by crack growth. It is not a pure tensile type failure where the classical chain scission or chain slippage concepts would apply across the specimen, but rather the concepts from fracture mechanics apply with, in particular, slow crack growth along the longitudinal planes. Thus it is more likely that chain scission is not occurring at the ultrastructural level during fatigue.

Environmental factors such as moisture and temperature will affect the basic nature of the cell wall material. Moisture diffusion will inevitably result in the making and breaking of hydrogen bond. The Barkas effect, (moisture adsorption during tension and desorption during compression) would suggest that there is a high number of active hydrogen bonds. Temperature would affect the kinetics of this bond breaking and formation. Therefore, like creep, viscous flow through chain slippage would occur. It is conceivable that the microfibrillar orientation in the cell wall layers could change as a consequence of this type of mechanism. This would contribute to the fatigue strains measured with tests at R=0.1 and also the increase in modulus found during fatigue at low stresses. If such a mechanism is occurring, there would be a greater degree of chain slippage at higher moisture contents particularly in tension fatigue. Unfortunately no results are available to prove or disprove this. Visual observations of fatigue tests at 98% RH do suggest a much greater fatigue strain.

The work of Chow (1973), in examining the molecular motion of carbohydrates and lignin during creep, suggest that some molecular movement must be occurring during fatigue. On the other hand, while molecular motion may account for reversible strains, it does not constitute damage and its role is probably secondary to the fatigue process.

Changes during fatigue at the microscopic and macroscopic level are more easily observed and they provide a more consistent picture of fatigue damage mechanisms. Most clear is the development of damage in compression. As the study using optical microscopy has shown, compression damage is progressive and develops from microscopic kinks or slip lines to gross macroscopic creases. In flexure, this might not directly result in failure of the specimen but in axial compression fatigue, the kinks will form the weak points which can develop to catastrophic failure in shear. If as Dinwoodie (1968) suggests, compression kinks form at as low a stress as 25% of ultimate compression strength, then compression fatigue would have a very low fatigue limit if any. The formation of compression kinks would occur during the primary and secondary stages of fatigue strain and likewise the decrease in modulus found at higher load levels.

In tension, the fatigue mechanisms are more difficult to ascertain. The possibilities include slippage between and within the cell wall layers, cracking in secondary wall layers, and longitudinal splitting along the grain. These mechanisms are based on the fact that these are the regions of weakness in the wood structure. Classical theories on tensile wood failure already suggest that the weakness in the cellular structure is in shear. Cracking in the secondary walls has been observed in fatigue fracture surfaces using the electron microscope although it is possible that these occurred during the final failure stage. Since wood is weakest transverse to the grain direction, longitudinal splitting can easily occur. The latewood/earlywood interface would be the likely interface where such failure would occur. This has been observed in Sitka specimens where cracks almost always initiate at the interface. It is unclear however, whether cracks will form, as compression kinks do, in many parts of a specimen or initiate simply at one or a few locations and propagate. Clearly tensile fatigue tests are necessary to investigate the possible mechanisms.

Where stress reversals occur, the combined tension and compression stresses create a situation where both tension and compression fatigue mechanisms operate together. Since the compressive strength of wood is only about a third of tensile strength, damage associated with compression must dominate unless the compressive stress is very low in proportion to the tensile stress. Imayama and Matsumoto (1970) has observed microcracks in fully reversed fatigue tests of specimens starting at 60% of fatigue life. These could conceivably be formed from the opening of compression kinks in the cell walls. Compression creases are also known to greatly reduce tensile strengths hence fatigue with stress reversals will be the most damaging and a definite reduction in residual strength must occur through all stages of the fatigue life. This is in contrast to unreversed fatigue mechanisms where they do not necessary affect residual strength except towards the final tertiary stage.

11.2 Flexural Fatigue Failure

The unique condition of tensile and compressive stresses on a flexural test specimen combined with timber having a compressive strength of about a third of its tensile strength, results in a particular type of fatigue failure mechanism. Firstly, compression failures will not, unlike an axial compression test, result in catastrophic compression failure. Failure will be tensile in nature. With unreversed flexural fatigue, the neutral axis shifts towards the tensile face, similar to static flexural tests, but the shift is greater and progressive. With reversed bending, the compression failures have a weakening effect on the strength in tension and lead to failure. The following is a detailed description of the progress of flexural fatigue as seen from the results on fatigue deflections and changes in dynamic modulus.

Unreversed Flexural Fatigue

A knee or transition is found in the results for fatigue deflections and dynamic modulus at around 65% to 75% of flexural strength (see figures (9.6), (9.7), (9.13) and (9.14)). This transition point must relate to the point where the neutral axis begins to shift significantly from the centroid of the specimen towards the tensile face. Although compression strengths are only approximately 50% of the flexural strengths, the transition at 65% to 75% is reasonable since the stress distribution through the section of a flexural specimen is not uniform. The stresses decreases linearly from the outer faces to the neutral axis. The results also suggest a small shift in the transition towards a lower load level with increased fatigue cycles. This is consistent and would be as expected since the fatigue cycles will increase the compression damage. Also, any small movement in the neutral axis will increase both the stress level at the compression side and the cross sectional area in compression. The stress level on the specimen is therefore effectively moved up to a higher level beyond the transition. The movement in transition is also small for every decade increase in fatigue cycles. This therefore implies that the fatigue strength will not be linearly related to the logarithm of cycles as the linear regression on the S-N curves suggest. It may be suggested that the fatigue life is asymptotic towards infinite life at below 50% of the flexural strength.

With the presence of the transition or knee, it is convenient to consider the development of fatigue for stresses below the transition or at low stresses and for stresses above the transition or high stresses. In the primary stage where the fatigue deflections are increasing rapidly for low stresses, only the weakest links in the cellular structure fail on the compression side of the specimen. At high stresses, similar compression kinks will also form. The difference between damage accumulation at the low and high stress levels will not just be in its extent, but also in its progressive nature. At high stresses, the stress redistribution which follows when one cell wall fails will immediately affect its neighboring cells causing more failures in a domino effect. This will progress from a few scattered compression kinks to kink bands and finally a macroscopic crease. Since the formation of compression kinks is an energy absorbing event with an energy barrier involved, it is a rate governed process which will be cycle dependent rather then time dependent with each cycle showing a small input of energy into the system.

The increase in modulus seen at low stresses was unexpected but may be due to three possible mechanisms. The first and most straight forward is as suggested earlier, due to chain slippage with resulting change in the microfibrillar angle of the cell wall layers. This can happen in compression or in tension although on the tension side this will cause increase stiffness. A second possibility may be due to the compression kinks formed and the fact that at low stresses these kinks are isolated throughout the structure. Neighboring cell will be more highly stressed. This situation will develop as the stress is increasing, but as the stress level decrease, the kink does not necessarily "unkink". To do so requires energy. Therefore, the structure does not return to its original state but the kink will be in tension exerting a residual compression stress on the neighbors. In energy terms, the "unkinking" process will require more energy than the relaxation process of undamaged cells. This must imply that the minimum strain peak will show an increased residual and consequently, the strain range of subsequent fatigue cycles would decrease resulting in an increased modulus. If the kinks formed are adjacent to one another, as in a kink band, such an effect cannot occur and the modulus of the kinks, which is much lower than the unkinked cells, will dominate.

The third possibility would be one peculiar to the flexural fatigue test. It has been thus far assumed that the neutral axis is static either at the centroid of the specimen or once moved is located at the new position. This is in fact unlikely to be the case. It is more likely for the neutral axis to move towards the tension face as the stress increases, but when the stress is decreased, it will move back towards the centroidal axis but not totally. If the structure recovers more slowly in compression than in tension (which is likely to be the case), there would be a hindrance to the return of the neutral axis to its original position. As a consequence, the movement of the neutral axis would gradually be reduced. This would create residual stresses within the specimen and prevent the specimen from returning to its original minimum strain peak. The change in the minimum strain peak would therefore be greater than the maximum giving an apparent increase in modulus.

During the secondary stage, the events of the primary stage may stabilize and develop progressively but at low stresses, it is likely that modulus increases will show a limit and subsequently decrease when the conditions necessary for the higher modulus begin to break down. It is possible that no decrease will occur hence implying an infinite fatigue life. This must be accompanied also by no further change in fatigue strains. At a high stress, the secondary stage is characterized by a linear decrease in modulus and a linear increase in fatigue strain. The gradual shift in the neutral axis will result in the first signs of failure when, due to weaknesses and stress concentrations on the tension face, cracks begin to initiate. There is a natural Mode 1 or crack opening condition created when the specimen bends. The crack will therefore grow along the length of the specimen in the direction of the grain. This would be crack growth on the LT and LR planes which is a low fracture toughness plane.

The weakening that comes from crack formation and growth must inevitably lead to the tertiary stage where the gradual breakup of the specimen results in a greatly reduced modulus and large fatigue strains. Only with the cracks formed on the tension face would there be seen any reduction in residual strength. As a consequence, it is not surprising that the results of Kommers (1943) and Dobraszczyk (1983) show no decrease in residual strengths. The tertiary stage is usually very short in comparison with the life of the specimen and it would be very unlikely to stop a test for residual strength evaluation within this stage to measure a fall in strength.

Reversed Flexural Fatigue

At an R-ratio of -1, the neutral axis is unlikely to move and become the dominant factor in fatigue damage. But clearly, the low compression strength of wood has a strong effect. Unlike unreversed bending fatigue, the complete opening of compression kinks would occur equally on both sides of the specimen and unless there is an imbalance of properties, both sides will suffer equal damage and the neutral axis will remain static. Kink bands formed during the compression part of the fatigue cycles will become the lines of weakness where tensile cracks can form. This would be the likely source of the microcracks observed by Imayama and Matsumoto (1970).

This damage process implies that no increase in modulus will occur but rather a consistent decrease, as was found in the tests. It might also be anticipated that a transition or knee in the reduction of modulus with load level characteristic might be present at or below 50% of flexural strength corresponding to the compression strength of wood. This transition however was not found and the reduction in modulus appeared to linearly decrease with load for different cycle levels as seen in figure (9.18). This suggests that the compression kinks had formed in the specimen even at very low load levels and that the extension of the kinks when stresses are reversed made their development progressive. This would reinforce the view suggested be Dinwoodie (1968) (see section 2.4.2) that compression kinks form from very low stresses, even as low as 25% of compression strength or 12% of flexural strength.

11.3 Implications of Results to WEC Blades Design

The work had a weakness in its applicability to WEC blade design in that the tests were conducted in flexure whereas the design of WEC blades requires axial fatigue data and in particular, compression fatigue data. This however does not mean that the research has had little benefit. The results in fact have seen use in some of the earlier blade designs before axial results became available. The first contribution of the results was to provide a body of data on the R-ratio effect.

As discussed in section 5.1, the many sources of fatigue loadings on the blade create a wide range of loading conditions. Without an appropriate model for the summation of fatigue cycles, two design approaches were possible. The first is to select an extreme load condition (such as loads due to a one in fifty year high velocity wind gust) and to design so that the blade does fail in such an event. Such a design is a static design and must be part of the total design procedure. The second approach is to use a modified form of the Goodman Diagram to 10^7 cycles shown in figure (8.27). The modified diagram is based on a parallel Goodman line to that obtained but factoring the data according to the ratio between the compression strength to the flexural strength. This new line is shown in figure (11.1). This line is therefore taken to be the 50% failure probability level upon which a further safety factor needs to be applied. The design stress level can therefore be obtained from the diagram to apply for blade design to withstand the dominant operating fatigue loads.

The two design approaches, especially when applied together, provide a basis for design of WEC blades. However, the work has also revealed the strengths and

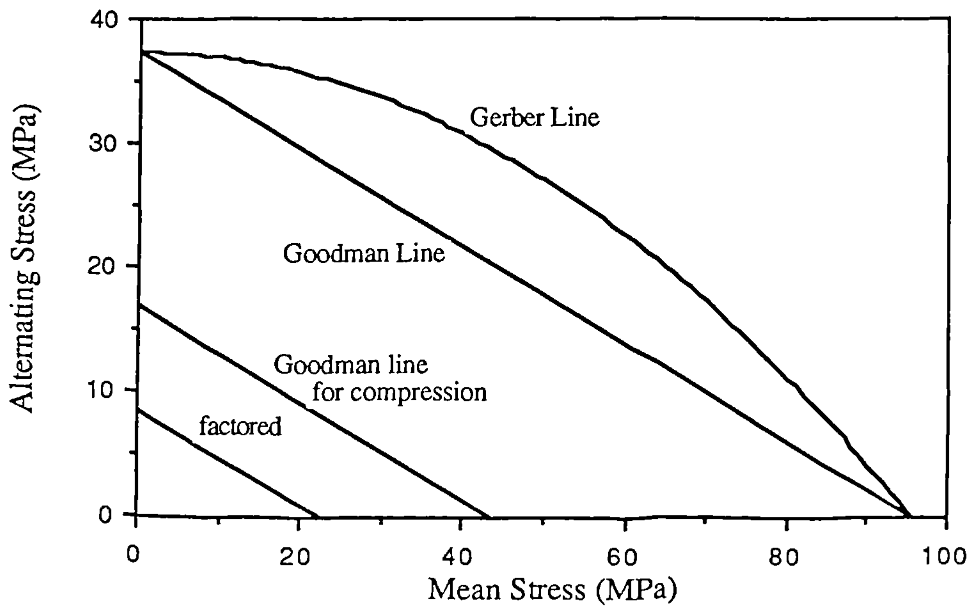


Figure (11.1) Constant life diagram for sliced Khaya showing the modified Goodman lines for fatigue design. The factored line has an arbitrary safety factor applied.

weaknesses in these two approaches. There are many advantages in these approaches. Firstly, both approaches are conservative in assessing the damaging effect of fatigue and have large safety factors. Also, whereas previous fatigue data is largely from reversed or unreversed fatigue tests at constant deflection, the data from the tests provides a spectrum of fatigue life data over a complete range of R-ratios at constant load. Another advantage is that the tests are based on laminated Khaya ivorensis, which is the species used in the blades. Fatigue tests on sliced and rotary cut veneers have also shown that the rotary cut material has a greater fatigue strength. Stiffness is also a very important consideration in design. Reductions in stiffness would endanger the blade integrity and the results confirm that at the relatively low fatigue load levels of the blade stiffness changes need not be of significant concern.

The results have however shown some great limitations in the design approaches for wood in fatigue. The discussions above have emphasized the weakness of wood in compression. The situation is especially severe when load reversals occur. While the range of R-ratios provided by the modified Goodman line gives design guidelines for stress reversals, they do not quantify the damage from sequence effects. The results in section 8.5 suggest that sequence effects are likely. Indeed, a simplistic consideration of Kommers' (1943) results shown in figure (4.2) also imply that sequence are present. The figure suggest that 5000 cycles at, for example, 80% of the static strength would, when the stress is reversed, result in a residual strength of only 30% of the static strength. It should be noted that 5000 cycles at 80% represent only about one tenth of the anticipated fatigue life and reversed bending fatigue at 30% should have a fatigue life of $>10^6$ cycles. The most dangerous sequences are therefore likely to be compression fatigue followed be tension or reversed fatigue stresses. Investigating sequence effects is probably the next area for further research. Another important gap in the understanding of wood fatigue is the interaction between fluctuations in moisture content with fatigue damage accumulation. The effect of moisture has been assessed and shown to greatly affect the fatigue strength of wood. WEC blades being operated in the outdoor environment will see not just high humidities but also seasonal fluctuating humidities. Although the wood is sealed in resin impregnated glass fabric, this will only reduce the rate and extremes of the fluctuating moisture content of the wood. No consideration of this is being made in the design.

Chapter 12

Conclusions

The problems in utilizing wood as a material for Wind Energy Converter blades have been considered in this study. The conclusions from this research may be listed as follows:

(a) A range of flexural fatigue tests under load control have been carried out on Sitka spruce, laminated sliced and rotary cut Khaya ivorensis, and unidirectional and 0/90 compressed Beech laminates.

(b) Fatigue life is largely species independent when normalized by its static strength. This is similar to the normalising effect density has on static strengths. Only where the wood structure is significantly changed as for the compressed beech laminates are fatigue lives affected.

(c) Moisture has a detrimental effect on fatigue life. Its effect is not only in reducing the static strength but also accelerates the fatigue damage process.

(d) The R-ratio tests show that the lower the R-ratio, the lower the fatigue life. In the Constant Life Diagram for sliced Khaya laminates constituted from S-N curves at each R ratio, the Gerber or parabolic function best relates the mean to alternating stress. It shows the especially damaging effect of stress reversals.

(e) Block loading tests have been carried out which suggests that the sequence of the load blocks can affect fatigue life. More work however is necessary to confirm this and to consider other combinations of R-ratios and load levels for sequence effects.

(f) SArGen, a computer control and data acquisition system has been developed for use in fatigue tests.

(g) The system can automatically correct for fluctuations in load levels and be programmed for block fatigue tests. Using the system, modulus and fatigue strains have been monitored during the test.

(h) With unreversed fatigue tests the changes in the minimum and maximum deflection peaks show three identifiable stages similar to creep tests. A transition has been found at around 65% of static flexural strength. Above the transition, fatigue deflections increase rapidly with cycle but below it, fatigue deflections remain relatively small. The rapid increase above the transition is believed to be associated with the movement of the neutral axis.

(i) With stress reversals, the minimum and maximum deflection peaks changes in opposite directions. This is due to the opening and closing of compression failures and the results clearly indicate the destructive nature of this mode of fatigue.

(j) The modulus changes in parallel with the deflections but at low stresses and without stress reversals, the modulus was found to initially increase with a transition at about 70% of the static strength above which a rapid decrease in modulus occurred. Three possible mechanisms have been proposed to explain this behavior. With stress reversals, the modulus decreased even at stresses as low 20% of the static strength.

(k) A study of fatigue fracture surfaces have been carried out on the electron microscope. Optical microscopy has also been used to study the development of compression creases with cycles. Sitka specimens fatigued at 70% of the static strength showed compression kinks developing after as few as 500 cycles, developing into macroscopic creases after 100,000 cycles.

(l) Possible fatigue mechanisms have been discussed and proposed. A complete description of flexural fatigue damage mechanisms has been proposed from the experimental evidence obtained.

(m) It is proposed that a probability based approach to fatigue design should be the direction for the future in developing a reliable WEC blade. This approach has been reviewed and it is concluded that much remains unknown especially in fatigue life prediction. A factored approach remains the only currently feasible design method but many limitations and uncertainties exists relating to sequence effects.

References

Armstrong, L.D. and Christensen, G.N. (1961). Influence of Moisture Change on deformation of Wood under Stress. *Nature*, **191**, pp 869-870.

American Institute of Timber Construction (1974). Timber Construction Manual. 2nd. Edition, John Wiley & Sons.

Ang, A.H.S. and Cornell, C.A. (1974). Reliability Basis of Structural Safety and Design. *Proc. of ASCE, Journ. of Struct. Div.*, **Vol 100, No. ST9**, pp 1755-1769.

Ansell, M P and Tsai, K T. (1984). Fatigue Testing of Wood Comp. for Aerogenerator Rotor Blades. *Proc. of the 6th BWEA Wind Energy Conf.*, (Musgrove, P, ed), pp. 239-255. Cambridge University Press.

American Society for Testing and Materials. *Standard Methods of Salt Spray (Fog) Testing*. ASTM Designation B117-73.

American Society for Testing and Materials. *Standard Methods of Testing Small Clear Specimens of Timber*. ASTM Designation D143-52 (72).

American Society for Testing and Materials. *Standard Methods of Static Test of Timber in Structural Sizes*. ASTM Designation D198-76.

Aplin, E N and Keenan, I J (1977). Limit-States Design in Wood : A Canadian Perspective. *For. Prod. Journ.* **27**(7), pp 14-18.

Atack, D, May, W D , Morris, E L and Sproule, R N (1961). The Energy of Tensile and Cleavage Fracture of Black Spruce. *Tappi,* **44**, pp 555-567.

Bach, L. (1973). Reiner-Weisenberg's Theory Applied to Time Dependent Fracture of Wood Subjected to Various Modes of Mechanical Loading. *Wood Science,* **5**:3 pp 161-171.

Barker, D and Smith, P (1983). A micro-processor controller for a servo-hydraulic Fatigue Machine. *Proc. of SEECO 83: Digital techniques in fatigue*, pp 277-290.

Barrett, J D and Foshi, R O (1978). On the Application of Brittle Fracture Theory, Fracture Mechanics, and creep rupture Models for the prediction of the Reliability of Wood Sructural Elements. *Nat. Science Foundation Workshop. On. Contiductive Rel. for Wood and Wood-based Mat.,* Minnowbrook N.Y.

British Standards Institute. *Methods of Testing Small Clear Specimens of Timber.* BS 373:1957.

Boatright, J.W.S. (1977). Studies on the Fracture and Fatigue of Wood. M.Sc. thesis, University of Cape Town, Pretoria, South Africa.

Bodig, J. and Jayne, B.A. (1982). *Mechanics of Wood and Wood Composites.* New York, USA: Van Nostrand Reinhold Company Inc.

Chow, S (1973). Molecular Rheology of Coniferous Wood Tissues. *Trans. Soc. Rheology* **17**:1 pp 109-128.

Craig, T M.(1978). The use of digital computer in a fatigue testing facility. *Proc. of SEECO '78, Applications of Computers in Fatigue.,* pp 4.1-4.17.

Dietz, A.G.H. and Grinsfelder, H (1943). Behaviour of Plywood Under Repeated Stresses. *Trans. ASME,*April 1943, pp 187-191.

Dinwoodie, J.M. (1966). Induction of Cell Wall Dislocations (slip Planes) during the Preparation of Microscope Sections of Wood. *Nature*, **Vol.212**, pp 525-527.

Dinwoodie, J M. (1968). Failure in Timber Part 1: Microscopic Changes in Cell-Wall Structure Associated with Compression Failure. *Journal of the Inst. of Wood Science*, **No. 21**, pp 37-53.

Dinwoodie, J.M. (1974). Failure in Timber Part 2: The Angle of Shear through Cell Wall during Longitudinal Compression Stressing. *Wood Science & Tech.*, **Vol. 8**, pp 56-67.

Dinwoodie, J M. (1978). Failure in Timber Part 3: The Effect of Longitudinal Compression on Some Mechanical Properties. *Wood Science & Tech.*, **Vol 12**, pp 271-285.

Dinwoodie, J.M. (1981). *Timber, its Nature and Behavior.* New York, USA: Van Nostrand Reinhold Company Inc.

Dobroszczyk, B.(1983). An Investigation into the Fracture and Fatigue Behaviour of Wood.Phd thesis, University of Bath, U.K.

Fleck, N.A. and Hooley, T. (1983). Development of Low Cost Computer Control. *Proc of SEECO '83, Digital Techniques in Fatigue.*, pp 309-316.

Foshi, R.O. and Barrett, J.D. (1982). Load Duration Effects in Western Hemlock Lumber. *Proc. of the ASCE, Journ. of Struct. Div.*, **108**:ST7, pp 1494-1510.

Freas, A. and Warren, F. (1959). Effect of Repeated Loading and Salt-water Immersion of Flexural Properties of Laminated White Oak. *For. Prod. Journ.*, **9**:2, pp 100-103.

Fuller, F.B. and Oberg, T.T. (1943). Fatigue Characteristics of Natural and Resin-Imnpregnated, Compressed, Laminated Woods. *J. Aeronautical Sciences,* March 1943, pp 81-85.

Gerhards, C.C. (1979). Time related Effects of Loading in Wood Strength: A linear Cumulative Damage Theory. *Wood Science,* **11**:3, pp 139-144.

Gerharz, J.J. (1982). Prediction of Fatigue failure. *AGARD lecture Series No. 124. Prac. Consideration of Design, Fabrication & Tests for Composite Materials.,* pp. 8.1-8.22.

Gillwald, W. (1961). On the Determination of the Deformation of Wood under Oscillating Load. *Holz Als Roh Und Werkstoff,* **19**:3, pp. 86-92.

Griffiths, J.R.(1983). Fatsys. A computer controlled system for fatigue crack growth data collection using an Ausler Vibrophone. *Int. J. Fatigue,* **5**:4, pp.193-197.

Grossman P.U.A., Armstrong L.D. and Kingston R.S.T. (1969). An assesment of Research in Wood Rheology. *Wood Science & Tech.,* **3**, pp. 324-328.

Grossman, P.U.A. (1976) Requirements for Model that exhibits Mechano-Sorptive Behaviour. *Wood Science & Tech.,* **10**, pp. 163-168.

Goodman, J.R. (1981). Code Comparison of Factor Design for Wood. *Proc. of the ASCE, Journal of the Structural Div.,* **107**: 518 pp. 1511-1527

Hearman, R.F.S. (1948). Elasticity of Wood and Plywood. *Special Report on Forest Products Research, London,* No 7, pp 87.

Hearman, R.F.S. and Paton, J.M. (1975). Moisture Content Changes and Creep of Wood. *For. Prod. Jour.*, 14(8) pp. 357-359.

Hoyle R.J., Griffith M.C. and Itani R.Y. (1985). Primary Creep in Douglas-fir Beams of Commercial Size and Quality. *Wood and Fiber Science,* **17**:3, pp 300-314.

Ibuki Y., Sasaki H., Kawamoto M. and Maku T. (1962). On the Endurance of Glued-Laminated Wood, (The plane bending fatigue strength of glued-laminated wood). *J. Jap. Soc. Test. Mater.,* **11**:101, pp.103-109.

Ibuki Y., Sasaki H., Kawamoto M. and Maku T. (1963). The Plane Bending Fatigue Strength of Glued Laminated Wood. *Mokuza Kenkyu,* No. 31 pp.11-22.

Imayama, N. and Matsumoto, T. (1970). Studies on the Fatigue of Wood I. *J. Jap Wood Res. Soc.,* **16**:7, pp. 319-325.

Imayama, N. and Matsumoto, T. (1974). Studies of the Fatigue of Wood II. *J. Jap Wood Res. Soc.,* **20**:2, pp. 53-62.

Jablonski, D.A. and Lee, B.H. (1983) Automated Fatigue Crack growth Rate Testing using a Computerised Test System. *Proc of SEECO '83, Original Techniques in Fatigue.,* pp. 291-308.

Jamieson, P. and McLeish, D. (1983). The HWP 300 Wind Turbine. *IEE Proc.,* Vol. **130** Pt.A. No.9 pp. 550-554.

Jenkins, G. (1962). Coupon Fatigue Tests on Plywood. *Bristol Aircraft Ltd., Struct.and Mats. Lab. Rep. No. 171-76B - 3040.*

Keith, C.T. and Cote, W.A. (1968). Microscopic Characterisation of Slip Lines and Compression Failures in Wood Cell Walls. *For Prod. Journ.*, **18**: 3. pp. 67-74.

Keith, C.T. (1971). The Anatomy of Compression Failure in Relation to Creep-Inducing Stresses. *Wood Science,* **4**:2, pp. 71-82.

Keith, C.T. (1972). The Mechanical Behaviour of Wood in Longitudinal Compression. *Wood Science,* **4**:4, pp. 234-244,

Keith, C.T. (1974). Longitudinal Compressive Creep and Failure Development in White Spruce Compression Wood. *Wood Science,* **7**:1, pp. 1-12.

Kellogg, R.M. (1958). Strain Behaviour of Wood Subjected to Repetitive Stressing in Tension Parallel to the Grain. *For. Prod. Journ.,* **8**:10, pp. 301-307.

Kellogg, R.M. (1960). Effect of Repeated Loading on Tensile Properties of Wood. *For. Prod. Journ.,* **10**:11, pp. 586-594.

Kollman, F. and Schmidt, E. (1962). Structural Derangement and Loss in Strength of Permanently Stressed Coniferous Wood. *Holz als Roh Und Werkstoff,* **20**:9, pp 333-338.

Kommers, W.J. (1943). Effect of 1000 Cycles of Repeated Bending Stress on 5-ply Sitka Spruce Plywood. *U.S. Forest Prod. Lab. Rep.No. 1305.*

Kommers, W.J. (1943). Effect of Ten Repetitions of Stress on the Bending and Compressive Strengths of Sitka Spruce and Douglas Fir. *U.S. Forest Prod. Lab. Rep.No. 1320.*

Kommers, W.J. (1943). The Fatigue Behaviour of Wood and Plywood subjected to Repeated and Reversed Bending Stresses. *U.S. Forest Prod. Lab. Rep.No. 1327.*

Lancioffi, A. (1983). A Method for Improving the Performance of Fatigue Machines in Variable Amplitude Loading Tests. *Int. J. Fatigue,* **5**:4.

Lark, R.F. (1983). Construction of Low-cost, MOD-0A Wood Composite Wind Turbine Blades. *Proc. 28th Nat. Soc. for the Advancement of Mat. and Process Eng.* pp 1277-1291.

Lavers, G.M. (1969). The Strength Poperties of Timber. *Bulletin 50, Forest Products Res. Lab. (2nd Edn.),* HMSO.

Lewis, W.C. (1946). Fatigue of Wood and Glued-Wood Constructions. *Proc. ASTM 46,* pp. 814-835.

Lewis, W.C., (1951). Fatigue of Wood and Glued Joints used in Laminated Constructions. *Proc. U.S. Forest Prod. Res. Soc.,* **5**, pp. 221-229.

Lewis, W.C. (1960). Design Considerations for Fatigue in Timber Sttructures. *J. of the Struct. Div. Proc. of ASCE.,* pp. 15-23.

Lewis, W.C. (1962). Fatigue Resistance of Quarter-Scale Bridge Stringers in Flexure and Shear. *U.S. For. Prod. Lab. Rep.No. 2236.*

Liska, J.A. (1950). Effect of Rapid Loading on the Compressive and Flexural Strength of Wood. *For. Prod. Lab. Rep. No. R 1767.*

Madsen, B. (1978). Time-Strength Relationships for Lumber. *Proc. Ist. Int. Conf. on Wood Fracture,* Canada. pp 111-128.

Maku, T. and Sasaki, H. (1963). The Rotating Bending Fatigue Strength of Glued Laminated Wood. *Mokuzai Kenkyu,* No. 31 pp 1-10.

Maku, T. and Sasaki, H. (1963). The Plane Bending Fatigue Test of Glued Laminated Wood Cantilever. *Mokuza Kenkyu,* No. 31 pp. 23-40.

Malhotra, S.K. and Bazan, I.M.M. (1980). Ultimate Bending Strength Theory for Timber Beams. *Wood Science,* **13**:1. pp.50-63.

Mark, R.E. (1967). *Cell Wall Mechanics of Trachieds.* Yale Univ. Press, New Haven.

McLain, T.E. and Woeste, F.E. (1986). Rate of loading adjustment for proof testing of lumber in tension. *For. Prod. Journ.,* **36**:9, pp 51-54.

McNatts, J.D. (1978). Linear Requesions of Fatigue Data. *Wood Science,* **11**:1, pp.39-41.

Meyer, K.H. (1950). *Natural and Synthetic High Polymers.* High Polymer Series Vol. IV., Interscience, New York.

Mindness S., Madsen B. and Barrett J.D. (1978). Rate of Loading and Duration of Load Tests on Douglas-fir in Tension Perpendicular to the Grain. *Proc. Ist. Int. Conf. on Wood Fracture,* Canada. pp 143-159.

Nadeau, J.S., Bennet, R. and Fuller, E.R. (1982). An Explanation for the Rate of Loading and Duration of Load Effects in Wood in Terms and Fracture Mechanics. *J. Mat. Sci.,* **17,** pp. 2831-2840.

Nakai, T., and Grossman, P.U.A. (1983). Deflection of Wood Under Intermittent Loading, Part 1: Fortnightly Cycles. *Wood Science & Tech.,* **17**:1, pp 55-67.

Noak, D. and Stockmann, V. (1969). Studies on the Dynamic Fatigue Behaviour of Wood under Alternating Tension Load - Part II : Dynamic Fatigue Behaviour of Beechwood under Constant Conditions of Temperature and Moisture Content. *Holz. als Roh und Wenstoff,* **27**:12, pp. 464-472.

Nowack H., Hamchamann D, Baumboff N. and Jocoby G. (1979). Developments in Hardware and software for Computer Controlled Servohydraulic Fatigue Testing Systems. *Int. J. Fatigue,* **1**:2, pp.93-102.

Nowack, H. (1981). Fatigue Test Machines. *AGARD Lecture Series No.118, Fatigue Test Methodology.,* pp 3.1-3.23.

Osgood, C.C. (1982). *Fatigue Design.* Pergamon Press, 2nd Edn.

Ota, M. and Tsubota, Y. (1966). Studies on the Fatigue of 2-Ply Laminated Wood I. *Journ. Japan Wood Res. Soc.,* **12**:1, pp. 26-29.

Ota, M. and Tsubota, Y. (1966). Studies on the Fatigue of 2-Ply Laminated Wood II. *Journ. Japan. Wood Res. Soc.,* **12**:2, pp. 90-95.

Ota, M. and Tsubota, Y. (1966). Studies on the Fatigue of 2-Ply Laminated Wood III. *Journ. Japan. Wood Res. Soc.,* **12**:5, pp. 210-214.

Ota, M. and Tsubota, Y. (1967). Studies on the Fatigue of 2-Ply Laminated Wood IV. *Journ. Japan Wood Res. Soc.*, **13**:4, pp. 131-137.

Panshin, A.J. and Zecuw, C D. (1970). *Textbook of Wood Technology VI. Structure, Identification, User and Properties of tCommercial Woods of the United States and Canada.* McGraw-Hill Book Co. 3rd Ed.

Pearson, R.G. (1972). The effect of duration of load on the bending strength of wood. *Holzforschung,* **26**:4, pp 153-158.

Pretlove, A.J. and Worthington, P.J. (1983). A review of Aero-generator Fatigue Problems. *Int. J. Fatigue,* **5**:1, pp. 15-22

Rose, G. (1971). The Mechanical Behaviour of Pinewood Under Dynamic Fatigue : Stress Depending on Kind and Amount of Load, Moisture Cntent and Temperature. *Canada Forest Prod. Lab. Trans. No. 198.* From: *Holz als Roh und Werkstoff,* **23**, pp. 271-284

Sato K., Noguchi M. and Fushitani M. (1983). The Characteriscs of Acoustic Emissions of Wood Generated during Several Types of Loading. *Journ. of the Japan Wood Res.*, **29**:6, pp 409-414.

Schijve, J. (1972). The Accumulation of Fatigue Damage in Aircraft Materials and Structures. *AGARD AG157.*

Schniewind, A.P. (1968). Recent Progress in the Study of the Rheology of Wood. *Wood Science & Tech.,* **2**, pp 188-206.

Sekhar, A.C., Sukla, N.K. and Gupta, V.K.(1963). A Note on Fatigue Properties of Timber : Effect of Torsional Stress and Moisture Content. *J. Natn. Bldgs. Org.*, **8**:4, pp. 36-40.

Sekhar, A.C., Sukla, N.K. and Gupta, V.K. (1964). Effect of Torsional Stress and Moisture Content on the Fatigue Prop. of Timber. *Holz. Als. Roh. und Werkstoff,* **22**:7, pp. 264-266.

Sekhar, A.C. and Sukla, N.K. (1979). Some Studies on the Influence of Specific Gravity on Fatigue Strength of Indian Timbers. *Journ. of the Indian Academy of Wood Sci.,* **10**:1, pp. 1-5.

Schutz, D. (1981). Variable Amplitude Fatigue Testing. *AGARD Lecture Series, No. 118, Fatigue Testing Methodology,* pp 4.1-4.31.

Schutz, D. and Gerharz, J.J. (1977). Fatigue Strength of a Fibre-Reinforced Material, *Composites,* Oct. 1977, pp 245-250.

Sieminski, R.(1960). Fatigue Strength of Pinewood (Pinus Silvestris). *Holz. Ab Roh und Werkotoff,* **18**:10, pp. 369-377. *Trans NTC-68 10252-11L.*

Sims, G.D. and Gladman, D.G. (1978). Effect of test Conditions on the Fatigue Strength of Glass-Fabric Laminate : Part A-Frequency. *Plastics and Rubber : Materials and Application,* pp.41-48.

Smith, M. (1984). Private communication, James Howden Ltd.

Smith, R.F. and Abbot, K.R. (1985). Microcomputer Control System for Mechanical Testing Machines. *J. of Inst. of Mat.: Metals and Materials.* **1**:10, pp 615-618.

Spencer, R. (1978) Rate of Loading Effect in Bending for Douglas Fir Lumber. *Proc. of 1st Int. Conf. on Wood Fracture,* Canada. pp 259-280.

Sterr, R. (1963). Investigation on the Fatigue Strength of Laminated Wood Beams. *Holz. als Roh und Werkstoff,* **21**:2, pp. 47-61, (Trans. No. 171. Canada. Dept. of For. For Prod. Res. Branch).

Strickler, M.D. and Pellerin, R.F. (1973). Rate of Loading Effect on Tensile Strength of Wood Parallel to Grain. *For. Prod. Journ.,* **23**:10, pp 34-36.

Styles, D D. (1978). Microcomputer Control of Fatigue Crack-Growth Experiments. *Proc. of SEECO 78,* pp.16.1-16.13.

Sugiyama, H. (1967). On the Effect of the Loading Time on the Strength Properties of Wood. *Wood Science & Tech.,* **1**:4, pp 289-303.

Ugulev, B.N. (1976). General Laws of Wood Deformation and Rheological Properties of Hardwood. *Wood Sci. & Tech.,* **10**, pp 169-181.

Urrey, S.A. (1953). *Solutions of Problems in Strength of Materials.* Pitman Press, Bath.

Westlund, R. (1976). A Qualitive Evaluation of Phenomenological Creep Rupture Theories. *Journ. Mech. Eng. Sci.,* **18**:4, pp 175-178.

Wood, L.W. (1951). Relation of strength of Wood to Duration of Load. *Forest Products Laboratory (Madison) Report No. 1916.*

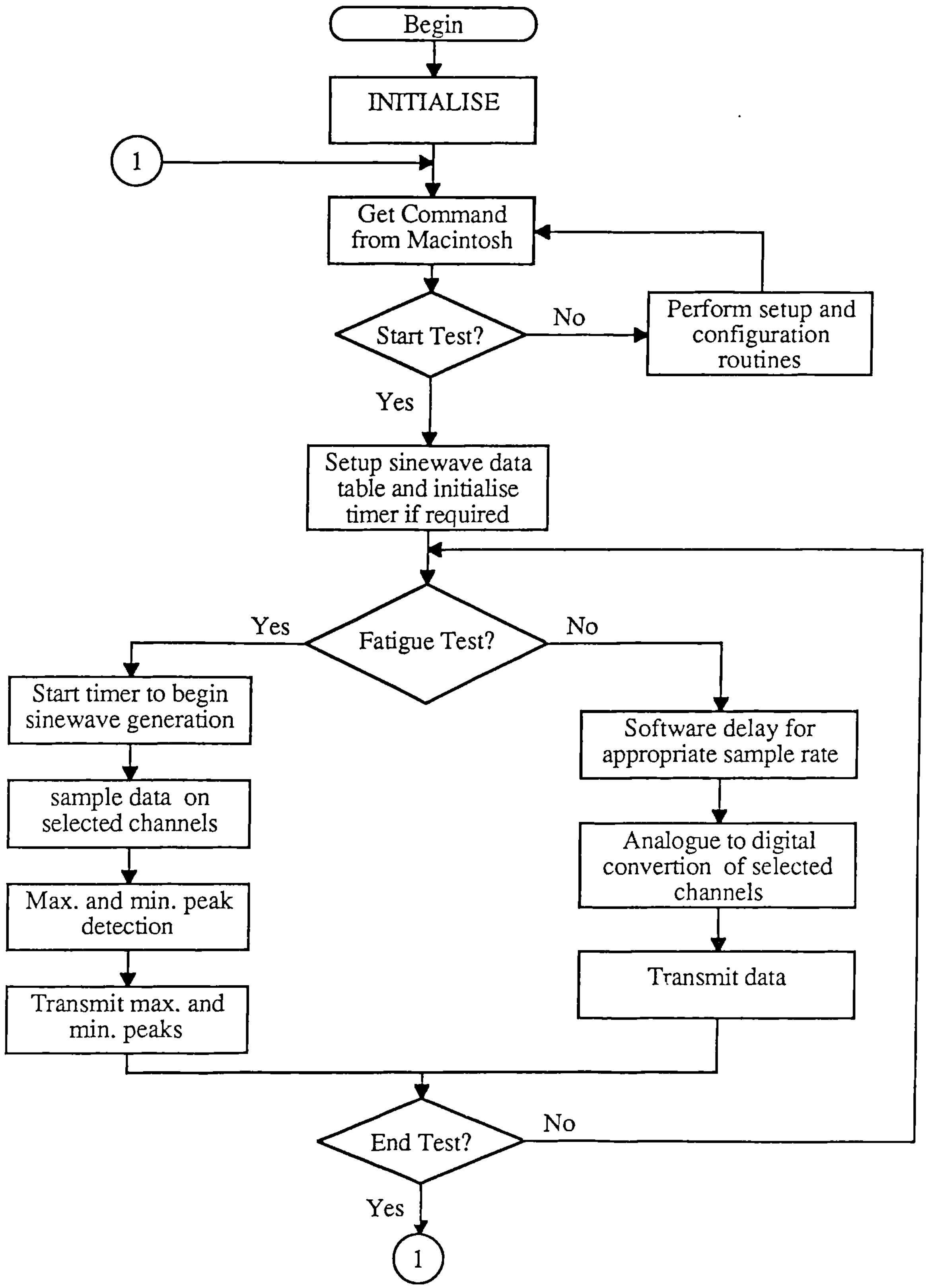
Wyatt, L.M. Lenel, U.R. and Moore, MA., (1983). Survey of Materials Suitable for use in MW Sized Aerogenerators. *Proc. of 5th BWEA Wind Energy Conf.*, pp. 235-244.

Zahn, J.J. (1977). Reliability-Based Design Procedures for Wood Structures. *For. Prod. Journ.*, **27**:3, pp. 21-28.

Zuteck, M.D. (1981). The Development and manufacture of Wood Composite Wind Turbine Rotors. *DOE/NASA Horizontal Axis Wind Turbine Workshop July 27, 1981.*

Appendix A

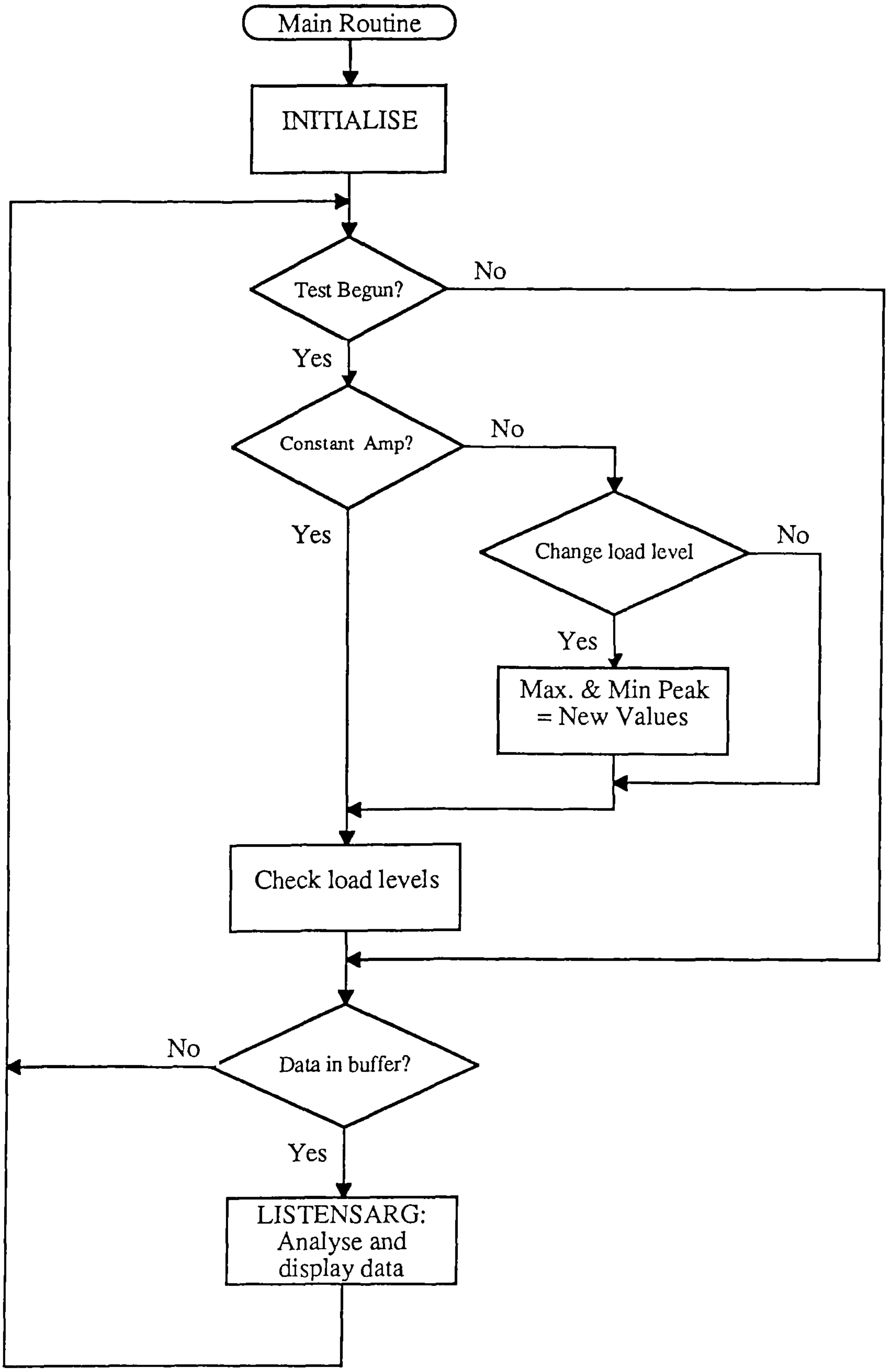
Flow Chart For SArGen Software

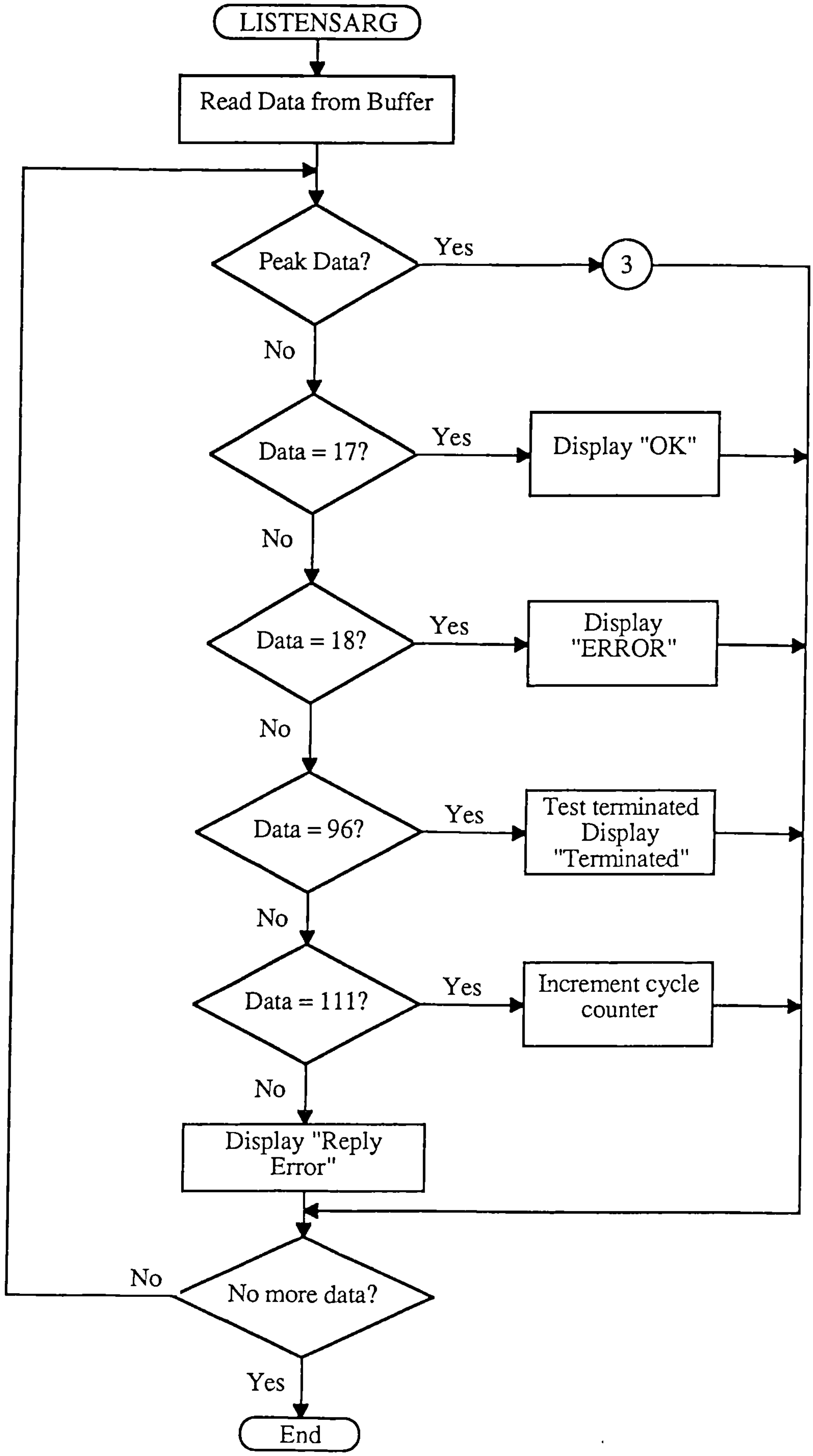


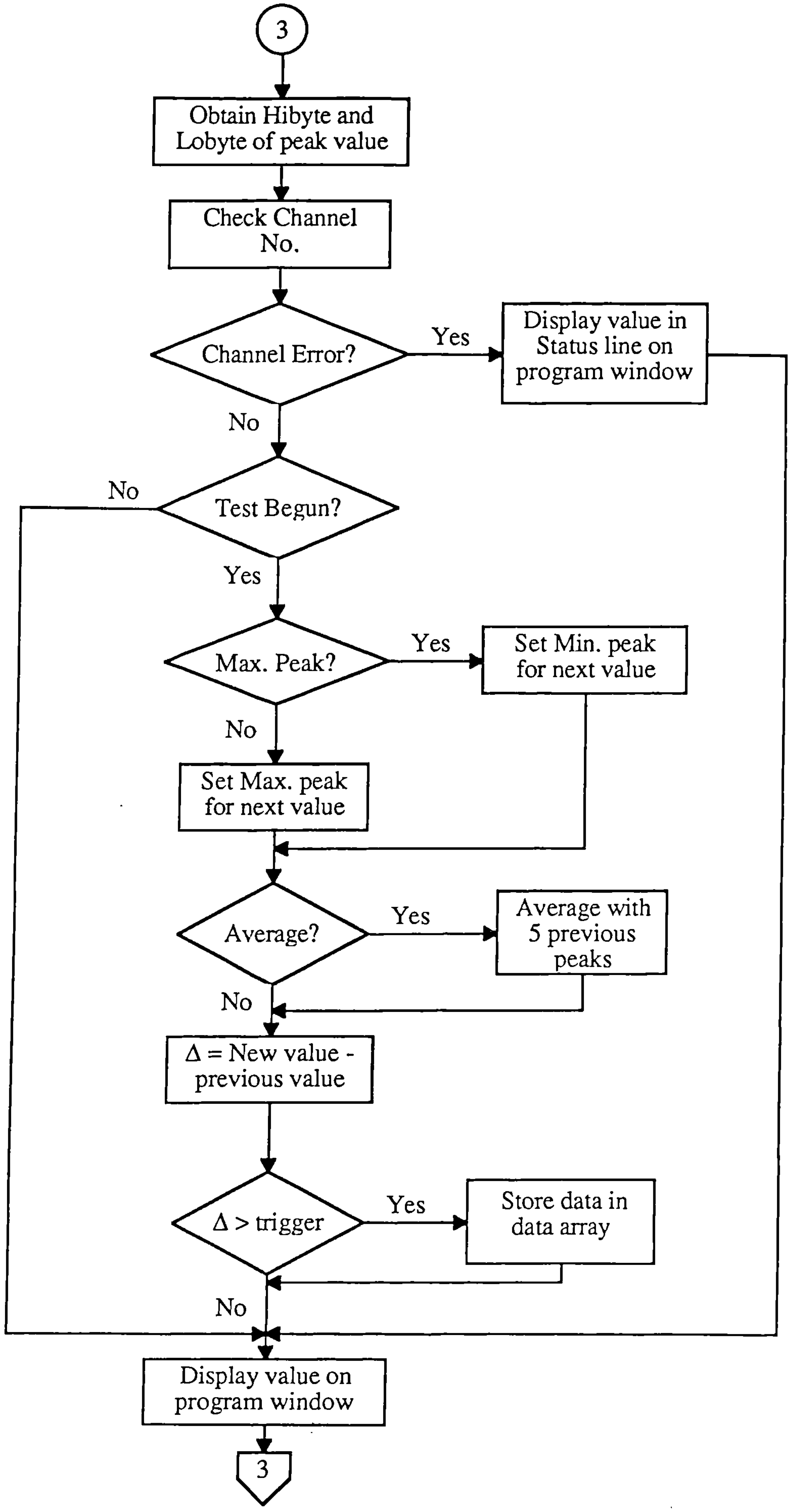
Flow chart showing structure of software in SArGen

Appendix B

Flow Chart For Macintosh Software


Main Routine
INITIALISE
Test Begun?
No
Yes
Constant Amp?
No
Yes
Change load level
No
Yes
Max. & Min Peak
= New Values
Check load levels
Data in buffer?
No
Yes
LISTENSARG:
Analyse and
display data


LISTENSARG
Read Data from Buffer
Peak Data?
Yes
3
No
Data = 17?
Yes
Display "OK"
No
Data = 18?
Yes
Display "ERROR"
No
Data = 96?
Yes
Test terminated Display "Terminated"
No
Data = 111?
Yes
Increment cycle counter
No
Display "Reply Error"
No
No more data?
Yes
End


3
Obtain Hibyte and Lobyte of peak value
Check Channel No.
Channel Error?
Yes
Display value in Status line on program window
No
Test Begun?
No
Yes
Max. Peak?
Yes
Set Min. peak for next value
No
Set Max. peak for next value
Average?
Yes
Average with 5 previous peaks
No
Δ = New value - previous value
Δ > trigger
Yes
Store data in data array
No
Display value on program window
3

APPENDIX C

SArGen SOFTWARE

```
C000    ACIACR     EQU  $C000
C000    ACIASR     EQU  $C000
C001    ACIATR     EQU  $C001
C001    ACIARR     EQU  $C001
C400    ADPIAORA   EQU  $C400
C400    ADPIADDA   EQU  $C400
C401    ADPIACRA   EQU  $C401
C402    ADPIAORB   EQU  $C402
C402    ADPIADDB   EQU  $C402
C403    ADPIACRB   EQU  $C403
C800    PTMCR1     EQU  $C800
C800    PTMCR3     EQU  $C800
C801    PTMCR2     EQU  $C801
C802    PTMMB1     EQU  $C802
C803    PTMTL1     EQU  $C803
C804    PTMMB2     EQU  $C804
C805    PTMTL2     EQU  $C805
C806    PTMMB3     EQU  $C806
C807    PTMTL3     EQU  $C807
C801    PTMSR      EQU  $C801
C802    PTMTC1     EQU  $C802
C803    PTMLB1     EQU  $C803
C804    PTMTC2     EQU  $C804
C805    PTMLB2     EQU  $C805
C806    PTMTC3     EQU  $C806
C807    PTMLB3     EQU  $C807
CC00    DAPIAORA   EQU  $CC00
CC00    DAPIADDA   EQU  $CC00
CC01    DAPIACRA   EQU  $CC01
CC02    DAPIAORB   EQU  $CC02
CC02    DAPIADDB   EQU  $CC02
CC03    DAPIACRB   EQU  $CC03
D000    SGA1       EQU  $D000
D001    SGA2       EQU  $D001
D002    SGA3       EQU  $D002
0000    TOPMEM     EQU  $0000
0000    ADMODE     EQU  TOPMEM
0001    DAMODE     EQU  TOPMEM+1
0002    NUMCYC     EQU  TOPMEM+2
0004    CYCCOUNT   EQU  TOPMEM+4
0008    BUFSIZE    EQU  TOPMEM+8
000A    BUFIN      EQU  TOPMEM+10
000C    BUFOUT     EQU  TOPMEM+12
000E    CHANNEL    EQU  TOPMEM+14
000F    XONOFF     EQU  TOPMEM+15
0010    ADRATE     EQU  TOPMEM+16
0012    MAXPK1     EQU  TOPMEM+18
0014    MINPK1     EQU  TOPMEM+20
0016    MAXPK2     EQU  MINPK1+2
0018    MINPK2     EQU  MAXPK2+2
001A    MAXPK3     EQU  MINPK2+2
001C    MINPK3     EQU  MAXPK3+2
001E    MAXPK4     EQU  MINPK3+2
0020    MINPK4     EQU  MAXPK4+2
0022    CRANGE     EQU  MINPK4+2
0024    CMEAN      EQU  CRANGE+2
0026    DAWAVADJ   EQU  CMEAN+2
```

```
           0027   TRANSCK   EQU  DAWAVADJ+1
           0028   MAX1MIN   EQU  TRANSCK+1
           0029   MAX2MIN   EQU  MAX1MIN+1
           002A   MAX3MIN   EQU  MAX2MIN+1
           002B   MAX4MIN   EQU  MAX3MIN+1
           002C   CMAX      EQU  MAX4MIN+1
           002E   CMIN      EQU  CMAX+2
           0030   WAVEADR   EQU  CMIN+2
           0032   TBLCOUNT  EQU  WAVEADR+2
           0034   INCDEC    EQU  TBLCOUNT+2
           0035   VAMPSIZE  EQU  INCDEC+1
           0037   TRIPPED   EQU  VAMPSIZE+2
           0038   NUMCHAN   EQU  TRIPPED+1
           0039   PROGLEN   EQU  NUMCHAN+1
           BE00   WTBLINC1  EQU  $BE00
           BEB4   WTBLINC2  EQU  $BEB4
           BD00   WTBLDEC1  EQU  $BD00
           BDB0   WTBLDEC2  EQU  $BDB0
           BE5A   INCMIDD   EQU  $BE5A
           BD58   DECMIDD   EQU  $BD58
           1000   BUFPTR1   EQU  $1000
           3000   WAVETBL   EQU  $3000
           003B   RATECT    EQU  PROGLEN+2
           003D   USERANGE  EQU  RATECT+2
           003F   WAVELVL   EQU  USERANGE+2
           0041   RESULT    EQU  WAVELVL+2
           0043   VARIAB    EQU  RESULT+2
           0045   WCOUNTER  EQU  VARIAB+2
           0046   PTR1      EQU  WCOUNTER+1
           0048   PTR2      EQU  PTR1+2
           004A   PTR3      EQU  PTR2+2
           004C   TRIGLVL   EQU  PTR3+2
           004E   CYCACK    EQU  TRIGLVL+2
           004F   SGAVAL    EQU  CYCACK+1
           0050   SGACUR    EQU  SGAVAL+1
      F000                  ORG  $F000
      F000 86 03            LDAA #03
      F002 B7 C000          STAA ACIACR
      F005 86 18            LDAA #%00011000
      F007 B7 C000          STAA ACIACR
      F00A 86 28            LDAA #%00101000
      F00C B7 C401          STAA ADPIACRA
      F00F C6 00            LDAB #00
      F011 F7 C400          STAB ADPIADDA
      F014 8A 04            ORAA #04
      F016 B7 C401          STAA ADPIACRA
      F019 86 30            LDAA #%00110000
      F01B B7 C403          STAA ADPIACRB
      F01E C6 F0            LDAB #%11110000
      F020 F7 C402          STAB ADPIADDB
      F023 8A 04            ORAA #04
      F025 B7 C403          STAA ADPIACRB
      F028 B6 C402          LDAA ADPIAORB
      F02B 84 0F            ANDA #$0F
      F02D B7 C402          STAA ADPIAORB
      F030 86 00            LDAA #%00000000
      F032 B7 CC01          STAA DAPIACRA
```

```
F035 C6 FF                LDAB #$FF
F037 F7 CC00              STAB DAPIADDA
F03A 8A 04                ORAA #04
F03C B7 CC01              STAA DAPIACRA
F03F F7 CC00              STAB DAPIAORA
F042 86 28                LDAA #%00101000
F044 B7 CC03              STAA DAPIACRB
F047 C6 FF                LDAB #$FF
F049 F7 CC02              STAB DAPIADDB
F04C 8A 04                ORAA #04
F04E B7 CC03              STAA DAPIACRB
F051 C6 F7                LDAB #$F7
F053 F7 CC02              STAB DAPIAORB
F056 8E BFFF              LDS  #$BFFF
F059 7E F072              JMP  FATPROG
F05C B6 C000 RECEIVE      LDAA ACIASR
F05F 84 01                ANDA #01
F061 27 F9                BEQ  RECEIVE
F063 B6 C001              LDAA ACIARR
F066 39                   RTS
F067 B6 C000 SEND         LDAA ACIASR
F06A 84 02                ANDA #02
F06C 27 F9                BEQ  SEND
F06E F7 C001              STAB ACIATR
F071 39                   RTS
F072 CE 1000 FATPROG      LDX  #BUFPTR1
F075 DF 0A                STX  BUFIN
F077 DF 0C                STX  BUFOUT
F079 CE 1000              LDX  #$1000
F07C DF 08                STX  BUFSIZE
F07E CE 3000              LDX  #WAVETBL
F081 DF 30                STX  WAVEADR
F083 CE 0000              LDX  #$0000
F086 DF 00                STX  ADMODE
F088 DF 02                STX  NUMCYC
F08A DF 04                STX  CYCCOUNT
F08C DF 06                STX  CYCCOUNT+2
F08E DF 0E                STX  CHANNEL
F090 DF 10                STX  ADRATE
F092 DF 14                STX  MINPK1
F094 DF 18                STX  MINPK2
F096 DF 1C                STX  MINPK3
F098 DF 20                STX  MINPK4
F09A DF 22                STX  CRANGE
F09C DF 24                STX  CMEAN
F09E DF 26                STX  DAWAVADJ
F0A0 DF 28                STX  MAX1MIN
F0A2 DF 2A                STX  MAX3MIN
F0A4 DF 37                STX  TRIPPED
F0A6 DF 4E                STX  CYCACK
F0A8 CE FFFF              LDX  #$FFFF
F0AB DF 12                STX  MAXPK1
F0AD DF 16                STX  MAXPK2
F0AF DF 1A                STX  MAXPK3
F0B1 DF 1E                STX  MAXPK4
F0B3 86 60                LDAA #$60
F0B5 97 4C                STAA TRIGLVL
```

```
F0B7 BD F05C CONFLOOP JSR  RECEIVE
F0BA 81 01             CMPA #$01
F0BC 26 03             BNE  OTHER
F0BE 7E F382           JMP  PROCST
F0C1 BD F0C7 OTHER     JSR  WHATOP
F0C4 7E F0B7           JMP  CONFLOOP
F0C7 81 11   WHATOP    CMPA #$11
F0C9 26 06             BNE  C1
F0CB BD F378           JSR  XONRCV
F0CE 7E F1B4           JMP  WHATFIN
F0D1 81 12   C1        CMPA #$12
F0D3 26 06             BNE  C2
F0D5 BD F37D           JSR  XOFFRCV
F0D8 7E F1B4           JMP  WHATFIN
F0DB 81 0A   C2        CMPA #$0A
F0DD 26 06             BNE  C3
F0DF BD F1DA           JSR  ADTRANS
F0E2 7E F1B4           JMP  WHATFIN
F0E5 81 0B   C3        CMPA #$0B
F0E7 26 06             BNE  C4
F0E9 BD F1E9           JSR  ADPKPK
F0EC 7E F1B4           JMP  WHATFIN
F0EF 81 13   C4        CMPA #$13
F0F1 26 06             BNE  C5
F0F3 BD F1CF           JSR  SAVAIL
F0F6 7E F1B4           JMP  WHATFIN
F0F9 81 0D   C5        CMPA #$0D
F0FB 26 06             BNE  C6
F0FD BD F1F3           JSR  DAAMPTBL
F100 7E F1B4           JMP  WHATFIN
F103 81 0E   C6        CMPA #$0E
F105 26 06             BNE  C7
F107 BD F211           JSR  DACONST
F10A 7E F1B4           JMP  WHATFIN
F10D 81 0F   C7        CMPA #$0F
F10F 26 06             BNE  C8
F111 BD F1D5           JSR  DAOFF
F114 7E F1B4           JMP  WHATFIN
F117 81 08   C8        CMPA #$08
F119 26 06             BNE  C9
F11B BD F244           JSR  DASETUP
F11E 7E F1B4           JMP  WHATFIN
F121 81 0C   C9        CMPA #$0C
F123 26 06             BNE  C10
F125 BD F23A           JSR  ADOFF
F128 7E F1B4           JMP  WHATFIN
F12B 16      C10       TAB
F12C C4 F0             ANDB #$F0
F12E C1 70             CMPB #$70
F130 26 03             BNE  C11
F132 BD F23F           JSR  CHSELECT
F135 81 21   C11       CMPA #$21
F137 26 06             BNE  C12
F139 BD E000           JSR  $E000
F13C 7E F1B4           JMP  WHATFIN
F13F 81 22   C12       CMPA #$22
F141 26 06             BNE  C13
```

```
F143 BD E002              JSR  $E002
F146 7E F1B4              JMP  WHATFIN
F149 81 23     C13        CMPA #$23
F14B 26 06                BNE  C14
F14D BD E005              JSR  $E005
F150 7E F1B4              JMP  WHATFIN
F153 81 24     C14        CMPA #$24
F155 26 06                BNE  C15
F157 BD E008              JSR  $E008
F15A 7E F1B4              JMP  WHATFIN
F15D 81 25     C15        CMPA #$25
F15F 26 06                BNE  C16
F161 BD E00B              JSR  $E00B
F164 7E F1B4              JMP  WHATFIN
F167 81 26     C16        CMPA #$26
F169 26 06                BNE  C17
F16B BD E00E              JSR  $E00E
F16E 7E F1B4              JMP  WHATFIN
F171 81 27     C17        CMPA #$27
F173 26 06                BNE  C18
F175 BD E011              JSR  $E011
F178 7E F1B4              JMP  WHATFIN
F17B 81 28     C18        CMPA #$28
F17D 26 06                BNE  C19
F17F BD E014              JSR  $E014
F182 7E F1B4              JMP  WHATFIN
F185 81 29     C19        CMPA #$29
F187 26 06                BNE  C20
F189 BD E017              JSR  $E017
F18C 7E F1B4              JMP  WHATFIN
F18F 81 2A     C20        CMPA #$2A
F191 26 06                BNE  C21
F193 BD E01A              JSR  $E01A
F196 7E F1B4              JMP  WHATFIN
F199 81 10     C21        CMPA #$10
F19B 26 06                BNE  C22
F19D BD F1B5              JSR  LOADPROG
F1A0 7E F1B4              JMP  WHATFIN
F1A3 81 14     C22        CMPA #$14
F1A5 26 06                BNE  C23
F1A7 BD 0100              JSR  $0100
F1AA 7E F1B4              JMP  WHATFIN
F1AD 81 15     C23        CMPA #$15
F1AF 26 03                BNE  WHATFIN
F1B1 BD F2A9              JSR  SGASET
F1B4 39        WHATFIN    RTS
F1B5 BD F05C LOADPROG     JSR  RECEIVE
F1B8 8B 01                ADDA #$01
F1BA 97 39                STAA PROGLEN
F1BC BD F05C              JSR  RECEIVE
F1BF 97 3A                STAA PROGLEN+1
F1C1 CE 0100              LDX  #$0100
F1C4 BD F05C GETPROG      JSR  RECEIVE
F1C7 A7 00                STAA 0,X
F1C9 08                   INX
F1CA 9C 39                CPX  PROGLEN
F1CC 26 F6                BNE  GETPROG
```

```
F1CE 39                   RTS
F1CF C6 11     SAVAIL     LDAB #$11
F1D1 BD F067              JSR  SEND
F1D4 39                   RTS
F1D5 C6 00     DAOFF      LDAB #$00
F1D7 D7 01                STAB DAMODE
F1D9 39                   RTS
F1DA BD F05C   ADTRANS    JSR  RECEIVE
F1DD 97 10                STAA ADRATE
F1DF BD F05C              JSR  RECEIVE
F1E2 97 11                STAA ADRATE+1
F1E4 C6 0F                LDAB #$0F
F1E6 D7 00                STAB ADMODE
F1E8 39                   RTS
F1E9 BD F05C   ADPKPK     JSR  RECEIVE
F1EC 97 4C                STAA TRIGLVL
F1EE C6 F0                LDAB #$F0
F1F0 D7 00                STAB ADMODE
F1F2 39                   RTS
F1F3 BD F05C   DAAMPTBL   JSR  RECEIVE
F1F6 8B 30                ADDA #$30
F1F8 97 35                STAA VAMPSIZE
F1FA BD F05C              JSR  RECEIVE
F1FD 97 36                STAA VAMPSIZE+1
F1FF CE 2FFE              LDX  #WAVETBL-2
F202 BD F05C   GETDATBL   JSR  RECEIVE
F205 A7 00                STAA 0,X
F207 08                   INX
F208 9C 35                CPX  VAMPSIZE
F20A 26 F6                BNE  GETDATBL
F20C C6 0F                LDAB #$0F
F20E D7 01                STAB DAMODE
F210 39                   RTS
F211 BD F05C   DACONST    JSR  RECEIVE
F214 B7 3000              STAA WAVETBL
F217 BD F05C              JSR  RECEIVE
F21A B7 3001              STAA WAVETBL+1
F21D BD F05C              JSR  RECEIVE
F220 B7 3002              STAA WAVETBL+2
F223 BD F05C              JSR  RECEIVE
F226 B7 3003              STAA WAVETBL+3
F229 BD F05C              JSR  RECEIVE
F22C B7 2FFE              STAA WAVETBL-2
F22F BD F05C              JSR  RECEIVE
F232 B7 2FFF              STAA WAVETBL-1
F235 C6 F0                LDAB #$F0
F237 D7 01                STAB DAMODE
F239 39                   RTS
F23A C6 00     ADOFF      LDAB #$00
F23C D7 00                STAB ADMODE
F23E 39                   RTS
F23F 84 0F     CHSELECT   ANDA #$0F
F241 97 0E                STAA CHANNEL
F243 39                   RTS
F244 B6 C403   DASETUP    LDAA ADPIACRB
F247 84 F7                ANDA #$F7      ! SAMPLE
F249 B7 C403              STAA ADPIACRB
```

```
F24C CE 2000             LDX  ##$2000
F24F B6 C000 DALOOP      LDAA ACIASR
F252 84 01               ANDA #01
F254 27 10               BEQ  DAS3
F256 F6 C001             LDAB ACIARR
F259 BD F05C             JSR  RECEIVE
F25C 81 50               CMPA #$50
F25E 27 44               BEQ  DAS1
F260 B7 CC00 DAS2        STAA DAPIAORA
F263 F7 CC02             STAB DAPIAORB
F266 09      DAS3        DEX
F267 26 E6               BNE  DALOOP
F269 B6 C403             LDAA ADPIACRB
F26C 8A 08               ORAA #$08        ! HOLD
F26E B7 C403             STAA ADPIACRB
F271 B6 C402             LDAA ADPIAORB
F274 84 0F               ANDA #$0F
F276 B7 C402             STAA ADPIAORB
F279 B6 C400             LDAA ADPIAORA
F27C C6 88               LDAB #$88
F27E BD F067             JSR  SEND
F281 CE 0050 WAIT        LDX  #0080
F284 09      W25         DEX
F285 26 FD               BNE  W25
F287 CE 0004             LDX  #$04
F28A F6 C402 DO4CH       LDAB ADPIAORB
F28D CB 10               ADDB #$10
F28F F7 C402             STAB ADPIAORB
F292 B6 C400             LDAA ADPIAORA
F295 36                  PSHA
F296 C0 10               SUBB #$10
F298 BD F067             JSR  SEND
F29B 33                  PULB
F29C BD F067             JSR  SEND
F29F 09                  DEX
F2A0 26 E8               BNE  DO4CH
F2A2 20 A0               BRA  DASETUP
F2A4 C1 50   DAS1        CMPB #$50
F2A6 26 B8               BNE  DAS2
F2A8 39                  RTS
F2A9 BD F05C SGASET      JSR  RECEIVE
F2AC 97 4F               STAA SGAVAL
F2AE 7F 0050             CLR  SGACUR
F2B1 D6 50   SGASET1     LDAB SGACUR
F2B3 F7 D000             STAB SGA1
F2B6 F6 C403             LDAB ADPIACRB
F2B9 C4 F7               ANDB #$F7
F2BB F7 C403             STAB ADPIACRB
F2BE CE 00FF             LDX  #$00FF
F2C1 09      SGAW1       DEX
F2C2 26 FD               BNE  SGAW1
F2C4 F6 C403             LDAB ADPIACRB
F2C7 CA 08               ORAB #$08
F2C9 F7 C403             STAB ADPIACRB
F2CC C6 1F               LDAB #$1F
F2CE F7 C402             STAB ADPIAORB
F2D1 F6 C400             LDAB ADPIAORA
```

```
00400 F2D4 CE 00FF              LDX  #$00FF
00401 F2D7 09          SGAW2    DEX
00402 F2D8 26 FD                BNE  SGAW2
00403 F2DA F6 C402              LDAB ADPIAORB
00404 F2DD B6 C400              LDAA ADPIAORA
00405 F2E0 57                   ASRB
00406 F2E1 46                   RORA
00407 F2E2 57                   ASRB
00408 F2E3 46                   RORA
00409 F2E4 57                   ASRB
00410 F2E5 46                   RORA
00411 F2E6 57                   ASRB
00412 F2E7 46                   RORA
00413 F2E8 91 4F                CMPA SGAVAL
00414 F2EA 23 05                BLS  SSET2
00415 F2EC 7C 0050              INC  SGACUR
00416 F2EF 26 C0                BNE  SGASET1
00417 F2F1 7F 0050 SSET2        CLR  SGACUR
00418 F2F4 D6 50   SGASET2      LDAB SGACUR
00419 F2F6 F7 D001              STAB SGA2
00420 F2F9 F6 C403              LDAB ADPIACRB
00421 F2FC C4 F7                ANDB #$F7
00422 F2FE F7 C403              STAB ADPIACRB
00423 F301 CE 00FF              LDX  #$00FF
00424 F304 09          SGAW3    DEX
00425 F305 26 FD                BNE  SGAW3
00426 F307 F6 C403              LDAB ADPIACRB
00427 F30A CA 08                ORAB #$08
00428 F30C F7 C403              STAB ADPIACRB
00429 F30F C6 2F                LDAB #$2F
00430 F311 F7 C402              STAB ADPIAORB
00431 F314 F6 C400              LDAB ADPIAORA
00432 F317 CE 00FF              LDX  #$00FF
00433 F31A 09          SGAW4    DEX
00434 F31B 26 FD                BNE  SGAW4
00435 F31D F6 C402              LDAB ADPIAORB
00436 F320 B6 C400              LDAA ADPIAORA
00437 F323 57                   ASRB
00438 F324 46                   RORA
00439 F325 57                   ASRB
00440 F326 46                   RORA
00441 F327 57                   ASRB
00442 F328 46                   RORA
00443 F329 57                   ASRB
00444 F32A 46                   RORA
00445 F32B 91 4F                CMPA SGAVAL
00446 F32D 23 05                BLS  SSET3
00447 F32F 7C 0050              INC  SGACUR
00448 F332 26 C0                BNE  SGASET2
00449 F334 7F 0050 SSET3        CLR  SGACUR
00450 F337 D6 50   SGASET3      LDAB SGACUR
00451 F339 F7 D002              STAB SGA3
00452 F33C F6 C403              LDAB ADPIACRB
00453 F33F C4 F7                ANDB #$F7
00454 F341 F7 C403              STAB ADPIACRB
00455 F344 CE 00FF              LDX  #$00FF
00456 F347 09          SGAW5    DEX
```

```
F348 26 FD              BNE  SGAW5
F34A F6 C403            LDAB ADPIACRB
F34D CA 08              ORAB #$08
F34F F7 C403            STAB ADPIACRB
F352 C6 3F              LDAB #$3F
F354 F7 C402            STAB ADPIAORB
F357 F6 C400            LDAB ADPIAORA
F35A CE 00FF            LDX  #$00FF
F35D 09         SGAW6   DEX
F35E 26 FD              BNE  SGAW6
F360 F6 C402            LDAB ADPIAORB
F363 B6 C400            LDAA ADPIAORA
F366 57                 ASRB
F367 46                 RORA
F368 57                 ASRB
F369 46                 RORA
F36A 57                 ASRB
F36B 46                 RORA
F36C 57                 ASRB
F36D 46                 RORA
F36E 91 4F              CMPA SGAVAL
F370 23 05              BLS  SGAFIN
F372 7C 0050            INC  SGACUR
F375 26 C0              BNE  SGASET3
F377 39         SGAFIN  RTS
F378 C6 00      XONRCV  LDAB #$00
F37A D7 0F              STAB XONOFF
F37C 39                 RTS
F37D C6 FF      XOFFRCV LDAB #$FF
F37F D7 0F              STAB XONOFF
F381 39                 RTS
F382 96 01      PROCST  LDAA DAMODE
F384 81 00              CMPA #$00
F386 26 03              BNE  SETPTM
F388 7E F450            JMP  ADCHECK
F38B C6 43      SETPTM  LDAB #$43
F38D F7 C801            STAB PTMCR2     ! SETUP CON
F390 BD F39B            JSR  DAINIT
F393 86 00              LDAA #0
F395 B7 C800            STAA PTMCR1
F398 7E F450            JMP  ADCHECK
F39B FE 2FFE    DAINIT  LDX  WAVETBL-2
F39E FF C804            STX  PTMMB2
F3A1 86 FF              LDAA #$FF
F3A3 97 26              STAA DAWAVADJ
F3A5 FE 3000            LDX  WAVETBL
F3A8 DF 2C              STX  CMAX
F3AA FE 3002            LDX  WAVETBL+2
F3AD DF 2E              STX  CMIN
F3AF BD F75A            JSR  DANEW
F3B2 DE 24              LDX  CMEAN
F3B4 8C 07FF            CPX  #$07FF
F3B7 2E 0D              BGT  STDEC
F3B9 7F 0034            CLR  INCDEC
F3BC 7F 0028            CLR  MAX1MIN
F3BF CE BE00            LDX  #WTBLINC1
F3C2 DF 32              STX  TBLCOUNT
```

```
F3C4 20 0B                   BRA  STCONT
F3C6 86 FF     STDEC         LDAA #$FF
F3C8 97 34                   STAA INCDEC
F3CA 97 28                   STAA MAX1MIN
F3CC CE BDB0                 LDX  #WTBLDEC2
F3CF DF 32                   STX  TBLCOUNT
F3D1 86 0F     STCONT        LDAA #$0F
F3D3 97 26                   STAA DAWAVADJ
F3D5 96 01                   LDAA DAMODE
F3D7 81 F0                   CMPA #$F0
F3D9 27 1E                   BEQ  ITSCONST
F3DB FE 3004                 LDX  WAVETBL+4
F3DE 27 05                   BEQ  CSHALF
F3E0 09                      DEX
F3E1 DF 02                   STX  NUMCYC
F3E3 20 14                   BRA  ITSCONST
F3E5 DE 30     CSHALF        LDX  WAVEADR
F3E7 08                      INX
F3E8 08                      INX
F3E9 08                      INX
F3EA 08                      INX
F3EB 08                      INX
F3EC 08                      INX
F3ED DF 30                   STX  WAVEADR
F3EF FE 3008                 LDX  WAVETBL+8
F3F2 DF 2E                   STX  CMIN
F3F4 FE 300A                 LDX  WAVETBL+10
F3F7 DF 02                   STX  NUMCYC
F3F9 BD F75A ITSCONST        JSR  DANEW
F3FC 0E                      CLI
F3FD 39                      RTS
F3FE B6 C000 LISTEN          LDAA ACIASR
F401 84 01                   ANDA #01
F403 27 49                   BEQ  LISFIN
F405 B6 C001                 LDAA ACIARR
F408 81 0D                   CMPA #$0D
F40A 26 09                   BNE  P3
F40C 0F                      SEI
F40D BD F1F3                 JSR  DAAMPTBL
F410 BD F39B                 JSR  DAINIT
F413 20 39                   BRA  LISFIN
F415 81 0E     P3            CMPA #$0E
F417 26 29                   BNE  P4
F419 7F 0037                 CLR  TRIPPED
F41C BD F211                 JSR  DACONST
F41F FE 3000                 LDX  WAVETBL
F422 DF 2C                   STX  CMAX
F424 FE 3002                 LDX  WAVETBL+2
F427 DF 2E                   STX  CMIN
F429 86 FF                   LDAA #$FF
F42B 97 26                   STAA DAWAVADJ
F42D BD F75A                 JSR  DANEW
F430 86 0F                   LDAA #$0F
F432 97 26                   STAA DAWAVADJ
F434 BD F75A                 JSR  DANEW
F437 FE 2FFE                 LDX  WAVETBL-2
F43A FF C804                 STX  PTMMB2
```

```
F43D 7F 0037           CLR  TRIPPED
F440 20 0C             BRA  LISFIN
F442 81 60    P4       CMPA #$60
F444 26 05             BNE  P5
F446 31                INS
F447 31                INS
F448 7E F83F           JMP  TERMITE
F44B BD F0C7 P5        JSR  WHATOP
F44E 39       LISFIN   RTS
F44F 01                NOP
F450 BD F3FE ADCHECK   JSR  LISTEN
F453 96 00             LDAA ADMODE
F455 81 F0             CMPA #$F0
F457 27 07             BEQ  ADSTPK
F459 81 0F             CMPA #$0F
F45B 26 F3             BNE  ADCHECK
F45D 7E F6A5           JMP  ADSTTRN
F460 BD F3FE ADSTPK    JSR  LISTEN
F463 B6 C403           LDAA ADPIACRB
F466 84 F7             ANDA #$F7
F468 B7 C403           STAA ADPIACRB
F46B 7D 004E           TST  CYCACK
F46E 27 11             BEQ  DANOCYC
F470 C6 6F             LDAB #$6F
F472 B6 C000 CYCSEND   LDAA ACIASR
F475 84 02             ANDA #02
F477 27 F9             BEQ  CYCSEND
F479 F7 C001           STAB ACIATR
F47C 7A 004E           DEC  CYCACK
F47F 26 F1             BNE  CYCSEND
F481 B6 C403 DANOCYC   LDAA ADPIACRB
F484 8A 08             ORAA #$08       ! HOLD
F486 B7 C403           STAA ADPIACRB
F489 B6 C402           LDAA ADPIAORB
F48C 84 0F             ANDA #$0F
F48E B7 C402           STAA ADPIAORB
F491 B6 C400           LDAA ADPIAORA
F494 86 05             LDAA #$05
F496 4A       WAITLP   DECA
F497 26 FD             BNE  WAITLP
F499 D6 0E             LDAB CHANNEL
F49B 54                LSRB
F49C 8B 10    NEXTCH   ADDA #$10
F49E 81 40             CMPA #$40
F4A0 26 14             BNE  CONT1
F4A2 F6 C402           LDAB ADPIAORB
F4A5 B6 C400           LDAA ADPIAORA
F4A8 BD F4D0           JSR  CHECKPK
F4AB 96 0B             LDAA BUFIN+1
F4AD 81 01             CMPA #$01
F4AF 23 AF             BLS  ADSTPK
F4B1 BD F72B           JSR  CLEARBUF
F4B4 20 AA             BRA  ADSTPK
F4B6 54       CONT1    LSRB
F4B7 24 E3             BCC  NEXTCH
F4B9 36                PSHA
F4BA 37                PSHB
```

```
F4BB F6 C402              LDAB ADPIAORB
F4BE 37                   PSHB
F4BF C4 0F                ANDB #$0F
F4C1 1B                   ABA
F4C2 B7 C402              STAA ADPIAORB
F4C5 33                   PULB
F4C6 B6 C400              LDAA ADPIAORA
F4C9 BD F4D0              JSR  CHECKPK
F4CC 33                   PULB
F4CD 32                   PULA
F4CE 20 CC                BRA  NEXTCH
F4D0 C1 0F     CHECKPK    CMPB #$0F
F4D2 23 16                BLS  ADCH1
F4D4 C1 1F                CMPB #$1F
F4D6 2E 03                BGT  CK3
F4D8 7E F55B              JMP  ADCH2
F4DB C1 2F     CK3        CMPB #$2F
F4DD 2E 03                BGT  CK4
F4DF 7E F5C9              JMP  ADCH3
F4E2 C1 3F     CK4        CMPB #$3F
F4E4 2E 03                BGT  FINISH
F4E6 7E F637              JMP  ADCH4
F4E9 39        FINISH     RTS
F4EA 7D 0028   ADCH1      TST  MAX1MIN
F4ED 27 35                BEQ  DO1MAX
F4EF D1 14                CMPB MINPK1
F4F1 2E 08                BGT  REMINPK1
F4F3 2D 0E                BLT  CK1TRIG
F4F5 91 15                CMPA MINPK1+1
F4F7 22 02                BHI  REMINPK1
F4F9 20 08                BRA  CK1TRIG
F4FB 97 15     REMINPK1   STAA MINPK1+1
F4FD D7 14                STAB MINPK1
F4FF 39                   RTS
F500 01                   NOP
F501 01                   NOP
F502 01                   NOP
F503 9B 4C     CK1TRIG    ADDA TRIGLVL
F505 C9 00                ADCB #0
F507 90 15                SUBA MINPK1+1
F509 D2 14                SBCB MINPK1
F50B 2A 16                BPL  MIN1FIN
F50D D6 14     SET1TRIG   LDAB MINPK1
F50F DE 0A                LDX  BUFIN
F511 E7 00                STAB 0,X
F513 08                   INX
F514 D6 15                LDAB MINPK1+1
F516 E7 00                STAB 0,X
F518 08                   INX
F519 DF 0A                STX  BUFIN
F51B 7F 0028              CLR  MAX1MIN
F51E CE 0000              LDX  #$0000
F521 DF 14                STX  MINPK1
F523 39        MIN1FIN    RTS
F524 D1 12     DO1MAX     CMPB MAXPK1
F526 2D 06                BLT  REMAXPK1
F528 2E 0C                BGT  CK1TMAX
```

```
00685 F52A 91 13           CMPA MAXPK1+1
00686 F52C 22 08           BHI  CK1TMAX
00687 F52E 97 13  REMAXPK1 STAA MAXPK1+1
00688 F530 D7 12           STAB MAXPK1
00689 F532 39              RTS
00690 F533 01              NOP
00691 F534 01              NOP
00692 F535 01              NOP
00693 F536 90 4C  CK1TMAX  SUBA TRIGLVL
00694 F538 C2 00           SBCB #0
00695 F53A 90 13           SUBA MAXPK1+1
00696 F53C 92 12           SBCA MAXPK1
00697 F53E 2B 1A           BMI  MAX1FIN
00698 F540 D6 12  SET1XTG  LDAB MAXPK1
00699 F542 DE 0A           LDX  BUFIN
00700 F544 E7 00           STAB 0,X
00701 F546 08              INX
00702 F547 D6 13           LDAB MAXPK1+1
00703 F549 E7 00           STAB 0,X
00704 F54B 08              INX
00705 F54C DF 0A           STX  BUFIN
00706 F54E 7F 0037         CLR  TRIPPED
00707 F551 CE 7FFF         LDX  #$7FFF
00708 F554 DF 12           STX  MAXPK1
00709 F556 C6 FF           LDAB #$FF
00710 F558 D7 28           STAB MAX1MIN
00711 F55A 39     MAX1FIN  RTS
00712 F55B 7D 0029 ADCH2   TST  MAX2MIN
00713 F55E 27 35           BEQ  DO2MAX
00714 F560 D1 18           CMPB MINPK2
00715 F562 2E 08           BGT  REMINPK2
00716 F564 2D 0E           BLT  CK2TRIG
00717 F566 91 19           CMPA MINPK2+1
00718 F568 22 02           BHI  REMINPK2
00719 F56A 20 08           BRA  CK2TRIG
00720 F56C 97 19  REMINPK2 STAA MINPK2+1
00721 F56E D7 18           STAB MINPK2
00722 F570 39              RTS
00723 F571 01              NOP
00724 F572 01              NOP
00725 F573 01              NOP
00726 F574 9B 4C  CK2TRIG  ADDA TRIGLVL
00727 F576 C9 00           ADCB #0
00728 F578 90 19           SUBA MINPK2+1
00729 F57A D2 18           SBCB MINPK2
00730 F57C 2A 16           BPL  MIN2FIN
00731 F57E D6 18  SET2TRIG LDAB MINPK2
00732 F580 DE 0A           LDX  BUFIN
00733 F582 E7 00           STAB 0,X
00734 F584 08              INX
00735 F585 D6 19           LDAB MINPK2+1
00736 F587 E7 00           STAB 0,X
00737 F589 08              INX
00738 F58A DF 0A           STX  BUFIN
00739 F58C 7F 0029         CLR  MAX2MIN
00740 F58F CE 0000         LDX  #$0000
00741 F592 DF 18           STX  MINPK2
```

```
F594 39         MIN2FIN  RTS
F595 D1 16      DO2MAX   CMPB MAXPK2
F597 2D 06               BLT  REMAXPK2
F599 2E 0C               BGT  CK2TMAX
F59B 91 17               CMPA MAXPK2+1
F59D 22 08               BHI  CK2TMAX
F59F 97 17      REMAXPK2 STAA MAXPK2+1
F5A1 D7 16               STAB MAXPK2
F5A3 39                  RTS
F5A4 01                  NOP
F5A5 01                  NOP
F5A6 01                  NOP
F5A7 90 4C      CK2TMAX  SUBA TRIGLVL
F5A9 C2 00               SBCB #0
F5AB 90 17               SUBA MAXPK2+1
F5AD D2 16               SBCB MAXPK2
F5AF 2B 17               BMI  MAX2FIN
F5B1 D6 16      SET2XTG  LDAB MAXPK2
F5B3 DE 0A               LDX  BUFIN
F5B5 E7 00               STAB 0,X
F5B7 08                  INX
F5B8 D6 17               LDAB MAXPK2+1
F5BA E7 00               STAB 0,X
F5BC 08                  INX
F5BD DF 0A               STX  BUFIN
F5BF C6 FF               LDAB #$FF
F5C1 D7 29               STAB MAX2MIN
F5C3 CE 7FFF             LDX  #$7FFF
F5C6 DF 16               STX  MAXPK2
F5C8 39         MAX2FIN  RTS
F5C9 7D 002A    ADCH3    TST  MAX3MIN
F5CC 27 35               BEQ  DO3MAX
F5CE D1 1C               CMPB MINPK3
F5D0 2E 08               BGT  REMINPK3
F5D2 2D 0E               BLT  CK3TRIG
F5D4 91 1D               CMPA MINPK3+1
F5D6 22 02               BHI  REMINPK3
F5D8 20 08               BRA  CK3TRIG
F5DA 97 1D      REMINPK3 STAA MINPK3+1
F5DC D7 1C               STAB MINPK3
F5DE 39                  RTS
F5DF 01                  NOP
F5E0 01                  NOP
F5E1 01                  NOP
F5E2 9B 4C      CK3TRIG  ADDA TRIGLVL
F5E4 C9 00               ADCB #0
F5E6 90 1D               SUBA MINPK3+1
F5E8 D2 1C               SBCB MINPK3
F5EA 2A 16               BPL  MIN3FIN
F5EC D6 1C      SET3TRIG LDAB MINPK3
F5EE DE 0A               LDX  BUFIN
F5F0 E7 00               STAB 0,X
F5F2 08                  INX
F5F3 D6 1D               LDAB MINPK3+1
F5F5 E7 00               STAB 0,X
F5F7 08                  INX
F5F8 DF 0A               STX  BUFIN
```

```
F5FA 7F 002A            CLR  MAX3MIN
F5FD CE 0000            LDX  #$0000
F600 DF 1C              STX  MINPK3
F602 39       MIN3FIN   RTS
F603 D1 1A    DO3MAX    CMPB MAXPK3
F605 2D 06              BLT  REMAXPK3
F607 2E 0C              BGT  CK3TMAX
F609 91 1B              CMPA MAXPK3+1
F60B 22 08              BHI  CK3TMAX
F60D 97 1B    REMAXPK3  STAA MAXPK3+1
F60F D7 1A              STAB MAXPK3
F611 39                 RTS
F612 01                 NOP
F613 01                 NOP
F614 01                 NOP
F615 90 4C    CK3TMAX   SUBA TRIGLVL
F617 C2 00              SBCB #0
F619 90 1B              SUBA MAXPK3+1
F61B D2 1A              SBCB MAXPK3
F61D 2B 17              BMI  MAX3FIN
F61F D6 1A    SET3XTG   LDAB MAXPK3
F621 DE 0A              LDX  BUFIN
F623 E7 00              STAB 0,X
F625 08                 INX
F626 D6 1B              LDAB MAXPK3+1
F628 E7 00              STAB 0,X
F62A 08                 INX
F62B DF 0A              STX  BUFIN
F62D C6 FF              LDAB #$FF
F62F D7 2A              STAB MAX3MIN
F631 CE 7FFF            LDX  #$7FFF
F634 DF 1A              STX  MAXPK3
F636 39       MAX3FIN   RTS
F637 7D 002B  ADCH4     TST  MAX4MIN
F63A 27 35              BEQ  DO4MAX
F63C D1 20              CMPB MINPK4
F63E 2E 08              BGT  REMINPK4
F640 2D 0E              BLT  CK4TRIG
F642 91 21              CMPA MINPK4+1
F644 22 02              BHI  REMINPK4
F646 20 08              BRA  CK4TRIG
F648 97 21    REMINPK4  STAA MINPK4+1
F64A D7 20              STAB MINPK4
F64C 39                 RTS
F64D 01                 NOP
F64E 01                 NOP
F64F 01                 NOP
F650 9B 4C    CK4TRIG   ADDA TRIGLVL
F652 C9 00              ADCB #0
F654 90 21              SUBA MINPK4+1
F656 D2 20              SBCB MINPK4
F658 2A 16              BPL  MIN4FIN
F65A D6 20    SET4TRIG  LDAB MINPK4
F65C DE 0A              LDX  BUFIN
F65E E7 00              STAB 0,X
F660 08                 INX
F661 D6 21              LDAB MINPK4+1
```

```
F663 E7 00            STAB 0,X
F665 08               INX
F666 DF 0A            STX  BUFIN
F668 7F 002B          CLR  MAX4MIN
F66B CE 0000          LDX  #$0000
F66E DF 20            STX  MINPK4
F670 39      MIN4FIN  RTS
F671 D1 1E   DO4MAX   CMPB MAXPK4
F673 2D 06            BLT  REMAXPK4
F675 2E 0C            BGT  CK4TMAX
F677 91 1F            CMPA MAXPK4+1
F679 22 08            BHI  CK4TMAX
F67B 97 1F   REMAXPK4 STAA MAXPK4+1
F67D D7 1E            STAB MAXPK4
F67F 39               RTS
F680 01               NOP
F681 01               NOP
F682 01               NOP
F683 90 4C   CK4TMAX  SUBA TRIGLVL
F685 C2 00            SBCB #0
F687 90 1F            SUBA MAXPK4+1
F689 D2 1E            SBCB MAXPK4
F68B 2B 17            BMI  MAX4FIN
F68D D6 1E   SET4XTG  LDAB MAXPK4
F68F DE 0A            LDX  BUFIN
F691 E7 00            STAB 0,X
F693 08               INX
F694 D6 1F            LDAB MAXPK4+1
F696 E7 00            STAB 0,X
F698 08               INX
F699 DF 0A            STX  BUFIN
F69B C6 FF            LDAB #$FF
F69D D7 2B            STAB MAX4MIN
F69F CE 7FFF          LDX  #$7FFF
F6A2 DF 1E            STX  MAXPK4
F6A4 39      MAX4FIN  RTS
F6A5 C6 00   ADSTTRN  LDAB #0
F6A7 CE 0004          LDX  #4
F6AA 96 0E            LDAA CHANNEL
F6AC 44      TRNCH1   LSRA
F6AD 24 02            BCC  TRNCH2
F6AF CB 02            ADDB #$02
F6B1 09      TRNCH2   DEX
F6B2 26 F8            BNE  TRNCH1
F6B4 CA 80            ORAB #$80
F6B6 D7 38            STAB NUMCHAN
F6B8 B6 C403 ADTRNLP  LDAA ADPIACRB
F6BB 84 F7            ANDA #$F7
F6BD B7 C403          STAA ADPIACRB
F6C0 DE 10            LDX  ADRATE
F6C2 DF 3B   CKTRN    STX  RATECT
F6C4 BD F3FE          JSR  LISTEN
F6C7 DE 3B            LDX  RATECT
F6C9 09               DEX
F6CA 26 F6            BNE  CKTRN
F6CC B6 C403 ADS3     LDAA ADPIACRB
F6CF 8A 08            ORAA #$08
```

```
00913 F6D1 B7 C403          STAA ADPIACRB
00914 F6D4 B6 C402          LDAA ADPIAORB
00915 F6D7 84 0F            ANDA #$0F
00916 F6D9 B7 C402          STAA ADPIAORB
00917 F6DC B6 C400          LDAA ADPIAORA
00918 F6DF D6 38            LDAB NUMCHAN
00919 F6E1 BD F067          JSR  SEND
00920 F6E4 CE 000A          LDX  #10
00921 F6E7 09         W2    DEX
00922 F6E8 26 FD            BNE  W2
00923 F6EA CE 0004          LDX  #$04
00924 F6ED 86 00            LDAA #0
00925 F6EF D6 0E            LDAB CHANNEL
00926 F6F1 54               LSRB
00927 F6F2 8B 10      AD4CH ADDA #$10
00928 F6F4 54               LSRB
00929 F6F5 24 1A            BCC  TRN3
00930 F6F7 36               PSHA
00931 F6F8 37               PSHB
00932 F6F9 F6 C402          LDAB ADPIAORB
00933 F6FC 37               PSHB
00934 F6FD C4 0F            ANDB #$0F
00935 F6FF 1B               ABA
00936 F700 B7 C402          STAA ADPIAORB
00937 F703 33               PULB
00938 F704 B6 C400          LDAA ADPIAORA
00939 F707 36               PSHA
00940 F708 BD F067          JSR  SEND
00941 F70B 33               PULB
00942 F70C BD F067          JSR  SEND
00943 F70F 33               PULB
00944 F710 32               PULA
00945 F711 09         TRN3  DEX
00946 F712 26 DE            BNE  AD4CH
00947 F714 F6 C402          LDAB ADPIAORB
00948 F717 BD F067          JSR  SEND
00949 F71A F6 C400          LDAB ADPIAORA
00950 F71D BD F067          JSR  SEND
00951 F720 B6 C403          LDAA ADPIACRB
00952 F723 84 F7            ANDA #$F7
00953 F725 B7 C403          STAA ADPIACRB
00954 F728 7E F6B8          JMP  ADTRNLP
00955 F72B 7D 000F CLEARBUF TST  XONOFF
00956 F72E 26 29            BNE  CBFIN
00957 F730 90 0D            SUBA BUFOUT+1
00958 F732 97 27            STAA TRANSCK
00959 F734 8A 80            ORAA #$80
00960 F736 B7 C001          STAA ACIATR
00961 F739 DE 0C            LDX  BUFOUT
00962 F73B 86 70            LDAA #$70
00963 F73D 4A         CB2   DECA
00964 F73E 26 FD            BNE  CB2
00965 F740 A6 00            LDAA 0,X
00966 F742 F6 C000    CB1   LDAB ACIASR
00967 F745 C4 02            ANDB #$02
00968 F747 27 F9            BEQ  CB1
00969 F749 B7 C001          STAA ACIATR
```

```
F74C 08                  INX
F74D A6 00               LDAA 0,X
F74F 7A 0027             DEC  TRANSCK
F752 26 EE               BNE  CB1
F754 CE 1000             LDX  #BUFPTR1
F757 DF 0A               STX  BUFIN
F759 39         CBFIN    RTS
F75A 96 2F      DANEW    LDAA CMIN+1
F75C D6 2E               LDAB CMIN
F75E 90 2D               SUBA CMAX+1
F760 D2 2C               SBCB CMAX
F762 54                  LSRB
F763 46                  RORA
F764 97 3E               STAA USERANGE+1
F766 D7 3D               STAB USERANGE
F768 9B 2D               ADDA CMAX+1
F76A D9 2C               ADCB CMAX
F76C 97 25               STAA CMEAN+1
F76E D7 24               STAB CMEAN
F770 96 3E               LDAA USERANGE+1
F772 D6 3D               LDAB USERANGE
F774 48                  ASLA
F775 59                  ROLB
F776 48                  ASLA
F777 59                  ROLB
F778 48                  ASLA
F779 59                  ROLB
F77A 44                  LSRA
F77B 44                  LSRA
F77C 44                  LSRA
F77D 97 3E               STAA USERANGE+1
F77F 97 23               STAA CRANGE+1
F781 D7 3D               STAB USERANGE
F783 D7 22               STAB CRANGE
F785 CE FC56             LDX  #QUARTERW+86
F788 DF 46               STX  PTR1
F78A 86 2C               LDAA #44
F78C 97 45               STAA WCOUNTER
F78E 96 26               LDAA DAWAVADJ
F790 81 0F               CMPA #$0F
F792 27 25               BEQ  RSDEC
F794 CE BE00    RSINC    LDX  #WTBLINC1
F797 96 2F               LDAA CMIN+1
F799 A7 01               STAA 1,X
F79B D6 2E               LDAB CMIN
F79D E7 00               STAB 0,X
F79F 08                  INX
F7A0 08                  INX
F7A1 DF 4A               STX  PTR3
F7A3 CE BEB4             LDX  #WTBLINC2
F7A6 96 2D               LDAA CMAX+1
F7A8 A7 01               STAA 1,X
F7AA D6 2C               LDAB CMAX
F7AC E7 00               STAB 0,X
F7AE 09                  DEX
F7AF 09                  DEX
F7B0 DF 48               STX  PTR2
```

```
F7B2 DE 24               LDX  CMEAN
F7B4 FF BE5A             STX  INCMIDD
F7B7 20 0F               BRA  WAVESET
F7B9 CE BD00 RSDEC       LDX  #WTBLDEC1
F7BC DF 4A               STX  PTR3
F7BE CE BDB0             LDX  #WTBLDEC2
F7C1 DF 48               STX  PTR2
F7C3 DE 24               LDX  CMEAN
F7C5 FF BD58             STX  DECMIDD
F7C8 DE 22   WAVESET     LDX  CRANGE
F7CA DF 3D               STX  USERANGE
F7CC DE 46               LDX  PTR1
F7CE E6 01               LDAB 1,X
F7D0 D7 40               STAB WAVELVL+1
F7D2 E6 00               LDAB 0,X
F7D4 09                  DEX
F7D5 09                  DEX
F7D6 DF 46               STX  PTR1
F7D8 CE 0005             LDX  #5
F7DB 4F                  CLRA
F7DC 7F 0043             CLR  VARIAB
F7DF 20 04               BRA  MULST
F7E1 78 0040 MULLP1      ASL  WAVELVL+1
F7E4 59                  ROLB
F7E5 76 003E MULST       ROR  USERANGE+1
F7E8 24 01               BCC  MULLP2
F7EA 1B                  ABA
F7EB 09      MULLP2      DEX
F7EC 26 F3               BNE  MULLP1
F7EE D7 44               STAB VARIAB+1
F7F0 5F                  CLRB
F7F1 CE 0006             LDX  #6
F7F4 78 0040 MULLP3      ASL  WAVELVL+1
F7F7 79 0044             ROL  VARIAB+1
F7FA 79 0043             ROL  VARIAB
F7FD 76 003D             ROR  USERANGE
F800 24 04               BCC  MULLP4
F802 9B 44               ADDA VARIAB+1
F804 D9 43               ADCB VARIAB
F806 09      MULLP4      DEX
F807 26 EB               BNE  MULLP3
F809 54                  LSRB
F80A 46                  RORA
F80B 54                  LSRB
F80C 46                  RORA
F80D 54                  LSRB
F80E 46                  RORA
F80F 97 42               STAA RESULT+1
F811 D7 41               STAB RESULT
F813 9B 25               ADDA CMEAN+1
F815 D9 24               ADCB CMEAN
F817 DE 4A               LDX  PTR3
F819 A7 01               STAA 1,X
F81B E7 00               STAB 0,X
F81D 08                  INX
F81E 08                  INX
F81F DF 4A               STX  PTR3
```

```
F821 96 25              LDAA CMEAN+1
F823 D6 24              LDAB CMEAN
F825 90 42              SUBA RESULT+1
F827 D2 41              SBCB RESULT
F829 DE 48              LDX  PTR2
F82B A7 01              STAA 1,X
F82D E7 00              STAB 0,X
F82F 09                 DEX
F830 09                 DEX
F831 DF 48              STX  PTR2
F833 7A 0045            DEC  WCOUNTER
F836 27 03              BEQ  DANFIN
F838 7E F7C8            JMP  WAVESET
F83B 7F 0026 DANFIN     CLR  DAWAVADJ
F83E 39                 RTS
F83F 86 01   TERMITE    LDAA #$01
F841 B7 C800            STAA PTMCR1
F844 0F                 SEI
F845 86 FF              LDAA #$FF
F847 B7 CC00            STAA DAPIAORA
F84A 86 F7              LDAA #$F7
F84C B7 CC02            STAA DAPIAORB
F84F DE 0A              LDX  BUFIN
F851 8C 1000            CPX  #BUFPTR1
F854 27 05              BEQ  TERM1
F856 96 0B              LDAA BUFIN+1
F858 BD F72B            JSR  CLEARBUF
F85B 7D 0001 TERM1      TST  DAMODE
F85E 27 05              BEQ  THATSIT
F860 C6 60              LDAB #$60
F862 BD F067            JSR  SEND
F865 7E F0B7 THATSIT    JMP  CONFLOOP
F868 CE 0070 DELAY      LDX  #$70
F86B 09      DEL1       DEX
F86C 26 FD              BNE  DEL1
F86E 39                 RTS
FA00                    ORG  $FA00
FA00 7E F000            JMP  $F000
FA03 B6 C801            LDAA PTMSR
FA06 B6 C804            LDAA PTMTC2
FA09 7D 0034            TST  INCDEC
FA0C 27 59              BEQ  WINC
FA0E DE 32   WDEC       LDX  TBLCOUNT
FA10 A6 01              LDAA 1,X
FA12 E6 00              LDAB 0,X
FA14 B7 CC00            STAA DAPIAORA
FA17 F7 CC02            STAB DAPIAORB
FA1A 09                 DEX
FA1B 09                 DEX
FA1C DF 32              STX  TBLCOUNT
FA1E 8C BD00            CPX  #WTBLDEC1
FA21 26 43              BNE  IRQDFIN
FA23 CE BE00            LDX  #WTBLINC1
FA26 DF 32              STX  TBLCOUNT
FA28 86 00              LDAA #$00
FA2A 97 34              STAA INCDEC
FA2C 7C 004E            INC  CYCACK
```

```
FA2F 96 01       IRQCONT2 LDAA DAMODE
FA31 81 F0                CMPA #$F0
FA33 27 31                BEQ  IRQDFIN
FA35 DE 02                LDX  NUMCYC
FA37 27 04                BEQ  CWHALF
FA39 09                   DEX
FA3A DF 02                STX  NUMCYC
FA3C 3B                   RTI
FA3D DE 30       CWHALF   LDX  WAVEADR
FA3F 08                   INX
FA40 08                   INX
FA41 08                   INX
FA42 08                   INX
FA43 08                   INX
FA44 08                   INX
FA45 9C 35                CPX  VAMPSIZE
FA47 22 04                BHI  TBLTOP
FA49 DF 30                STX  WAVEADR
FA4B 20 05                BRA  IRQCONT1
FA4D CE 3000     TBLTOP   LDX  #WAVETBL
FA50 DF 30                STX  WAVEADR
FA52 A6 03       IRQCONT1 LDAA 3,X
FA54 E6 02                LDAB 2,X
FA56 97 2F                STAA CMIN+1
FA58 D7 2E                STAB CMIN
FA5A A6 05                LDAA 5,X
FA5C E6 04                LDAB 4,X
FA5E 97 03                STAA NUMCYC+1
FA60 D7 02                STAB NUMCYC
FA62 86 0F                LDAA #$0F
FA64 97 26                STAA DAWAVADJ
FA66 3B          IRQDFIN  RTI
FA67 DE 32       WINC     LDX  TBLCOUNT
FA69 A6 01                LDAA 1,X
FA6B E6 00                LDAB 0,X
FA6D B7 CC00              STAA DAPIAORA
FA70 F7 CC02              STAB DAPIAORB
FA73 08                   INX
FA74 08                   INX
FA75 DF 32                STX  TBLCOUNT
FA77 8C BEB6              CPX  #WTBLINC2+2
FA7A 26 2F                BNE  IRQFIN
FA7C CE BDB0              LDX  #WTBLDEC2
FA7F DF 32                STX  TBLCOUNT
FA81 86 FF                LDAA #$FF
FA83 97 34                STAA INCDEC
FA85 91 37                CMPA TRIPPED
FA87 26 08                BNE  NOTTRIP
FA89 7D 0000              TST  ADMODE
FA8C 27 03                BEQ  NOTTRIP
FA8E 7E F83F              JMP  TERMITE
FA91 97 37       NOTTRIP  STAA TRIPPED
FA93 96 01                LDAA DAMODE
FA95 81 F0                CMPA #$F0
FA97 27 12                BEQ  IRQFIN
FA99 DE 02                LDX  NUMCYC
FA9B 26 0E                BNE  IRQFIN
```

```
FA9D DE 30              LDX   WAVEADR
FA9F A6 01              LDAA  1,X
FAA1 E6 00              LDAB  0,X
FAA3 97 2D              STAA  CMAX+1
FAA5 D7 2C              STAB  CMAX
FAA7 86 FF              LDAA  #$FF
FAA9 97 26              STAA  DAWAVADJ
FAAB 3B        IRQFIN   RTI
FC00                    ORG   $FC00
FC00 0047      QUARTERW FDB   $0047
FC02 008F               FDB   $008F
FC04 00D6               FDB   $00D6
FC06 011D               FDB   $011D
FC08 0164               FDB   $0164
FC0A 01AA               FDB   $01AA
FC0C 01EF               FDB   $01EF
FC0E 0234               FDB   $0234
FC10 0279               FDB   $0279
FC12 02BC               FDB   $02BC
FC14 02FF               FDB   $02FF
FC16 0341               FDB   $0341
FC18 0382               FDB   $0382
FC1A 03C1               FDB   $03C1
FC1C 0400               FDB   $0400
FC1E 043D               FDB   $043D
FC20 0479               FDB   $0479
FC22 04B4               FDB   $04B4
FC24 04ED               FDB   $04ED
FC26 0524               FDB   $0524
FC28 055A               FDB   $055A
FC2A 058F               FDB   $058F
FC2C 05C1               FDB   $05C1
FC2E 05F2               FDB   $05F2
FC30 0621               FDB   $0621
FC32 064E               FDB   $064E
FC34 0679               FDB   $0679
FC36 06A2               FDB   $06A2
FC38 06C9               FDB   $06C9
FC3A 06ED               FDB   $06ED
FC3C 0710               FDB   $0710
FC3E 0731               FDB   $0731
FC40 074F               FDB   $074F
FC42 076B               FDB   $076B
FC44 0784               FDB   $0784
FC46 079C               FDB   $079C
FC48 07B1               FDB   $07B1
FC4A 07C3               FDB   $07C3
FC4C 07D3               FDB   $07D3
FC4E 07E1               FDB   $07E1
FC50 07EC               FDB   $07EC
FC52 07F5               FDB   $07F5
FC54 07FB               FDB   $07FB
FC56 07FF               FDB   $07FF
FFF8                    ORG   $FFF8
FFF8 FA03               FDB   $FA03
FFFA FA00               FDB   $FA00
FFFC F000               FDB   $F000
FFFE F000               FDB   $F000
                        END
```

Appendix D

Fatigue Testing

```
PROGRAM FatigueTesting;

USES  {$U-}
          {$U  OBJ/Memtypes    } Memtypes,
          {$U  OBJ/QuickDraw  } QuickDraw,
          {$U  OBJ/OSintf      } OSintf,
          {$U  OBJ/PackIntf    } PackIntf,
          {$U  OBJ/SaneLib     } Sane,
          {$U  Sargen/drvrimpl     }  drvrimpl,
          {$U  OBJ/TOOLintf    } TOOLintf;

CONST
   lastMenu  = 4;          { number of menus }
   appleMenu = 1;              { menu ID for desk accessory menu }
   genmenu = 2;            { menu ID for general menu }
   Sendmenu = 3;
   Smoothmenu = 4;

TYPE
   GenRecord = Record
              GSArG       : SArG;
              fname       : STR255;
              dataFname : STR255;
              vrefnum     : INTEGER;
              channel         : INTEGER;
              StoreData  : BOOLEAN;
              SArGAvail  : BOOLEAN;
              progX       : BOOLEAN;
              PReady      : BOOLEAN;
              Start       : BOOLEAN;
              Quit        : BOOLEAN;
          END;

   cycdat = PACKED Array [0..1000] OF Longint;
   cycptr = ^cycdat;
   cychdl = ^cycptr;

   PkStc = Record
             Maxpk   : Integer;
             Minpk   : Integer;
          End;

   PkData = PACKED Array [0..1000] OF PkStc;
   pkptr   = ^Pkdata;
   pkhdl   = ^pkptr;
```

```
AmpliRec = Record
                VMaxpk      : Integer;
                VMinpk      : Integer;
                VRate       : Integer;
                Ncycle      : LongInt;
             End;
AmpliDat = Packed Array [1..200] OF AmpliRec;

VAR
  myMenus       : ARRAY [1..lastMenu] OF MenuHandle;
  progRec           : GenRecord;
  mainWindow,
  tempwindow    :  WindowPtr;
  myEvent       : EventRecord;
  rn            : OSErr;
  DAMode,
  genint,
  refnum,
  code          : Integer;
  genlong,
  numbytes      : Longint;
  dragrect      : Rect;
  genrect           : Rect;
  genstr            : STR255;
  DALevelCtl        : ControlHandle;
  rate,
  Maxpk,
  MinPk,
  CurMaxpk,
  CurMinpk      : Integer;
  Lnumcyc       : LongInt;
  cycdata       : LongInt;
  D1Str         :  Str255;
  DVal          : ARRAY [1..9] OF Integer;
  DBool         : ARRAY [1..10] OF BOOLEAN;
  chselected        : ARRAY [1..4] OF BOOLEAN;
  StorData      : ARRAY [1..4] OF BOOLEAN;
  DataMax       : ARRAY [1..4] OF BOOLEAN;
  DataCount     : ARRAY [1..4] OF Integer;
  LMax          : ARRAY [1..4] OF Integer;
  LMin          : ARRAY [1..4] OF Integer;
  SMax          : ARRAY [1..4] OF Integer;
  SMin          : ARRAY [1..4] OF Integer;
  ChMinaver     : ARRAY [1..4,1..5] OF Integer;
  ChMaxaver         : ARRAY [1..4,1..5] OF Integer;
  avMinpt       : ARRAY [1..4] OF Integer;
```

```
avMaxpt       : ARRAY [1..4] OF Integer;
NumMinpt      : ARRAY [1..4] OF Integer;
NumMaxpt      : ARRAY [1..4] OF Integer;
Maxtotal,
Mintotal      : ARRAY [1..4] OF Integer;
Avselected        : ARRAY [1..4] OF BOOLEAN;
pth           : ARRAY [1..8] OF INTEGER;
ptv           : ARRAY [1..8] OF INTEGER;
checklev      : Integer;
Cyc1datHdl    : cychdl;
pk1dathdl     : pkHdl;
Cyc2datHdl    : cychdl;
pk2dathdl     : pkHdl;
Cyc3datHdl    : cychdl;
pk3dathdl     : pkHdl;
Cyc4datHdl    : cychdl;
pk4dathdl     : pkHdl;
VANumlev      : Integer;
VACurlev      : Integer;
VALastcyc     : LongInt;
VAmpTbl       : AmpliDat;
genchar           : CHAR;
InLevCtl,
DumClr,
dumBool       : BOOLEAN;
dataConst     : EXTENDED;
Data1F            : DecForm;
triglev       : Integer;


Procedure Anyerr(rn: Integer);
VAR   astr: STR255;
      resA: Integer;

Begin
   NumtoString(rn,astr);
   paramtext(astr,'','','');
   resA:=Alert(10,NIL);
End;

Procedure SargJack;
VAR   Hival,
      theval: INTEGER;

Begin
   theval:=GetCtlVAlue(DALevelCtl);
   Hival:=theval div 256;
```

```
      SendAChar(ProgRec.GSArG,chr(Hival));
      theval:=theval - Hival*256;
      SendAChar(ProgRec.GSArG,chr(theval));
  End;

  Procedure GoCheck;
  VAR    val1    : Integer;
          val2   : Integer;

  Begin
     checklev:=0;
     val1:=Maxpk-Imax[1];
     val2:=Minpk-Imin[1];
     IF (val1<-2) OR (val1>2) OR (val2<-2) OR (val2>2) Then
     Begin
        IF (val1>20) OR (val1<-20) Then val1:=(val1*7) div 8;
        IF (val2>20) OR (val2<-20) Then val2:=(val2*7) div 8;
        SendAChar(ProgRec.GSArG,chr(14));
        curmaxpk:=curmaxpk+val1;
        curminpk:=curminpk+val2;
        IF (curmaxpk<1) Then curmaxpk:=1
        ELSE IF (curmaxpk>4095) Then curmaxpk:=4095;
        IF (curminpk>4096) Then curminpk:=4095
        ELSE IF (curminpk<1) Then curminpk:=1;
        val1:=curmaxpk div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=curmaxpk-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        val1:=curminpk div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=curminpk-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        val1:=rate div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=rate-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        SendAChar(ProgRec.GSArG,chr(17));
    End;
 End;

PROCEDURE TrackScroll(theControl: ControlHandle; partCode: INTEGER);
Var     amount, StartValue : INTEGER;
        up : BOOLEAN;
        tickdel,
        delend,
        therefcon : LongInt;
```

```
 BEGIN
     up := partcode IN [inUpButton, inPageUp];
     StartValue := GetCtlValue (theControl);
     IF up THEN amount := -1 ELSE amount := 1;
     IF partCode IN [inPageUp, inPageDown] THEN  amount:=amount*10;
     StartValue:=StartValue+amount;
     IF startValue<0 Then startValue:=0
     ELSE IF startvalue>GetCtlMax(thecontrol) Then startvalue:=GetCtlMax(thecontrol);
     SetCtlValue(theControl, StartValue);
     tickdel:=5;
     Delay(tickdel,delend);
     SargJack;
 END; {of TrackScroll}


 Procedure DiskRErr (io : INTEGER; FName : STR255);
 Var    str: str255;
        readfromstr, loadedstr, str1: Str255;
        dummy: INTEGER;
 BEGIN
    GetIndString (readfromstr, 256, 9); {this says 'reading from'}
    GetIndString (loadedstr ,256, 11); {this says 'loaded'}
    IF io = IOErr
    THEN GetIndString (str, 256, 21)  {this says 'IO error'}
    ELSE
    BEGIN
       NumToString (io, str1);
       GetIndString (str, 256, 22); {this is the generic 'ID ='}
       str := Concat (str, str1)
    END;
    Paramtext (readfromstr, FName, loadedstr, str);
    dummy := StopAlert (256, NIL); {discribe error to user in generic way.}
END;

PROCEDURE DiskWErr (io : INTEGER; fname:STR255);
Var    str:str255;
       writetostr, savedstr, str1: Str255;
       dummy, errstr: INTEGER;
BEGIN
   GetIndString (writetostr,256,10);{read resource for writeto}
   GetIndString (savedstr,256,12);{read resource for saved}
   errstr := 0;
   Case io of
      DskFulErr : errstr := 17;
      DirFulErr : errstr := 18;
      FLckdErr : errstr := 19;
```

```
            VLckdErr, WPrErr : errstr := 20;
            IOErr : errstr := 21;
            OTHERWISE
            BEGIN
                NumToString (io, str);
                GetIndString (str1,256,22);
                str := Concat (str1,str)
            END
        END;
        IF errstr <> 0 THEN GetIndString (str,256,errstr);
        Paramtext (writetostr,FName,savedstr,str);
        dummy := StopAlert (256, NIL);
    END;

    Procedure DrawChData;
    VAR     darect: Rect;
            arect:   RECT;
            loop,
            dataval2:  Integer;
            dataExt:    EXTENDED;
            ExStr:   DecStr;
            DStr:   Str255;

    Begin
        SetPort(mainwindow);
        IF DBool[1] Then
        Begin
            dataval2:=2047-dval[1];
            DataExt:=dataval2*DataConst;
            Num2Str(Data1F,DataExt,Exstr);
            DStr:=STR255(Exstr);
            setRect(darect,pth[1],ptv[1],pth[1]+63,ptv[1]+23);
            EraseRect(darect);
            MoveTo(pth[1]+4,ptv[1]+19);
            DrawString(DStr);
        End;
        IF DBool[2] Then
        Begin
            dataval2:=2047-dval[2];
            DataExt:=dataval2*DataConst;
            Num2Str(Data1F,DataExt,Exstr);
            DStr:=STR255(Exstr);
            setRect(darect,pth[2],ptv[2],pth[2]+63,ptv[2]+23);
            EraseRect(darect);
            MoveTo(pth[2]+4,ptv[2]+19);
            DrawString(DStr);
```

```
 End;
 IF DBool[3] Then
 Begin
     dataval2:=2047-dval[3];
     DataExt:=dataval2*DataConst;
     Num2Str(Data1F,DataExt,Exstr);
     DStr:=STR255(Exstr);
     setRect(darect,pth[3],ptv[3],pth[3]+63,ptv[3]+23);
     EraseRect(darect);
     MoveTo(pth[3]+4,ptv[3]+19);
     DrawString(DStr);
 End;
 IF DBool[4] Then
 Begin
     dataval2:=2047-dval[4];
     DataExt:=dataval2*DataConst;
     Num2Str(Data1F,DataExt,Exstr);
     DStr:=STR255(Exstr);
     setRect(darect,pth[4],ptv[4],pth[4]+63,ptv[4]+23);
     EraseRect(darect);
     MoveTo(pth[4]+4,ptv[4]+19);
     DrawString(DStr);
 End;
 IF DBool[5] Then
 Begin
     dataval2:=2047-dval[5];
     DataExt:=dataval2*DataConst;
     Num2Str(Data1F,DataExt,Exstr);
     DStr:=STR255(Exstr);
     setRect(darect,pth[5],ptv[5],pth[5]+63,ptv[5]+23);
     EraseRect(darect);
     MoveTo(pth[5]+4,ptv[5]+19);
     DrawString(DStr);
 End;
 IF DBool[6] Then
 Begin
     dataval2:=2047-dval[6];
     DataExt:=dataval2*DataConst;
     Num2Str(Data1F,DataExt,Exstr);
     DStr:=STR255(Exstr);
     setRect(darect,pth[6],ptv[6],pth[6]+63,ptv[6]+23);
     EraseRect(darect);
     MoveTo(pth[6]+4,ptv[6]+19);
     DrawString(DStr);
 End;
 IF DBool[7] Then
```

```
        Begin
           dataval2:=2047-dval[7];
           DataExt:=dataval2*DataConst;
           Num2Str(Data1F,DataExt,Exstr);
           DStr:=STR255(Exstr);
           setRect(darect,pth[7],ptv[7],pth[7]+63,ptv[7]+23);
           EraseRect(darect);
           MoveTo(pth[7]+4,ptv[7]+19);
           DrawString(DStr);
        End;
        IF DBool[8] Then
        Begin
           dataval2:=2047-dval[8];
           DataExt:=dataval2*DataConst;
           Num2Str(Data1F,DataExt,Exstr);
           DStr:=STR255(Exstr);
           setRect(darect,pth[8],ptv[8],pth[8]+63,ptv[8]+23);
           EraseRect(darect);
           MoveTo(pth[8]+4,ptv[8]+19);
           DrawString(DStr);
        End;
        IF DBool[9] Then
        Begin
           SetRect(arect,151,40,254,58);
           EraseRect(arect);
           MoveTo(155,55);
           DrawString(D1Str);
        End;
        IF Lnumcyc<>cycdata Then
        Begin
           Setrect(arect,151,21,254,39);EraseRect(arect);
           Lnumcyc:=cycdata;
           numtostring(Lnumcyc,Dstr);
           MoveTo(155,36);DrawString(Dstr);
        End;
End;

Procedure VarSet;
VAR        valint,
      val1,
      val2:       Integer;

Begin
   VACurlev:=1;
   VALastcyc:=0;
   Maxpk:=VAmpTbl[1].VMaxpk;
   Minpk:=VAmpTbl[1].VMinpk;
```

```
        valint:=(minpk-maxpk) div 32;
        Lmax[1]:=MaxPk;
        Lmin[1]:=MinPk;
        rate:=VAmpTbl[1].VRate;
        SendAChar(ProgRec.GSArG,chr(14));
        curmaxpk:=maxpk+valint;
        IF (curmaxpk<0) Then curmaxpk:=0
        ELSE IF (curmaxpk>4095) Then curmaxpk:=4095;
        curminpk:=minpk-valint;
        IF (curminpk>4095) Then curminpk:=4095
        ELSE IF (curminpk<1) Then curminpk:=1;
        val1:=curmaxpk div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=curmaxpk-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        val1:=curminpk div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=curminpk-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        val1:=rate div 256;
        SendAChar(ProgRec.GSArG,chr(val1));
        val2:=rate-val1*256;
        SendAChar(ProgRec.GSArG,chr(val2));
        SendAChar(ProgRec.GSArG,chr(17));
  End;

  Procedure ReadVAmp(VRefNo: Integer; name: STR255);
  TYPE  chrarray=  PACKED ARRAY[1..2000] OF CHAR;
        ChrAPtr=    ^chrarray;

  VAR      refnum:     Integer;
        LogEOF:     LongInt;
        aptr:     Ptr;
        ahdl:    Handle;
        ACAPtr:     chrAPtr;
        io:  OSErr;
        loop2,
        loop:    LONGINT;
        num:    LongInt;
        chrval:  Integer;
        dummy:     BOOLEAN;

Begin
   io:=FSOpen(name,vrefno,refnum);
   IF io <> 0 THEN DiskRErr (io,name);
   io := GetEOF (RefNum, logEOF);
```

```
IF io <> 0 THEN DiskRErr (io,name);
ahdl:=NewHandle(SizeOf(ChrArray));
HLock(ahdl);
aptr:=ahdl^;
ACAPtr:=ChrAPtr(aptr);
io := FSRead (refNum, logEOF, aptr);
IF io <> 0 THEN DiskRErr (io,name);
io:=FSClose(refnum);
IF io <> 0 THEN DiskRErr (io,name);
num:=0;dummy:=FALSE;loop2:=1;
Repeat
    Chrval:=ord(ACAPtr^[loop2]);
    IF (Chrval>=$30) AND (Chrval<=$39) Then num:=num*10+(Chrval-48)
    ELSE dummy:=TRUE;
    loop2:=loop2+1;
Until dummy;
VANumlev:=num;
FOR loop:= 1 TO VANumlev DO
Begin
    num:=0;dummy:=FALSE;
    Repeat
        Chrval:=ord(ACAPtr^[loop2]);
        IF (Chrval>=$30) AND (Chrval<=$39) Then num:=num*10+(Chrval-48)
        ELSE dummy:=TRUE;
        loop2:=loop2+1;
    Until dummy;
    VAmpTbl[loop].VMaxpk:=num;

    num:=0;dummy:=FALSE;
    Repeat
        Chrval:=ord(ACAPtr^[loop2]);
        IF (Chrval>=$30) AND (Chrval<=$39) Then num:=num*10+(Chrval-48)
        ELSE dummy:=TRUE;
        loop2:=loop2+1;
    Until dummy;
    VAmpTbl[loop].VMinpk:=num;

    num:=0;dummy:=FALSE;
    Repeat
        Chrval:=ord(ACAPtr^[loop2]);
        IF (Chrval>=$30) AND (Chrval<=$39) Then num:=num*10+(Chrval-48)
        ELSE dummy:=TRUE;
        loop2:=loop2+1;
    Until dummy;
    VAmpTbl[loop].VRate:=num;

    num:=0;dummy:=FALSE;
```

```
        Repeat
            Chrval:=ord(ACAPtr^[loop2]);
            IF (Chrval>=$30) AND (Chrval<=$39) Then num:=num*10+(Chrval-48)
            ELSE dummy:=TRUE;
            loop2:=loop2+1;
        Until dummy;
        VAmpTbl[loop].Ncycle:=num;
    End;
    VarSet;
    DisposHandle(ahdl);
End;

Procedure Sendch;
VAR    val : Integer;
Begin
    val:=112;
    IF chselected[1] THEN val:=Val+1;
    IF chselected[2] THEN val:=Val+2;
    IF chselected[3] THEN val:=Val+4;
    IF chselected[4] THEN val:=Val+8;
    SendAChar(ProgRec.GSArG,chr(val));
End;

Procedure SaveData;
TYPE   ChrPtr = ^Byte;
VAR    refNum,
       loop,
       loop2,
       acount,
       dataval2,
       io : INTEGER;
       gennum1,
       gennum2,
       dataval: LongInt;
       acptr:  ChrPtr;
       dataExt:    EXTENDED;
       DataF:   DecForm;
       ExStr:   DecStr;
       Astr: STR255;
       genptr,
       dStrPtr:  Ptr;
       Nextch: BOOLEAN;

Begin
   EnableItem(myMenus[1],0);
   EnableItem(myMenus[2],3);
```

```
  EnableItem(myMenus[2],10);
  EnableItem(myMenus[3],9);
  io := FSOpen(progrec.dataFName, progrec.VRefNum, refNum);
  acptr:=chrptr(Newptr(sizeof(Byte)));
  acptr^:=42;gennum1:=1;
  genptr:=pointer(ord(acptr)+1);
  io := FSWrite (refNum, gennum1, genptr);

  acptr^:=13;gennum1:=1;
  genptr:=pointer(ord(acptr)+1);
  io := FSWrite (refNum, gennum1, genptr);

  acptr^:=9;gennum1:=1;
  genptr:=pointer(ord(acptr)+1);
  IF datacount[1]>0 Then
  Begin
     aStr:='Cycles  (1)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
     io := FSWrite (refNum, gennum1, genptr);
     aStr:='Max.  (1)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
     io := FSWrite (refNum, gennum1, genptr);
     aStr:='Min.  (1)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
  End;
  IF datacount[2]>0 Then
  Begin
     io := FSWrite (refNum, gennum1, genptr);
     aStr:='Cycles  (2)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
     io := FSWrite (refNum, gennum1, genptr);
     aStr:='Max.  (2)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
     io := FSWrite (refNum, gennum1, genptr);
     aStr:='Min.  (2)';gennum2:=length(astr);
     dstrptr:=pointer(ord(@astr)+1);
     io := FSWrite (refNum, gennum2, dstrptr);
  End;
  IF datacount[3]>0 Then
  Begin
     io := FSWrite (refNum, gennum1, genptr);
```

```
    aStr:='Cycles  (3)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
    io := FSWrite (refNum, gennum1, genptr);
    aStr:='Max.  (3)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
    io := FSWrite (refNum, gennum1, genptr);
    aStr:='Min.  (3)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
End;
IF datacount[4]>0 Then
Begin
    io := FSWrite (refNum, gennum1, genptr);
    aStr:='Cycles  (4)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
    io := FSWrite (refNum, gennum1, genptr);
    aStr:='Max.  (4)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
    io := FSWrite (refNum, gennum1, genptr);
    aStr:='Min.  (4)';gennum2:=length(astr);
    dstrptr:=pointer(ord(@astr)+1);
    io := FSWrite (refNum, gennum2, dstrptr);
End;
DataF.style:=FixedDecimal;
DataF.digits:=2;
acptr^:=13;gennum1:=1;
genptr:=pointer(ord(acptr)+1);
io := FSWrite (refNum, gennum1, genptr);
IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
acount:=datacount[1]-1;
IF acount<datacount[2] Then acount:=datacount[2]-1;
IF acount<datacount[3] Then acount:=datacount[3]-1;
IF acount<datacount[4] Then acount:=datacount[4]-1;
FOR loop:= 0 TO acount DO
Begin
    acptr^:=9;gennum1:=1;
    genptr:=pointer(ord(acptr)+1);
    IF datacount[1]>0 Then
    Begin
        IF loop<datacount[1] Then
        Begin
            dataval:=cyc1dathdl^^[loop];
            numtostring(dataval,astr);
```

```
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            io := FSWrite (refNum, gennum1, genptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
            dataval:=pk1dathdl^^[loop].Maxpk;
            dataval2:=2048-dataval;
            DataExt:=dataval2*DataConst;
            Num2Str(DataF,DataExt,Exstr);
            aStr:=STR255(Exstr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            io := FSWrite (refNum, gennum1, genptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
            dataval:=pk1dathdl^^[loop].Minpk;
            dataval2:=2048-dataval;
            DataExt:=dataval2*DataConst;
            Num2Str(DataF,DataExt,Exstr);
            aStr:=STR255(Exstr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
        End
        ELSE
        Begin
            io := FSWrite (refNum, gennum1, genptr);
            io := FSWrite (refNum, gennum1, genptr);
        End;
    End;
    IF datacount[2]>0 Then
    Begin
        io := FSWrite (refNum, gennum1, genptr);
        IF loop<datacount[2] Then
        Begin
            dataval:=cyc2dathdl^^[loop];
            numtostring(dataval,astr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            io := FSWrite (refNum, gennum1, genptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
            dataval:=pk2dathdl^^[loop].Maxpk;
            dataval2:=2048-dataval;
            DataExt:=dataval2*DataConst;
```

```
        Num2Str(DataF,DataExt,Exstr);
        aStr:=STR255(Exstr);
        gennum2:=length(astr);
        dstrptr:=pointer(ord(@astr)+1);
        io := FSWrite (refNum, gennum2, dstrptr);
        io := FSWrite (refNum, gennum1, genptr);
        IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
        dataval:=pk2dathdl^^[loop].Minpk;
        dataval2:=2048-dataval;
        DataExt:=dataval2*DataConst;
        Num2Str(DataF,DataExt,Exstr);
        aStr:=STR255(Exstr);
        gennum2:=length(astr);
        dstrptr:=pointer(ord(@astr)+1);
        io := FSWrite (refNum, gennum2, dstrptr);
        IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
    End
    ELSE
    Begin
        io := FSWrite (refNum, gennum1, genptr);
        io := FSWrite (refNum, gennum1, genptr);
    End;
End;
IF datacount[3]>0 Then
Begin
    io := FSWrite (refNum, gennum1, genptr);
    IF loop<datacount[3] Then
    Begin
        dataval:=cyc3dathdl^^[loop];
        numtostring(dataval,astr);
        gennum2:=length(astr);
        dstrptr:=pointer(ord(@astr)+1);
        io := FSWrite (refNum, gennum2, dstrptr);
        io := FSWrite (refNum, gennum1, genptr);
        IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
        dataval:=pk3dathdl^^[loop].Maxpk;
        dataval2:=2048-dataval;
        DataExt:=dataval2*DataConst;
        Num2Str(DataF,DataExt,Exstr);
        aStr:=STR255(Exstr);
        gennum2:=length(astr);
        dstrptr:=pointer(ord(@astr)+1);
        io := FSWrite (refNum, gennum2, dstrptr);
        io := FSWrite (refNum, gennum1, genptr);
        IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
        dataval:=pk3dathdl^^[loop].Minpk;
        dataval2:=2048-dataval;
```

```
            DataExt:=dataval2*DataConst;
            Num2Str(DataF,DataExt,Exstr);
            aStr:=STR255(Exstr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
        End
        ELSE
        Begin
            io := FSWrite (refNum, gennum1, genptr);
            io := FSWrite (refNum, gennum1, genptr);
        End;
    End;
    IF datacount[4]>0 Then
    Begin
        io := FSWrite (refNum, gennum1, genptr);
        IF loop<datacount[4] Then
        Begin
            dataval:=cyc4dathdl^^[loop];
            numtostring(dataval,astr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            io := FSWrite (refNum, gennum1, genptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
            dataval:=pk4dathdl^^[loop].Maxpk;
            dataval2:=2048-dataval;
            DataExt:=dataval2*DataConst;
            Num2Str(DataF,DataExt,Exstr);
            aStr:=STR255(Exstr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            io := FSWrite (refNum, gennum1, genptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
            dataval:=pk4dathdl^^[loop].Minpk;
            dataval2:=2048-dataval;
            DataExt:=dataval2*DataConst;
            Num2Str(DataF,DataExt,Exstr);
            aStr:=STR255(Exstr);
            gennum2:=length(astr);
            dstrptr:=pointer(ord(@astr)+1);
            io := FSWrite (refNum, gennum2, dstrptr);
            IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
      End
```

```
            ELSE
            Begin
                io := FSWrite (refNum, gennum1, genptr);
                io := FSWrite (refNum, gennum1, genptr);
            End;
        End;
        acptr^:=13;gennum1:=1;
        genptr:=pointer(ord(acptr)+1);
        io := FSWrite (refNum, gennum1, genptr);
        IF io <> 0 THEN DiskWErr(io,progrec.dataFName);
    End;
    io := GetFPos(refNum, gennum1);
    IF io<>0 Then DiskWErr(io,progrec.dataFName);
    io := SetEOF (refNum, gennum1);
    IF io<>0 Then DiskWErr(io,progrec.dataFName);
    io := FSClose(refnum);
    IF io<>0 Then DiskWErr(io,progrec.dataFName);
    io := Flushvol(NIL,progrec.vrefnum);
    IF io<>0 Then DiskWErr(io,progrec.dataFName);
End;

Procedure DoMenuCommand(mResult: LongInt);
VAR   refnum,
      theItem,
      theitm,
      tp,
      loop2,
      loop,
      theMenu: INTEGER;
      numcode:Longint;
      Adialg: Dialogptr;
      edthdl: Handle;
      arect: Rect;
      astr,
      astr1,
      astr2,
      astr3,
      name: STR255;
      val: LOngInt;
      err: OSErr;
      achr: Char;
      pcode: integer;
      numDec:  Extended;
      io : INTEGER;
      aptr: Ptr;
      wher:      Point;
```

```
        SFTList:   SFTypeList;
        Preply:     SFReply;
        ioOK:       BOOLEAN;

BEGIN
    theMenu  := HiWord(mResult);
    theItem  := LoWord(mResult);
    CASE theMenu OF

    appleMenu:
     BEGIN
        GetItem(myMenus[1],theItem,name);

        refNum := OpenDeskAcc(name);
     END;

    GenMenu:
    CASE theItem OF
    1:   Begin
            D1Str:='';
            SetPort(mainwindow);
            setRect(arect,151,81,214,104);
            EraseRect(arect);
            setRect(arect,151,106,214,129);
            EraseRect(arect);
            setRect(arect,151,131,214,154);
            EraseRect(arect);
            setRect(arect,151,156,214,179);
            EraseRect(arect);
            SetRect(arect,231,81,294,104);
            EraseRect(arect);
            SetRect(arect,231,106,294,129);
            EraseRect(arect);
            SetRect(arect,231,131,294,154);
            EraseRect(arect);
            SetRect(arect,231,156,294,179);
            EraseRect(arect);
            SetRect(arect,151,40,254,58);
            EraseRect(arect);
            For loop:=1 TO 9 DO DBool[loop]:=FALSE;
      End;
  2:  Begin
           FOR loop:=1 TO 4 DO
           Begin
               StorData[loop]:=FALSE;
               SMax[loop]:=0;
               SMin[loop]:=0;
```

```
            LMax[loop]:=0;
            LMin[loop]:=0;
            avMinpt[loop]:=1;
            avMaxpt[loop]:=1;
            NumMinpt[loop]:=0;
            NumMaxpt[loop]:=0;
            MinTotal[loop]:=0;
            MaxTotal[loop]:=0;
            FOR loop2:=1 TO 5 DO
            Begin
                ChMinaver[loop,loop2]:=0;
                ChMaxaver[loop,loop2]:=0;
            End;
            DataCount[loop]:=0;
        End;
        DisposHandle(Handle(cyc1dathdl));
        DisposHandle(Handle(pk1dathdl));
        DisposHandle(Handle(cyc2dathdl));
        DisposHandle(Handle(pk2dathdl));
        DisposHandle(Handle(cyc3dathdl));
        DisposHandle(Handle(pk3dathdl));
        DisposHandle(Handle(cyc4dathdl));
        DisposHandle(Handle(pk4dathdl));
        Lnumcyc:=0;numbytes:=0;cycdata:=0;
        cyc1dathdl:=cychdl(NewHandle(sizeof(cycdat)));
        pk1dathdl:=pkhdl(Newhandle(sizeof(pkData)));
        cyc2dathdl:=cychdl(NewHandle(sizeof(cycdat)));
        pk2dathdl:=pkhdl(Newhandle(sizeof(pkData)));
        cyc3dathdl:=cychdl(NewHandle(sizeof(cycdat)));
        pk3dathdl:=pkhdl(Newhandle(sizeof(pkData)));
        cyc4dathdl:=cychdl(NewHandle(sizeof(cycdat)));
        pk4dathdl:=pkhdl(Newhandle(sizeof(pkData)));
        Setrect(arect,151,21,254,39);EraseRect(arect);
        IF DAMode=3 Then VarSet;
        DisableItem(myMenus[2],3);
    End;
3:  Begin
        wher.h:=100;
        wher.v:=100;
        sfPutFile(wher,'Save data in file:',ProgRec.datafname,NIL,PReply);
        With PReply DO
        IF good Then
        Begin
            Progrec.datafname:=Fname;
            progrec.vrefnum:=vrefnum;
            io := FSOpen(FName, VRefNum, refNum);
```

```
            IF io = {file not found Err} -43 THEN
            BEGIN
                io := Create (FName,VRefNum,'CGRF','CGTX');
                io := FSOpen(FName, VRefNum, refNum);
            END; {Create}
            val:=0;
            io := SetEOF(refnum,val);
            io := FSClose (refNum);
            io := FlushVol (NIL, VrefNum);
            SaveData;
        End;
    End;
 5: Begin
        IF chselected[1] Then
        Begin
            chselected[1]:=FALSE;
            checkitem(mymenus[2],5,FALSE);
        End
        ELSE
        Begin
            chselected[1]:=TRUE;
            checkitem(mymenus[2],5,TRUE);
        End;
        SendCh;
    End;
 6: Begin
        IF chselected[2] Then
        Begin
            chselected[2]:=FALSE;
            checkitem(mymenus[2],6,FALSE);
        End
        ELSE
        Begin
            chselected[2]:=TRUE;
            checkitem(mymenus[2],6,TRUE);
        End;
        SendCh;
    End;
 7: Begin
        IF chselected[3] Then
        Begin
            chselected[3]:=FALSE;
            checkitem(mymenus[2],7,FALSE);
        End
        ELSE
        Begin
            chselected[3]:=TRUE;
```

```
            checkitem(mymenus[2],7,TRUE);
          End;
          SendCh;
      End;
  8:  Begin
          IF chselected[4] Then
          Begin
              chselected[4]:=FALSE;
              checkitem(mymenus[2],8,FALSE);
          End
          ELSE
          Begin
              chselected[4]:=TRUE;
              checkitem(mymenus[2],8,TRUE);
          End;
          SendCh;
      End;
  10: ProgRec.Quit:=TRUE;
  End;
  Sendmenu:
  CASE theItem OF
  1:  SendAChar(ProgRec.GSArG,chr(17));
  2:  SendAChar(ProgRec.GSArG,chr(18));
  3:  SendAChar(ProgRec.GSArG,chr(19));
  4:  Begin
          SendAChar(ProgRec.GSArG,chr(15));
          DAMode:=1;
      End;
  5:  Begin
          Adialg:=GetNewDialog(13   ,NIL,Pointer(-1));
          ModalDialog(NIL,theitm);
          GetDItem(Adialg,2,tp,edthdl,arect);
          GetItext(edthdl,astr1);
          GetDItem(Adialg,4,tp,edthdl,arect);
          GetItext(edthdl,astr2);
          GetDItem(Adialg,6,tp,edthdl,arect);
          GetItext(edthdl,astr3);
          DisposDialog(Adialg);
          SendAChar(ProgRec.GSArG,chr(17));
          numDec:=Str2Num(DecStr(astr1));
          numDec:=2047-numDec*2048/100;
          Maxpk:=Num2Integer(numDec);
          numDec:=Str2Num(DecStr(astr2));
          numDec:=2047-numDec*2048/100;
          Minpk:=Num2Integer(numDec);
          pcode:=(minpk-maxpk)  div  32;val:=(minpk+maxpk)  div  2;
```

```
            curmaxpk:=val-15*pcode;
            curminpk:=val+15*pcode;
            numDec:=Str2Num(DecStr(astr3));
            numDec:=2000000/(numDec*180);
            rate:=Num2Integer(numDec);
            SendAChar(ProgRec.GSArG,chr(14));
            Lmax[1]:=Maxpk;
            Lmin[1]:=Minpk;
            IF (curmaxpk<0) Then curmaxpk:=0;
            IF (curminpk>4096) Then curminpk:=4095;
            val:=curmaxpk div 256;
            SendAChar(ProgRec.GSArG,chr(val));
            val:=curmaxpk-val*256;
            SendAChar(ProgRec.GSArG,chr(val));
            val:=curminpk div 256;
            SendAChar(ProgRec.GSArG,chr(val));
            val:=curminpk-val*256;
            SendAChar(ProgRec.GSArG,chr(val));
            val:=rate div 256;
            SendAChar(ProgRec.GSArG,chr(val));
            val:=rate-val*256;
            SendAChar(ProgRec.GSArG,chr(val));
            DAMode:=2;
        End;
    6:  Begin
            wher.h:=100;
            wher.v:=100;
            SFTList[0]:='TEXT';
            SFGetFile(wher,'',NIL,1,SFTList,NIL,PReply);
            With PReply DO
            IF good Then
            Begin
                ReadVAmp(vrefnum,fname);
                DAMode:=3;
            End;
        End;
    7:  Begin
            SendAChar(ProgRec.GSArG,chr(21));
            SendAChar(ProgRec.GSArG,chr(128));
        End;
    9:  Begin
            IF InLevCtl=FALSE Then
            Begin
                ShowControl(DALevelCtl);
                InLevCtl:=TRUE;
                DisableItem(myMenus[1],0);
```

```
            DisableItem(myMenus[2],0);
            DisableItem(myMenus[4],0);
            FOR loop:=1 TO 12 DO DisableItem(myMenus[3],loop);
            astr:='End Jack Control';
            SetItem(myMenus[3],9,astr);
            EnableItem(myMenus[3],9);
            TextFont(Geneva);
            MoveTo(5,203);
            DrawString('Tension');
            MoveTo(226,203);
            DrawString('Compression');
            TextFont(SystemFont);
            SendAChar(ProgRec.GSArG,chr(8));
        End
        ELSE
        Begin
            SendAChar(ProgRec.GSArG,chr(80));
            SendAChar(ProgRec.GSArG,chr(80));
            InLevCtl:=FALSE;
            EnableItem(myMenus[1],0);
            EnableItem(myMenus[2],0);
            EnableItem(myMenus[4],0);
            FOR loop:=1 TO 12 DO EnableItem(myMenus[3],loop);
            astr:='Jack  Control';
            SetItem(myMenus[3],9,astr);
            HideControl(DALevelCtl);
            SetRect(arect,4,190,305,210);EraseRect(arect);
            DisableItem(myMenus[2],3);
            DisableItem(myMenus[2],4);
            DisableItem(myMenus[3],8);
        End;
    End;
10: Begin
        IF ProgRec.StoreData Then
        Begin
            checklev:=0;
            wher.h:=100;
            wher.v:=100;
            sfPutFile(wher,'Save  data  in  file:',ProgRec.datafname,NIL,PReply);
            With PReply DO
            IF good Then
            Begin
                Progrec.datafname:=Fname;
                progrec.vrefnum:=vrefnum;
                io := FSOpen(FName, VRefNum, refNum);
                IF io = {file not found Err} -43 THEN
                BEGIN
```

```
                    io := Create (FName,VRefNum,'CGRF','CGTX');
                    io := FSOpen(FName, VRefNum, refNum);
                END; {Create}
                val:=0;
                io := SetEOF(refnum,val);
                io := FSClose (refNum);
                io := FlushVol (NIL, VrefNum);
            End;
            Adialg:=GetNewDialog(14  ,NIL,Pointer(-1));
            ModalDialog(NIL,theitm);
            GetDItem(Adialg,2,tp,edthdl,arect);
            GetItext(edthdl,astr);
            DisposDialog(Adialg);
            StringToNum(astr,val);
            triglev:=val;
        End;
        DisableItem(myMenus[2],10);
        DisableItem(myMenus[1],0);
        DisableItem(myMenus[3],9);
        progrec.start:=TRUE;
        SendAChar(ProgRec.GSArG,chr(17));
        SendAChar(ProgRec.GSArG,chr(1));
    End;
 11: Begin
        SendAChar(ProgRec.GSArG,chr(96));
    End;
 12:Begin
        Adialg:=GetNewDialog(12  ,NIL,Pointer(-1));
        ModalDialog(NIL,theitm);
        GetDItem(Adialg,2,tp,edthdl,arect);
        GetItext(edthdl,astr);
        DisposDialog(Adialg);
        StringToNum(astr,val);
        IF val>255 Then
        Begin
            tp:=val div 256;
            SendAChar(ProgRec.GSArG,chr(tp));
            val:=val-tp*256;
        End;
        SendAChar(ProgRec.GSArG,chr(val));
    End;
End;
Smoothmenu:
Begin
    IF theItem<5 then
    Begin
```

```
            IF avselected[theItem] Then
            Begin
                avselected[theItem]:=FALSE;
                checkitem(mymenus[4],theitem,FALSE);
            End
            ELSE
            Begin
                avselected[theItem]:=TRUE;
                checkitem(mymenus[4],theitem,TRUE);
            End;
        End
        ELSE IF theItem=6 Then
        Begin
            progrec.storeData:=TRUE;
            checkitem(mymenus[4],6,TRUE);
            checkitem(mymenus[4],7,FALSE);
        End
        ELSE IF theItem=7 Then
        Begin
            progrec.storeData:=FALSE;
            checkitem(mymenus[4],7,TRUE);
            checkitem(mymenus[4],6,FALSE);
        End;
    End;
    End;
    HiliteMenu(0);
End;


PROCEDURE STestWind;
VAR    arect: Rect;
       loop:  Integer;
Begin
    TextFont(systemFont);
    SetRect(arect,150,20,255,40);FrameRect(arect);
    MoveTo(40,36);DrawString('Cycle  counter');
    SetRect(arect,150,39,255,59);FrameRect(arect);
    MoveTo(40,55);
    DrawString('SArGen  Status');
    MoveTo(150,76);
    DrawString('Max  Peak');
    MoveTo(230,76);
    DrawString('Min  Peak');
    SetRect(arect,150,80,215,180);FrameRect(arect);
    MoveTo(150,105);LineTo(214,105);
    MoveTo(150,130);LineTo(214,130);
    MoveTo(150,155);LineTo(214,155);
```

```
      SetRect(arect,230,80,295,180);FrameRect(arect);
      MoveTo(230,105);LineTo(294,105);
      MoveTo(230,130);LineTo(294,130);
      MoveTo(230,155);LineTo(294,155);
      MoveTo(20,100);DrawString('CHANNEL  1');
      MoveTo(20,125);DrawString('CHANNEL  2');
      MoveTo(20,150);DrawString('CHANNEL  3');
      MoveTo(20,175);DrawString('CHANNEL  4');
 End;

 Procedure Initialise;
 VAR    arect: Rect;
        loop2,
        loop:  Integer;
        DSize: Size;

 PROCEDURE SetUpMenus;          { Once-only initialization for menus }
 VAR   i   :   INTEGER;
         appleTitle:    STRING[1];        { This is set to the Apple character }

 BEGIN
    InitMenus;
    appleTitle := ' ';
    appleTitle[1] := CHR(20);
    myMenus[1] := NewMenu(appleMenu,appleTitle);
    AddResMenu(myMenus[1],'DRVR');
    FOR i:=2 TO lastMenu DO mymenus[i] := GetMenu(i);
    FOR i:=1 TO lastMenu DO InsertMenu(myMenus[i],0);
    DrawMenuBar;
END;   { of SetUpMenus }

BEGIN
   InitGraf(@thePort);
   InitFonts;

   FlushEvents(everyEvent,0);
   InitCursor;

   InitWindows;
   mainwindow:=GetNewWindow(128,NIL,pointer(-1));

   TEInit;
   SetPort(mainwindow);
   tempwindow:=mainwindow;
   STestWind;
   SetRect(arect,5,205,305,225);
```

```
DALevelCtl:=NewControl(mainwindow,arect,'',FALSE,2047,0,4095,scrollbarproc,2);
InitDialogs(NIL);
InitCursor;
SetUpMenus;
InitSArG;
ProgRec.GSArG:= NewSArG;
OpenSArG(Progrec.GSArG, SPortB);
SendAChar(ProgRec.GSArG,chr(11));
SendAChar(ProgRec.GSArG,chr(80));
With ProgRec DO
Begin
    SarGAvail  := TRUE;
    channel    := 1;
    fname   := '';
    datafname  := '';
    Vrefnum    := 0;
    StoreData  := TRUE;
    progX   := FALSE;
    PReady := FALSE;
    Start   := FALSE;
    Quit    := FALSE;
End;
FOR loop:=1 TO 9 DO
Begin
    DBool[loop]:=FALSE;
    DVal[loop]:=2047;
End;
D1Str:='';
FOR loop:=1 TO 4 DO
Begin
    datacount[loop]:=0;
    StorData[loop]:=FALSE;
    DataMax[loop]:=TRUE;
    SMax[loop]:=0;
    SMin[loop]:=0;
    LMax[loop]:=0;
    LMin[loop]:=0;
    avMinpt[loop]:=1;
    avMaxpt[loop]:=1;
    NumMinpt[loop]:=0;
    NumMaxpt[loop]:=0;
    MinTotal[loop]:=0;
    MaxTotal[loop]:=0;
    FOR loop2:=1 TO 5 DO
    Begin
        ChMinaver[loop,loop2]:=0;
```

```
            ChMaxaver[loop,loop2]:=0;
        End;
    End;
    cyc1dathdl:=cychdl(NewHandle(sizeof(cycdat)));
    pk1dathdl:=pkhdl(Newhandle(sizeof(pkData)));
    cyc2dathdl:=cychdl(NewHandle(sizeof(cycdat)));
    pk2dathdl:=pkhdl(Newhandle(sizeof(pkData)));
    cyc3dathdl:=cychdl(NewHandle(sizeof(cycdat)));
    pk3dathdl:=pkhdl(Newhandle(sizeof(pkData)));
    cyc4dathdl:=cychdl(NewHandle(sizeof(cycdat)));
    pk4dathdl:=pkhdl(Newhandle(sizeof(pkData)));
    Lnumcyc:=0;numbytes:=0;cycdata:=0;DAMode:=0;
    InlevCtl:=FALSE;
    chselected[1]:=TRUE;
    chselected[2]:=TRUE;
    chselected[3]:=FALSE;
    chselected[4]:=FALSE;
    Sendch;
    FOR loop:=1 TO 4 DO
    Begin
        pth[loop]:=151;
        pth[loop+4]:=231;
    End;
    ptv[1]:=81;ptv[2]:=106;ptv[3]:=131;ptv[4]:=156;
    ptv[5]:=81;ptv[6]:=106;ptv[7]:=131;ptv[8]:=156;
    Data1F.style:=FixedDecimal;
    Data1F.digits:=2;
    DataConst:=100/2048;
    avselected[2]:=TRUE;
    avselected[1]:=FALSE;
    avselected[3]:=FALSE;
    avselected[4]:=FALSE;
    checkitem(mymenus[4],2,TRUE);
    checkitem(mymenus[2],6,TRUE);
    checkitem(mymenus[2],5,TRUE);
    checkitem(mymenus[4],6,TRUE);
END;

Procedure ProcContent(whwindow:WindowPtr; theEvent:EventRecord);
VAR   ptlocal : point;
      code,dummy: Integer;
      wcontrol: ControlHandle;
      therefcon: LongInt;

Begin
   ptlocal:=theEvent.where;
   GlobalToLocal(ptlocal);
```

```
    code:=FindControl(ptlocal,whwindow,wcontrol);
    IF wControl<>NIL Then
    Begin
        IF code = inThumb THEN
        BEGIN
            dummy := TrackControl (wcontrol, ptlocal, NIL);
            SargJack;
        END
        ELSE dummy := TrackControl (wcontrol, ptlocal, @TrackScroll);
    End;
End;

Procedure KeyInput(theevt: EventRecord);
VAR    keych,spch : CHAR;
       hexspace,
       keyval: integer;
       themess:Longint;

Begin
    themess:=theevt.message;
    keyval:=themess mod 256;
End;

Procedure Checkbytes(VAR thebytes:Longint);
VAR         err:    OSErr;
       aptr:   Ptr;
       nbptr: ^LongInt;

BEGIN
    err:=Status(-20,2,aptr);
    IF err<>noerr Then anyerr(err);
    nbptr:=pointer(ord(aptr));
    thebytes:=nbptr^;
END;

PROCEDURE ToUpDWindow(TheEvent: EventRecord);
VAR         tempport:   GrafPtr;
       whichwindow:  WindowPtr;

Begin
    whichwindow:=Pointer(theEvent.message);
    BeginUpDate(whichwindow);
    GetPort(tempPort);
    SetPort(whichWindow);
    STestWind;
    IF InLevCtl Then Drawcontrol(whichwindow);
```

```
    DrawChData;
    SetPort(TempPort);
    EndUpDate(whichwindow);
End;

Procedure  ListenSArG(Lisbytes:Longint);
VAR   err:        OSErr;
      aptr:       Ptr;
      io:         OSErr;
      Horzshift,
      loop,
      loop2,
      avval,
      avtot,
      dataval,
      dataval2,
      dataval3:   Integer;
      dataval1:   Integer;
      Lisbuf:     PACKED ARRAY [0..3000] OF CHAR;
      AlertRpy:   BOOLEAN;

Begin
    aptr:=@Lisbuf;
    err:=FSRead(-20,Lisbytes,aptr);
    IF  err<>noerr  Then  Anyerr(err);
    loop:=0;
    While loop<Lisbytes DO
    Begin
        dataval:=ord(Lisbuf[loop]);
        IF dataval>=$80 Then
        Begin
            loop:=loop+1;
            IF loop<Lisbytes Then
            Begin
                dataval2:=ord(Lisbuf[loop]);
                loop:=loop+1;
                IF loop<Lisbytes Then
                Begin
                    dataval1:=dataval2*256+ord(Lisbuf[loop]);
                    IF  dataval=$80  Then  horzshift:=1
                    ELSE  IF  dataval=$90  Then  horzshift:=2
                    ELSE  IF  dataval=$A0  Then  horzshift:=3
                    ELSE  IF  dataval=$B0  Then  horzshift:=4
                    ELSE  horzshift:=9;
                    IF  horzshift<5  Then
                    Begin
                        IF  progrec.start=TRUE  Then
```

```
Begin
    IF datamax[horzshift]=TRUE Then
    Begin
        datamax[horzshift]:=FALSE;
        dataval3:=dataval1-Smax[horzshift];
        IF avselected[horzshift] Then
        Begin
            IF (ProgRec.StoreData) AND
                    ((dataval3<-25) OR (dataval3>25)) Then
            Begin
                avMaxpt[horzshift]:=1;NumMaxpt[horzshift]:=0;
                Maxtotal[horzshift]:=0;
            End
            ELSE
            Begin
                avval:=avmaxpt[horzshift];
                avtot:=MaxTotal[horzshift];
                IF NumMaxpt[horzshift]<3 Then
                Begin
                    chmaxaver[horzshift,avval]:=dataval1;
                    avtot:=avtot+dataval1;
                    dataval1:=avtot div avval;
                    avval:=avval+1;
                    IF avval>3 Then avval:=1;
                    NumMaxpt[horzshift]:=NumMaxpt[horzshift]+1;
                End
                ELSE
                Begin
                    avtot:=avtot-chmaxaver[horzshift,avval];
                    chmaxaver[horzshift,avval]:=dataval1;
                    avval:=avval+1;
                    IF avval>3 Then avval:=1;
                    avtot:=avtot+dataval1;
                    dataval1:=avtot div 3;
                End;
                avmaxpt[horzshift]:=avval;
                Maxtotal[horzshift]:=avtot;
            End;
        End;
        IF ProgRec.StoreData Then
        IF (dataval3<-triglev) OR (dataval3>triglev) Then
        Begin
            StorData[horzshift]:=TRUE;
            SMax[horzshift]:=dataval1;
        End;
        Lmax[horzshift]:=dataval1;
```

```
End
ELSE
Begin
    dataval3:=dataval1-Smin[horzshift];
    IF avselected[horzshift] Then
    Begin
        IF (ProgRec.StoreData) AND
            ((dataval3<-25) OR (dataval3>25)) Then
        Begin
            StorData[horzshift]:=TRUE;
            avMinpt[horzshift]:=1;NumMinpt[horzshift]:=0;
            Mintotal[horzshift]:=0;
        End
        ELSE
        Begin
            avval:=avMinpt[horzshift];
            avtot:=Mintotal[horzshift];
            IF NumMinpt[horzshift]<3 Then
            Begin
                chminaver[horzshift,avval]:=dataval1;
                avtot:=avtot+dataval1;
                dataval1:=avtot div avval;
                avval:=avval+1;
                IF avval>3 Then avval:=1;
                NumMinpt[horzshift]:=NumMinpt[horzshift]+1;
            End
            ELSE
            Begin
                avtot:=avtot-chminaver[horzshift,avval];
                chminaver[horzshift,avval]:=dataval1;
                avval:=avval+1;
                IF avval>3 Then avval:=1;
                avtot:=avtot+dataval1;
                dataval1:=avtot div 3;
            End;
            avminpt[horzshift]:=avval;
            MinTotal[horzshift]:=avtot;
        End;
    End;
    IF ProgRec.StoreData Then
    IF (dataval3<-triglev) OR (dataval3>triglev) Then
    Begin
        StorData[horzshift]:=TRUE;
        SMin[horzshift]:=dataval1;
    End;
    Imin[horzshift]:=dataval1;
```

```
            IF lmax[horzshift]<dataval1 Then datamax[horzshift]:=TRUE
            ELSE
            Begin
                datamax[horzshift]:=FALSE;
                lmin[horzshift]:=lmax[horzshift];
                lmax[horzshift]:=dataval1;
            End;
            IF StorData[horzshift]=TRUE Then
            Begin
                CASE horzshift OF
                1: Begin
                        pk1dathdl^^[datacount[1]].maxpk:=Lmax[1];
                        pk1dathdl^^[datacount[1]].Minpk:=Lmin[1];
                        cyc1dathdl^^[datacount[1]]:=cycdata;
                        datacount[1]:=datacount[1]+1;
                    End;
                2: Begin
                        pk2dathdl^^[datacount[2]].maxpk:=Lmax[2];
                        pk2dathdl^^[datacount[2]].Minpk:=Lmin[2];
                        cyc2dathdl^^[datacount[2]]:=cycdata;
                        datacount[2]:=datacount[2]+1;
                    End;
                3: Begin
                        pk3dathdl^^[datacount[3]].maxpk:=Lmax[3];
                        pk3dathdl^^[datacount[3]].Minpk:=Lmin[3];
                        cyc3dathdl^^[datacount[3]]:=cycdata;
                        datacount[3]:=datacount[3]+1;
                    End;
                4: Begin
                        pk4dathdl^^[datacount[4]].maxpk:=Lmax[4];
                        pk4dathdl^^[datacount[4]].Minpk:=Lmin[4];
                        cyc4dathdl^^[datacount[4]]:=cycdata;
                        datacount[4]:=datacount[4]+1;
                    End;
                End;
                StorData[horzshift]:=FALSE;
                IF datacount[horzshift]>990 Then
                Begin
                    SendAChar(ProgRec.GSArG,chr(96));
                End;
            End;
            horzshift:=horzshift+4;
        End;
    End;
    DVal[horzshift]:=dataval1;
End
ELSE numtostring(dataval1,D1Str);
```

```
                    DBool[horzshift]:=TRUE;
                    loop:=loop+1;
                End;
            End;
        End
        ELSE
        Begin
            IF dataval=$11 Then D1Str:='OK'
            ELSE IF dataval=$12 Then D1Str:='ERROR'
            ELSE IF dataval=$60 Then
            Begin
                Progrec.start:=FALSE;
                D1Str:='Terminated';
                IF ProgRec.StoreData Then SaveData;
            End
            ELSE IF Dataval=$6F Then
            Begin
                cycdata:=cycdata+1;
                checklev:=checklev+1;
            End
            ELSE  D1Str:='Reply  Error';
            DBool[9]:=TRUE;
            loop:=loop+1;
        End;
    End;
End;


BEGIN  { main routine }
    Initialise;
    REPEAT                          { Main event loop
    }
        SystemTask;
        IF progrec.start Then
        Begin
            IF (DAMode=2) Then
            Begin
                IF (cycdata<50) AND (checklev>4) Then Gocheck
                ELSE IF checklev>100 Then Gocheck;
            End
            ELSE IF DAMode=3 Then
            Begin
                genlong:=cycdata-VALastcyc;
                IF genlong>=VAmpTbl[VACurLev].Ncycle Then
                Begin
                    VALastcyc:=VALastcyc+VAmpTbl[VACurLev].Ncycle;
```

```
                VACurLev:=VACurLev+1;
                IF VACurLev>VANumLev Then VACurLev:=1;
                Maxpk:=VAmpTbl[VACurLev].VMaxPk;
                Minpk:=VAmpTbl[VACurLev].VMinPk;
                Rate:=VAmpTbl[VACurLev].VRate;
                GoCheck;
            End
            ELSE IF (genlong<40) AND (checklev>4) Then Gocheck
            ELSE IF checklev>49 Then GOCheck;
        END;
    End;
    Checkbytes(numbytes);
    IF numbytes<>0 Then
    Begin
        ListenSArG(numbytes);
        DrawChData;
    End;
    dumBool:=GetNextEvent(EveryEvent,myEvent);
    CASE myEvent.what OF
    mouseDown :
        Begin
            code := FindWindow(myEvent.where,tempWindow);
            CASE code OF
                inMenuBar:    Begin
                                  genlong:=MenuSelect(myEvent.where);
                                  DoMenuCommand(genlong);
                              End;
                inSysWindow: SystemClick(MyEvent,tempWindow);
                inDrag:           Begin
                                  Setrect(Dragrect,0,0,364,720);
                                  DragWindow(tempWindow,myEvent.where,dragrect);
                              End;
                inContent:    IF tempWindow<>FrontWindow Then
SelectWindow(tempwindow)
                              ELSE  ProcContent(tempWindow,myevent);
            END;
        End;
   UpDateEvt :
        Begin
            ToUpDWindow(MyEvent);
        End;
   keyDown :
        Begin
            genlong:=myEvent.modifiers div 256;
            genlong:=genlong mod 2;
            IF genlong=1  Then
            Begin
```

```
                    code:=myEvent.message mod 256;
                    genchar:=chr(code);
                    genlong:=menukey(genchar);
                    DoMenuCommand(genlong);
                End
                ELSE  keyInput(myEvent);
            End;
        END;
    UNTIL ProgRec.Quit;
END.
```

Sargen/Fatigue.Rsrc

```
Type TSAI = STR
,0  (32)
Fatigue test, by K T TSAI   Version 1.0    June 20, 1986

Type MENU
,2
General
Clear Replies/R
Clear Data
(Save Data
( -
Channel 1
Channel 2
Channel 3
Channel 4
( -
Quit/Q

,3
SArGen
XON/1
XOFF/2
Calling Sargen/3
DA OFF/4
DA Constant Amp Wave/5
DA Variable Amp Wave/6
Zero SGA/7
( -
Jack Control/J
Begin Test/B
Stop Test/S
Send Integer/I

,4
Data
Smooth Channel 1
Smooth Channel 2
Smooth Channel 3
Smooth Channel 4
( -
Sample and Save
Sample only
```

Type WIND
,128
SArGen Fatigue Testing
70 101 300 411
Visible NoGoAway
0
0

Type DLOG
,12
40 100 75 370
Visible 1 NoGoAway 0
110

,13
40 120 135 370
Visible 1 NoGoAway 0
111

,14
40 100 75 370
Visible 1 NoGoAway 0
112

Type ALRT
,10
100 120 200 390
100
F765

* a stop alert - an error occured while reading or writing the disk
,256 (32)
80 81 180 431
105
5555

Type DITL
,110
2

StatText Disabled
10 10 30 210
Enter decimal value of Char :

EditText Disabled
10 210 26 255

,105
3
BtnItem Enabled
90 267 110 337
OK

StatText Disabled
10 60 70 350
An error occured while ^0 the disk. The file '^1' was not ^2.

StatText Disabled
70 60 90 350
Err Number ^3

,100
3
StatText Disabled
20 55 40 240
SArGen is not responding

StatText Disabled
50 70 70 200
Err Number ^0

BtnItem Enabled
65 10 85 50
OK

,111
6

StatText Disabled
10 10 30 190
Maximum Peak(% of FS) :

EditText Disabled
10 190 26 235

StatText Disabled
40 10 60 190
Minimum Peak(% of FS) :

EditText Disabled
40 190 56 235

StatText Disabled
70 10 90 190
Frequency(Hz) :

EditText Disabled
70 190 86 235

,112
2

StatText Disabled
10 10 30 220
Trigger level for data storage :

EditText Disabled
10 220 26 255
4

Type STR#
,256 (36)
Untitled
Save this document as:
This example was written to demonstrate the Macintosh User Interface.
Show Clipboard
Hide Clipboard
-6
-7
before quitting
reading from
writing to
loaded
saved
-13
Print...
Stop Printing
Copy of\20
This disk is full.
The disk directory is full.
This file
The disk is locked.
The disk is unreadable.
ID =\20

Type DRVR

SArGen/Drvrl.BSarGD,19 (16)

Type CODE
SArgen/Fatiguel,0

Appendix E

Serial Interface Driver

```
{Copyright 1985 by       K. T. Tsai}

{$R-}
{$X-}

Unit  DrvrImpl;

Interface

Uses  {$U-}
        {$U Obj/Memtypes     } Memtypes,
        {$U Obj/QuickDraw    } QuickDraw,
        {$U Obj/OSIntf      } OSIntf,
        {$U Obj/PackIntf     } PackIntf,
        {$U Obj/ToolIntf     } ToolIntf;

Type
    SParity  =     (SParNone, SParOdd, SPar, SParEven);
    SWidth=  (SWid5, SWid7, SWid6, SWid8);
    SStop  =  (SStp, SStp1, SStp1andHalf, SStp2);
    SArG   =  ^SArGRec;
    SArGRec =    Record
                    port:   SPortSel;
                    chrn:   Integer;
                    spd:   Integer;
                    par:    SParity;
                    wid:   SWidth;
                    stp:  SSTop;
                    xon:  Char;
                    xoff:  Char;
                    fXOn:  Boolean;
                    fCTS:  Boolean;
                    fEvBrk: Boolean;
                    fEvCTS: Boolean;
              End;

Procedure     InitSArG;
Function      NewSArG: SArG;
Procedure        OpenSArG(VAR s: SArG; port: SPortSel);
Procedure     SendAChar(s: SArG; thech: char);
Procedure     CloseSArG(s: SArG);
Procedure     EndSArG;
Implementation
```

```
Const
    controlErr = -17;
    statuserr = -18;
    badunitErr = -21;
    unitEmptyErr = -22;
    openerr = -23;
    dInstErr = -26;
    notOpenErr = -28;

Type
    SDirSel  =  (SDirRcv, SDirSnd);


Var
    ASHandle:   Handle;
    ASLoaded:   Boolean;
    ASInstalled:  Array [SPortSel] Of Boolean;

Procedure SArGError(s:sarg;  err:OSErr);
Var
    thestr   : STR255;
    astr,
    a2str  :  str255;
    Arect        : Rect;
    anevent       : eventrecord;
    replied       : Boolean;
    windptr        : Windowptr;
    theev        : Boolean;

Begin
    setrect(arect,10,40,500,90);
    a2str:='';
    astr:='';
    CASE err OF
        noErr :
        Begin
            theStr := ' no Error in Sargen communication';
            numtostring(err,a2str);numtostring(s^.chrn,astr);
        End;
        badunitErr :
        begin
            thestr := 'Error in Sargen communication : Bad Reference number ';
            numtostring(s^.chrn,astr);
```

```
        End;
        dInstErr  :
        Begin
            thestr := 'Error in Sargen communication : Could not find Driver';
            a2str:= 'in Resource file';
        End;
        openErr:
        Begin
            thestr := 'Error in Sargen communication : Driver cannot perform';
            a2str:='the requested reading or writing';astr:='';
        End;
        unitEmptyErr:
        Begin
            thestr := 'Error in Sargen communication : unitEmptyErr ';
            numtostring(s^.chrn,astr);
        End;
        notOpenErr :
            thestr := 'Error in Sargen communication : Driver not open       ';
        controlErr  :
            thestr := 'Error in Sargen communication : Invalid control call ';
        statuserr  :
            thestr := 'Error in Sargen communication : Invalid status call  ';
    END;
    windptr:=Newwindow(NIL,arect,'',TRUE,dboxproc,POINTER(-1),TRUE,10);
    IF  thestr='' Then  numtostring(err,thestr);
    MoveTo(15,60);
    Drawstring(thestr);
    MoveTo(15,80);Drawstring(a2str);
    MoveTo(200,80);Drawstring(astr);
    replied:=FALSE;
    Repeat
        theev:=Getnextevent(EveryEvent,anEvent);
        If anevent.what=mousedown THEN replied:=TRUE;
    Until  replied;
    disposeWindow(windptr);
End;

{$SSInit}
Procedure SetSArGConfig(s: SArG);
Var
    err:        OSErr;
    config:     Integer;
```

```
Begin
    config := 0;
    config := BitAnd(Ord(s^.stp), 3);
    config := BitShift(config, 2);
    config := BitOr(config, BitAnd(Ord(s^.par), 3));
    config := BitShift(config, 2);
    config := BitOr(config, BitAnd(Ord(s^.wid), 3));
    config := BitShift(config, 10);
    config := BitOr(config, BitAnd(94, 1023));

    err := Control(s^.chrn,8,@config);
    If err<>noErr Then SArGError(s,err);         { ** Report error using Dialog ** }
End;

Procedure SetHandshake(s: SArG);
Type
    flgs  =     record
                fXOn : Byte;
                fCTS : Byte;
                fInX : Byte;
                xon  : Char;
                xoff : Char;
                errs : Byte;
                evts : Integer;
            End;
    theflgs = ^flgs;

Var
    aflgs:      theflgs;
    theparm:    ptr;
    err:        OSErr;
    eventFlags: Integer;

Begin
    eventFlags :=       0;
    If s^.fEvCTS Then eventFlags :=     32;                 { 2**5 }
    If s^.fEvBrk Then eventFlags :=     128 + eventFlags;   { 2**7 + eventFlags }
    With aflgs^ Do
    Begin
        fXOn    :=    255;
        fCTS    :=    0;
        fInX    :=    255;
        xon     :=    s^.xon;
        xoff    :=    s^.xoff;
```

```
        errs      :=    0;
        evts      :=    eventFlags;
     End;
     theparm:=pointer(ord(aflgs));
     err  :=  Control(s^.chrn,10,theparm);
     IF err<>noErr Then SArGError(s,err);        { ** Report error using Dialog ** }
 End;

 {               InitSArG; }
 Procedure           InitSArG;
 Const
     SArGDriver = 19;

 Begin
     ASHandle := Nil;
     ASLoaded := False;
     ASInstalled[SPortA] := False;
     ASInstalled[SPortB] := False;

     ASHandle := GetResource('DRVR', SArGDriver);
     DetachResource(ASHandle);
     HNoPurge(ASHandle);
     HLock(ASHandle);
     ASLoaded := True;
 End;

 Procedure DisposNPtr(p:  Ptr);
 Begin
     If p <> Nil Then
     DisposPtr(p);
 End;

 {               NewSArG: SArG; }
 Function        NewSArG;
 Var
     p:         Ptr;
     s:       SArG;

 Begin
     p    :=     NewPtr(SizeOf(SArGRec));
     If (p = Nil) Then
     Begin
         DisposNPtr(p);
```

```
        s := Nil;
    End
    Else
    Begin
        s := Pointer(Ord(p));
        With s^ Do
        Begin
            port :=    SPortA;
            chrn  := -20;
            spd :=     94;           { redundant since running on ext. clock}
            par :=     SParEven;
            wid :=     SWid8;
            stp :=     SStp1;
            xon :=     Chr(17);    { Default XOn character }
            xoff :=    Chr(18);    { Default XOff character }
            fXOn :=    False;
            fCTS :=    False;        { Can't have CTS on *** }
            fEvBrk :=  False;
            fEvCTS := False;
        End;
    End;
    NewSArG := s;
End;


{               OpenSArG(VAR s: SArG; port: SPortSel); }
Procedure       OpenSArG;

Var
    rn:   OSErr;
    drvrrefnum:    Integer;
    name: Str255;

Begin
    s^.port :=    port;
    rn  :=  CloseDriver(-20);
    IF port=SPortA Then name:='.ASArGD'
    ELSE IF port=SPortB Then name:='.BSArGD';
    rn  := OpenDriver(name,s^.chrn);
    IF rn<>noerr  Then  SArGError(s,rn);

    SetSarGConfig(s);
    SetHandshake(s);
```

```
End;

{               SendAChar(s: SArG; thech: Char ); }
Procedure       SendAChar;
Var
    abufptr:  ptr;
    err:      OSErr;
Begin
    abufptr  :=    @thech;
    err   :=  Control(-20,11,abufptr);
    If err <> 0 Then SArGError(s,err);          { ** Report error using Dialog ** }
End;

{               CloseSArG(s: SArG); }
Procedure       CloseSArG;
Var
    rn : OsErr;
Begin
    rn:=CloseDriver(-20);
End;

{               EndSArG; }
Procedure       EndSArG;
Begin
    If ASHandle <> Nil Then
    Begin
        HPurge(ASHandle);
        HUnLock(ASHandle);
        DisposHandle(ASHandle);
        ASHandle := Nil;
        ASLoaded := False;
    End;
End;

End.
```

```
;_______________________________________________________________
;
; SCC Async Driver for Macintosh - SarGen Communication
;
;   written by Kuo Tsing Tsai
;
;    Copyright (c) 1985 by University of Bath
;
;_______________________________________________________________

            .NoList
            .INCLUDE  tlasm/SysEqu.Text     ; general system equates
            .INCLUDE   tlasm/SysErr.Text
            .INCLUDE  tlasm/SysTraps.Text
            .List

            .PROC              SarGdrvr,0
            .DEF   SarGB

PortBVars     .EQU      $2D0            ; serial port B variables and buffer
BInDCE        .EQU      PortBVars+4     ; Device Control Entry ptr for input

; next come variable offsets within the user's local variable buffer

OutDCE        .EQU      0               ;(4) long DCE pointer for output driver
SCCOffset     .EQU      4               ;(2) word of SCC offset . . .

InBufPtr      .EQU      6               ;(4) pointer to local input buffer
BufSize                 .EQU   10
      ;(2) size of local input buffer
BufLow                  .EQU   12
      ;(2) low buf byte count to send XOn
BufHigh       .EQU      14              ;(2) bytes from end of buffer to send XOff

SWHS          .EQU      16              ;(1) software handshake enable
HWHS          .EQU      17              ;(1) hardware handshake enable
XONChar       .EQU      18              ;(1) input char which stops output (SWHS)
XOFFChar      .EQU      19              ;(1) input char which continues output

Options       .EQU      20              ;(1) bit 4 = abort on parity error
                                        ;   bit 5 = abort on overrun
                                        ;   bit 6 = abort on framing error
PostOptions   .EQU      21              ;(1) bit 7=1 enables posting break changes
```

```
                                    ;  bit 5=1 enables posting handshake changes
InSWHS      .EQU    22              ;(1) input XOn/XOff flow control enable
SendXOnff   .EQU    23              ;(1) flag to xmit logic to send XOn/XOff

AsyncErr    .EQU    24              ;(1) error indications (cumulative)
SoftOR      .EQU    0               ;      bit 0 = soft overrun
                                    ;      bit 4 = parity error
                                    ;      bit 5 = overrun error
                                    ;      bit 6 = framing error

FlowOff     .EQU    25              ;(1) 80 = input flow shut off
ReadCmd     .EQU    26              ;(1) FF = read command pending
WriteCmd    .EQU    27              ;(1) FF = write command pending
CTSFlag                 .EQU   28
        ;(1) FF = CTS asserted
XOFFlag     .EQU    29              ;(1) FF = XOFF pending
ContData    .EQU    30              ;(1) FF = continous data input
BufOutIn    .EQU    31              ;(1) FF = BufOut>BufIn

SCCReset    .EQU    32              ;(1) WR9 value for reset
StopBits    .EQU    33              ;(1) stop bits/parity option (WR4 value)
WR1AVal     .EQU    34              ;(1) first WR1 value to write
WR3AVal     .EQU    35              ;(1) first WR3 value to write
WR5AVal     .EQU    36              ;(1) first WR5 value to write
BaudLoCnst  .EQU    37              ;(2) 2 byte baud rate constant (WR12-13)
BaudHiCnst  .EQU    38
RcvrBits    .EQU    39              ;(1) 1 byte receiver bits/char (WR3 value)
XmitBits    .EQU    40              ;(1) 1 byte xmitter bits/char (WR5 value)
WReqPin     .EQU    41              ;(1) w/req pin state (WR1 value)
lastSetup   .EQU    42              ;(2) last SCC init values

BufIndex    .EQU    44              ;(2) index into local buffer (insert)
BufOutdex   .EQU    46              ;(2) index into local buffer (remove)
LocalBuf    .EQU    48              ;(64) local buffer for input chars

LclBufSize  .EQU    1024            ; default input buffer size = 1024 bytes
LclVarSize  .EQU    LocalBuf+LclBufSize+5  ; output driver storage size

SarGB
            .WORD  $4F00            ; read, write, control, status, lock
            .WORD  0,0              ; not an ornament
            .WORD  0                ; no menu

            .WORD  BOpen-SarGB      ; Initialization routine
```

```
        .WORD  BPrime-SarGB      ; input Prime routine
        .WORD  BControl-SarGB    ; shared Control routine
        .WORD  BStatus-SarGB     ; shared Status routine
        .WORD  BClose-SarGB      ; Close routine

        .BYTE  7
; channel A input driver
        .ASCII '.BSarGD'


;_______________________________________________________________
;
; Routine:    Open routines
;
; Arguments:      A1 (input)   --  DCE pointer for this driver
;
; Function:   These are the Open routines for the SCC async-mode drivers;
;             local variables are initialized, buffer storage allocated,
;             interrupt vectors installed, and the SCC initialized.  For
;             input drivers only the Device Control Entry pointer is noted:
;             SCC initialization is done for output drivers only.
;
;             An 'Open' of the RefNum associated with an output port will install
;             interrupt receivers and enable interrupts for both input and
;             output; two 'Open's need to be done for a port to configure input
;             and output DCEs; the Open for the input driver can be done
;             before or after the Open for the output driver.
;
;             Channel A is treated special: the wait/req pin (chan A and B pins
;             are tied together) is programmed so that it is an output which
;             goes low whenever channel A has input data available.  This
;             output can be read via the 6522 and is used by the disk driver
;             to poll for data during disk routines.  If any data is accumulated,
;             it is passed to the special "poll-process" routine of this
;             driver.
;_______________________________________________________________

BOpen
        MOVE.L  A1,BInDCE       ; note the DCE pointer
        MOVEQ   #0,D0           ; get secondary dispatch table offset
        LEA     PortBVars,A2
; local variables address
        LEA     ExtBIntHnd,A4
```

```
        ;   routines
                LEA         RBIntHnd,A5
                LEA         SCBIntHnd,A6
                MOVEQ       #0,D1               ; SCC read/write offsets
                MOVEQ       #$09,D2             ; WR1 value - w/req (for Rx & SC Int only)
                MOVEQ       #64,D3              ; reset channel B ($40)
                MOVE.W      SPPortB,D4

; D0 = ExtIntDT offset                          A0 = open parameter block ptr (not used)
; D1 = SCC read/write offset
        A1 = DCE address
; D2 = WR1 value;Int on 1st char                A2 = local variables pointer location
; D3 = channel reset data
        A3 = transmitter interrupt(no more!!)
; D4 = clk param init values
        A4 = External/Status interrupt handler
;                                               A5 = receiver interrupt handler
;                                               A6 = special-receiver condition rupt handler

OpenInstall     LEA         ExtStsDT,A0         ; install secondary and primary
                MOVE.L      A4,0(A0,D0)
        ;   interrupt handlers
                LEA         Lvl2DT,A0           ; get dispatch table address
                ADD         D0,D0               ; offset is 2x smaller table offset
                MOVEM.L     A5-A6,8(A0,D0)

                MOVE.W      #LclVarSize,D0
                _NewHandle ,SYS,CLEAR           ; get local storage on system heap
                                                ;  clear errors, error options
                                                ; read, write, XOFF and CTS flags
                                                ; index, outdex, inSWHS
                                                ; HWHS, SWHS, XONChar, XOFFChar
                BSET        #Lock,(A0)          ; lock it down
                MOVE.L      A0,DCtlStorage(A1)  ; save handle in our storage pointer
                MOVE.L      (A0),(A2)           ; save pointer in lo-mem
                MOVE.L      (A0),A2             ; and get the pointer

                MOVE.L      A2,A3               ; locals pointer
                MOVE.L      A1,(A3)+            ; output DCE pointer
                MOVE.W      D1,(A3)+            ; SCC channel address offset

                BSR         InstlLBuf           ; install our local buffer

                ST          HWHS(A2)            ; HWHS defaults on
```

```
        ST          Options(A2)       ; abort input on errors also defaults on
        MOVE.B      D3,SCCReset(A2)     ; channel reset data
        MOVE.B      D2,WReqPin(A2) ; WR1 value for SCC initialization

ToInitSCC
        MOVE.W      D4,lastSetup(A2) ; remember this config
        LEA         StopBits(A2),A3
        LEA         InitDefs,A4       ; process clock data
        MOVEQ       #7,D0             ; expand into 8 bytes variable data
        BSET        #12,D4            ; parity enable
        BSET        #14,D4            ; 1 stop bit
        BCLR        #15,D4
        BCLR        #13,D4            ; even parity

@1      MOVE.B      (A4)+,D1          ; rotate left count
        MOVE.W      D4,D3             ; clock pram data
        ROL.W       D1,D3             ; get appropriate bits into low byte
        AND.B       (A4)+,D3          ; only keep relevant bits
        OR.B        (A4)+,D3          ; add in constant bits
        MOVE.B      D3,(A3)+          ; store processed data
        DBRA        D0,@1             ; do all 8 bytes

        MOVEQ       #$64,D1           ; get WR1A mask
        AND.B       (A3),D1           ; form value
        MOVE.B      D1,WR1AVal(A2) ; and store

        BSR.S       InitSCC           ; initialize SCC channel
        MOVEQ       #0,D0             ; no errors
        RTS

;___________________________________________________________________
;   D4  =  [V][V][W][W][X][X][Y][Y]  [Z][Z][Z][Z][Z][Z][Z][Z]
;
;   VV = 1,2,3,   for 1,1.5,2 stop bits  (00 for AppleBus)
;   WW = 0,1,2,3  for no,odd,no,even parity
;   XX = 0,1,2,3, for 5,7,6,8 data bits
;   YY = high byte of baud rate constant, low 2 bits
;   ZZZZZZZZ = low byte of baud rate constant
;___________________________________________________________________

InitDefs    .BYTE   4,$0F,$00     ;(WR4) rotate left 4, leave 4 low bits, add $40
            .BYTE   0,$00,$00     ;(WR4) (dummy entry-WR1AVal will go here)
            .BYTE   12,$C0,$00    ; (WR3)   WR3 - first write
```

```
                .BYTE     11,$60,$02       ; (WR5)  WR5 - first write
                .BYTE     0,$FF,$00        ; (WR12) baud constant, low byte
                .BYTE     8,$03,$00        ; (WR13) baud constant, hi byte
                .BYTE     12,$C0,$01       ; (WR3)  WR3 - final value
                .BYTE     11,$60,$0A       ; (WR5)  WR5 - final value

;_______________________________________________________________
;
;Routine:     SCC Initialize Routine
;
; Arguments:       A2 (input) -- pointer to local variables for this port
;
; Function:    This routine initializes one port of the SCC for asynchronous
;              communication; the baud rate, data bits, stop bits, and parity
;              options are set according to local variable values. The channel
;              is reset before options are configured. The baud-rate generator
;              output is used for both transmitter and receiver clocks, and
;              interrupts are enabled: only DCD and Break external
;                 interrupts are enabled; all transmitter and receiver interrupts
;              are enabled. Parity errors are configured to generate special
;              condition vectors.
;
; Other:       Registers A0-A3 are used.
;
;_______________________________________________________________
;
;_______________________________________________________________
;
; initialization data for SCC: RS-232 async communication:
;     FORMAT: data,register#  - for immediate data
;             register#,$FF   - for variable data
;_______________________________________________________________

InitData     .BYTE     9,$FF            ; reset SCC channel
             .BYTE     4,$FF
      ; x1 clk, stop bits, parity options
             .BYTE     1,$FF            ; WR1 reg, first write
             .BYTE     $00,2
      ; zero interrupt vector
             .BYTE     3,$FF            ; bits/char option rcvr
             .BYTE     5,$FF            ; bits/char option xmitter
             .BYTE     $02,9            ; status in low bits
             .BYTE     $AD,11           ; rcvr,xmitter to TRxC pin
             .BYTE     12,$FF           ; set baud rate low byte
```

```
            .BYTE     13,$FF              ; set baud rate high byte

            .BYTE     3,$FF               ; enable rcvr
            .BYTE     5,$FF               ; enable xmitter
            .BYTE     $00,14              ; disb baud rate generator from RTxC pin

            .BYTE     $08,15              ; Break, DCD external interrupts
            .BYTE     $10,0               ; reset ext/status twice
            .BYTE     $10,0
            .BYTE     1,$FF               ; w/req pin configuration
            .BYTE     $0A,9               ; enable interrupts, status in low bits
InitLth     .EQU      *-InitData

InitSCC     LEA       InitData,A3         ; get pointer to init data
            MOVEQ     #InitLth,D1         ; and init length

InitSCC1    MOVE      SR,-(SP)            ; save interrupt state
            MOVEM.L   A0-A2,-(SP)

            MOVE.W    SCCOffset(A2),D2; get SCC offset
            MOVEM.L   SCCRd,A0-A1         ; and get SCC addresses
            ADD.W     D2,A0               ; add in offset
            ADD.W     D2,A1

            ORI       #$0300,SR           ; disable all but debug interrupts
            MOVE.B    (A0),D2             ; read to make sure SCC is sync'ed up
            LEA       SCCReset(A2),A2 ; point to data we will need

NextReg     MOVE.W    (A3)+,D0            ; get next init data, reg
            TST.B     D0
            BPL.S     @1                  ; bit 7=1 means get variable init data
            MOVE.B    (A2)+,D0            ; get variable data
            ROR.W     #8,D0               ; adjust to [data][ptr] format

@1          MOVE.B    D0,(A1)             ; write register pointer
            ROR.W     #8,D0
            MOVE.L    (SP),(SP)           ; delay
            MOVE.B    D0,(A1)             ; write register
            SUBQ.W    #2,D1
            BGT.S     NextReg

            MOVE.B    (A0),D2             ; read SCC register 1
            LSL.B     #2,D2               ; use this data to set CTS flag
```

```
                MOVEM.L  (SP)+,A0-A2
                SMI      CTSFlag(A2)

                RTE                          ; restore interrupt state and return

;--------------------------------------------------------------------------------
;
; Routine:     Close routines
;
; Arguments:      A1 (input)   --  DCE pointer for this driver
;
; Function:    These are the Close routines for the SCC async-mode drivers;
;             the SCC channel is reset and configured for only external/status
;              DCD interrupts and interrupt vectors are replaced with
;             the default RTE vector.  Input drivers simply RTS.
;
;             The master interrupt enable register should not have to be
;              written with $0A again since only the channel is reset (not
;            the entire SCC).
;
;--------------------------------------------------------------------------------

ResetData     .BYTE     9,$FF          ; reset SCC channel
              .BYTE     $08,15         ; only DCD (mouse) ext/sts interrupts
              .BYTE     $10,0          ; reset ext/status twice
              .BYTE     $10,0
              .BYTE     $01,1          ; only external/status interrupts
              .BYTE     $45,4          ; x16, parity even
              .BYTE     $50,11         ; br gen clk to rcvr, xmitter
              .BYTE     $01,14         ; enable baud rate generator
              .BYTE     $0A,9          ; enable interrupts, status in low bits
ResetLth      .EQU      *-ResetData

BClose        LEA       Lvl2DT,A4      ; get dispatch table address
              LEA       ExtStsDT,A5    ; and secondary dispatch table
              MOVE.L    PortBVars,A2   ; local variables address

ABClose       ; should have a wait here for last character to clear the buffer
              ; except that close is not really used in the normal course of
              ; operation . . .

              LEA       ResetData,A3
              MOVEQ     #ResetLth,D1
              BSR.S     InitSCC1
```

```
            LEA       BlnClose,A3
            MOVE.L    A3,(A5)          ; and reinstall default interrupt
            ADDQ.L    #8,A4            ;  receivers (just RTS)
            MOVE.L    A3,(A4)+
            MOVE.L    A3,(A4)

            MOVE.L    DCtlStorage(A1),A0; get storage handle
            _DisposHandle              ; get rid of it
            CLR.L     DCtlStorage(A1)  ; without a trace
            CLR.L     (A2)
BlnClose    RTS

;_____________________________________________________________________
;
; Routine:    Status routines
;
; Arguments:     A0 (input)  --  pointer to Status parameter block:
;              (0) Header
;               (12) Completion routine
;             (16) IOResult code
;              (24) RefNum
;             (26) Opcode
;               (28) Parameters
;            A1 (input)  --  DCE pointer for this driver
;
; Function:   For operation code 8, 3 words of status information are
;          returned:
;               (28) cumulative errors:
;                    bit 0  = soft overrun (local buffer overflow)
;                   bit 4 = parity error
;                   bit 5 = hard overrun error
;                   bit 6 = framing error
;              (29) 80 = input flow shut off
;            (30) read command pending flag
;             (31) write command pending flag
;            (32) XOFF flag
;            (33) CTS flag
;
;          For operation code 2, 1 longword of status information is
;         returned:
;             (28) buffered bytes available
;
```

```
;           Other opcodes are not implemented.
;
;_____________________________________________________________________

BStatus     MOVE.L   PortBVars,A2       ; local variables address

Status      LEA      CSCode(A0),A0      ; get pointer to return parameters
            MOVEQ    #StatusErr,D0      ; assume unimplemented code error
            MOVEQ    #2,D1
            SUB.W    (A0)+,D1           ; opcode 2?
            BNE.S    @1                 ; br if not

; opcode 2 is a standard system code used to return a longword count of
; available bytes in the driver buffer (if any)

            BSR      GetBufRegs
            BSR      GetBufCnt

            CLR.W    (A0)+              ; high word is zero . . .
            MOVE.W   D0,(A0)            ; load bytes-in-buffer parameter
            BRA.S    @2                 ; exit with 0 result code

@1          ADDQ.W   #6,D1              ; opcode 8?
            BNE.S    StsExit            ; exit with error if not
            MOVE.W   AsyncErr(A2),(A0)+        ; get errors,  flow off flag
            CLR.B    AsyncErr(A2)       ; and reset indicators

            MOVE.L   ReadCmd(A2),(A0)+       ; read, write, CTS, XOFF flags
@2          MOVEQ    #0,D0              ; set zero error code
StsExit

toIODone    MOVE.L   JIODone,-(SP)      ; go to IODone (A1 must point
            RTS                         ; to the DCE and D0 = IOResult)

;_____________________________________________________________________
;
; Routine:      Control routines
;
; Arguments:       A0 (input)  --  pointer to Control parameter block:
;                 (0) Header
;                  (12) Completion routine
;                (16) IOResult code
;                 (24) RefNum
;                (26) Opcode
```

```
;                 (28) Parameters
;               A1 (input)  --  DCE pointer for this driver
;
; Function:     These are the Control routines for the SCC async-mode drivers.
;
;           Operation code 1 is the standard KillIO call: pending read
;           and write requests are reset and any buffered bytes discarded.
;
;           For operation code 8, the appropriate SCC channel is reset and
;            reinitialized according to the new defaults.  If a parameter
;            is zero, the current corresponding value will not be changed:
;
;             (26) [$0008]
;                   (28)  [V][V][W][W][X][X][Y][Y]  [Z][Z][Z][Z][Z][Z][Z][Z]
;
;               VV = 1,2,3,   for 1,1.5,2 stop bits
;               WW = 0,1,2,3  for no,odd,no,even parity
;                XX = 0,1,2,3, for 5,7,6,8 data bits
;             YY = high byte of baud rate constant, low 2 bits
;            ZZZZZZZZ = low byte of baud rate constant
;
;            Opcode 9 is used to install a new buffer for input buffering (this control
;           command may be given to either the input or output driver):
;          a pointer to the buffer and the buffer length are passed.  The
;           async driver uses this buffer to store input characters when
;          no input user request is pending:
;
;             (26) [$0009]
;             (28) [pointer to buffer]
;             (32) [buffer length] - 2 bytes
;
;         Opcode 10 is used to specify handshake options and other
;          miscellaneous controls:
;
;            (26) [$000A]
;           (28) enable XON/XOFF output flow control if non-zero
;          (29) enable CTS output handshake if non-zero
;          (30) XON char for software handshake
;          (31) XOFF char for software handshake
;            (32) errors which cause abort of input requests (1 for abort):
;                 bit 4 = abort on parity error
;                 bit 5 = abort on overrun error
;                 bit 6 = abort on framing error
```

```
;           (33) status changes which cause events to be posted
;                bit 7 = post event on break status change
;                bit 5 = post event on CTS change
;            (34) enable XON/XOFF input flow control if non-zero
;
;           Opcode 11 is used to transmit a control character
;
;            (26) [$000A]
;           (28) the character
;_____________________________________________________________________

BControl     MOVE.L   PortBVars,A2       ; local variables address

Control      LEA      CSCode(A0),A0      ; get parameters
             MOVE.W   (A0)+,D1           ; get opcode
             SUBQ.W   #1,D1              ; opcode 1?
             BNE.S    CtlConfig          ; branch if not
             CLR.W    ReadCmd(A2)        ; clear ReadCmd and WriteCmd flags
             CLR      BufOutIn(A2)       ; InDex>Outdex
             CLR.B    ContData(A2)       ; clear continuous data input
             CLR.L    BufIndex(A2)       ; get rid of any buffered bytes
             RTS                         ; special direct return

CtlConfig    SUBQ.W   #7,D1              ; opcode 8?
             BNE.S    CtlBuffer          ; branch if not
             MOVE.W   (A0)+,D4           ; get word of configuration data
             CMP.W    lastSetup(A2),D4   ; same setup?
             BEQ.S    CtlGood            ; then just exit
             BSR      ToInitSCC          ; go initialize

CtlGood      MOVEQ    #0,D0              ; IOResult=0 for success
CtlExit      BRA.S    toIODone           ; and go to IODone (A1 must point
                                         ;  to the DCE and D0 = IOResult)

CtlBuffer    SUBQ.W   #1,D1              ; opcode 9?
             BNE.S    NewOptions         ; br if not
             PEA      CtlGood            ; end up here
             CLR.L    BufIndex(A2)       ; clear in and out indices

             MOVE.L   (A0)+,A4
             MOVE.W   (A0),D1            ; if zero, revert to our own buffer
             BNE.S    InstlABuf          ; otherwise, ring in a new one . . .

InstlLBuf    LEA      LocalBuf(A2),A4    ; use our meager local buffer for now
```

```
            MOVE.W   #LclBufSize,D1

InstllABuf  LEA      InBufPtr(A2),A3
            MOVE.L   A4,(A3)+
            MOVE.W   D1,(A3)+
            LSR.W    #2,D1           ; set new bufmin and bufmax values
            MOVE.W   D1,(A3)+        ; BufLow
            MOVE.W   D1,(A3)+        ; BufHigh
            CLR.B    BufOutIn(A2)    ; InDex>Outdex
            RTS

NewOptions  SUBQ.W   #1,D1           ; opcode 10?
            BNE.S    SendChar        ; br if not
            ADD      #SWHS,A2        ; point to options
            MOVE.L   (A0)+,(A2)+     ; set new SWHS, HWHS, XON/XOFF chars
            MOVE.W   (A0)+,(A2)+     ; errors which cause aborts (Options), and
                                     ;  status changes on which to post events
            MOVE.B   (A0)+,(A2)      ; set new InSWHS
            BRA.S    CtlGood         ; exit ok

SendChar    SUBQ.W   #1,D1           ; opcode 11?
            BNE.S    CtrlErr         ; br if not
            MOVE.W   (A0)+,D0        ; Set character to send
@1          BSR      WrSetUp         ; set up A3, see if xmit Buffer is empty
            BEQ.S    @1              ; exit if transmit buffer is full
            ADD.L    #SCCWrite,A3    ; form SCC base write address
            MOVE.B   D0,SCCData(A3)    ; start it out
            BRA.S    CtlGood         ; exit ok


CtrlErr     MOVEQ    #ControlErr,D0
            BRA.S    CtlExit         ;go IODone

;___________________________________________________________________
;
; Routine:    Prime routines
;
; Arguments:     A0 (input)  --  pointer to Control parameter block:
;               (0) Header
;                (12) Completion routine
;               (16) IOResult code
;                (24) RefNum
;               (26) Opcode
```

```
;               (28) Parameters
;             A1 (input)  --  DCE pointer for this driver
;
; Function:   The function call is deciphered here: aRdCmd for Input
;                                               aWrCmd for Output
;
; Other:
;_______________________________________________________________________

BPrime       MOVE.W   ioTrap(A0),D0     ; Get trap
             CMPI.B   #aRdCmd,D0        ; Is it a Read?
             BEQ      BInPrime          ; Branch to Input Routine
             CMPI.B   #aWrCmd,D0        ; Is it a Write?
             BEQ      BOutPrime         ; Branch to Output Routine
             RTS



;_______________________________________________________________________
;
; Routine:    Output Prime routines
;
; Arguments:     A0 (input)  --  pointer to Prime parameter block:
;
;             A1 (input)  --  DCE pointer for this driver
;
; Function:   The first character is loaded into the SCC.
;
; Other:
;_______________________________________________________________________

BOutPrime    MOVE.L   SCCRd,A2          ; get SCC read address
             MOVE.L   A2,A3
             ADD.L    #SCCWrite,A3      ; form SCC base write address

FetchNext    MOVE.L   JFetch,A0
             JSR      (A0)              ; get it from user buffer
Wrlp         BTST     #TXBE,(A2)        ; check for TBE
             BEQ.S    Wrlp              ; exit if transmit buffer is still full

             MOVE.B   D0,SCCData(A3)    ; start it out
             TST.W    D0                ; only one?
             BMI.S    resetWrCmd        ; if so, we've had a good finish
             BRA      FetchNext
```

```
resetWrCmd    MOVE.L    PortBVars,A2        ; local variables address
              CLR.B     WriteCmd(A2)        ; no more pending output

GoodFinish    MOVEQ     #0,D0               ; IOResult=0 for success
              BRA.S     toIODone            ; and go to IODone (A1 must point
                                            ;  to the DCE and D0 = IOResult)

WrSetUp       MOVE.L    SCCRd,A3            ; get SCC read address
              ADD       SCCOffset(A2),A3
              BTST      #TXBE,(A3)          ; check for TBE
@1            RTS

SendNextChar
              MOVE.L    OutDCE(A2),A1       ; get DCE pointer for Fetch
              MOVE.W    SR,-(SP)            ; fake out prime routine
              BRA.S     FetchNext

ContOut       CLR.B     XOFFlag(A2)         ; come here if we got an XON
ContOut1      TST.B     WriteCmd(A2)        ; come here for XON or CTS high
              BNE.S     SendNextChar        ; if no output request, just exit
OutputRTS     RTS                           ; otherwise, wait for the interrupt

;_______________________________________________________________________
;
; Routine:      Input Prime routines
;
; Arguments:     A0 (input)  --  pointer to Prime parameter block:
;               A1 (input)   --  DCE pointer for this driver
;
; Function:     Get characters from local buffer, if any (satisfy request
;               possibly). Note the input request.
;              Seperates the sargen channel info from data.
;               Checks if Sargen has sent terminated. Returns -10 as result
;             code if has.
;
;_______________________________________________________________________


BInPrime      MOVEM.L  A2-A3,-(SP)
              MOVE.L    PortBVars,A2        ; local variables address

InPrime       BSR       GetBufRegs          ; load D1, D2, D3, and A3
```

```
FeedBufLoop  CMP       D1,D2               ; indices equal means we're through
             BEQ.S     InFinish
             MOVE.B    0(A3,D2.W),D0       ; get the next byte
             ADDQ      #1,D2               ; bump outdex
             CMP       D3,D2               ; wrap it if we're at buffer limit
             BNE.S     @1
             MOVEQ     #0,D2

@1           BSR.S     toStash             ; stash into user's buffer
             TST.W     D0                  ; done with request?
             BPL.S     FeedBufLoop         ; go again if we're not done

InFinish     MOVE      D2,BufOutdex(A2); update out index
             MOVEQ     #0,D0               ; 0 for Good finish
             MOVEM.L   (SP)+,A2-A3
             MOVE.L    JIODone,-(SP)       ; go to IODone (A1 must point
             RTS                           ; to the DCE and D0 = IOResult)

toStash      MOVE.L    JStash,-(SP)        ; push the vector
             RTS

GetBufRegs   MOVE.L    InBufPtr(A2),A3
             MOVE.W    BufSize(A2),D3
             MOVEM.W   BufIndex(A2),D1-D2      ; get BufIndex and BufOutdex
             RTS

GetBufCnt    MOVE.W    D1,D0               ; BufIndex
             SUB.W     D2,D0               ; minus BufOutdex
             BHS.S     @1                  ; br if positive value
             ADD.W     D3,D0               ; add BufSize to make it positive
@1           RTS

;_______________________________________________________________
;
; Routine:     RXIntHnd
;
; Arguments:      A0 (input)  --  port A/B control read address
;              A1 (input)  --  port A/B control write address
;
; Function:     This routine handles SCC receiver interrupts for
;          both ports; the data is read and stashed, IODone called
;            if necessary. This is done by first identifying what is the
;            first input character:
```

```
;           $11     XON character
;           $12     XOFF character
;           $03     Data recieving - 128 bytes
;           $04     Data recieving - 256 bytes
;           $05     Data recieving - 512 bytes
;           $06     Data recieving - 1024 bytes
;           $80-$FF Data recieving - 1-128 bytes
;           $07     Continous data input
;           $60     Sargen terminate processing (twice)
;______________________________________________________________

RBIntHnd    MOVEM.L  A0-A3,-(SP)
            MOVE.L   PortBVars,A2       ; get pointer to local variables

RXIntHnd    MOVE.B   SCCData(A0),D0     ; get the data byte

            BCLR     #7,D0              ; Char between $80-FF - <= 127 bytes?
            BNE.S    IData              ; if yes, branch to collect data

            CMP.B    #$03,D0            ; 128 bytes to input?
            BNE.S    @2                 ; if not, branch
            MOVE.W   #$80,D0            ; set register to 128
            BRA.S    IData              ; branch to collect data

@2          CMP.B    #$04,D0            ; 256 bytes to input?
            BNE.S    @3                 ; if not, branch
            MOVE.W   #$0100,D0          ; set register to 255
            BRA.S    IData              ; branch to collect data

@3          CMP.B    #$05,D0            ; 512 bytes to input?
            BNE.S    @4                 ; if not, branch
            MOVE.W   #$0200,D0          ; set register to 512
            BRA.S    IData              ; branch to collect data

@4          CMP.B    #$06,D0            ; 1024 bytes to input?
            BNE      NotData            ; if not, branch
            MOVE.W   #$0400,D0          ; set register to 1024

IData       MOVE.L   InBufPtr(A2),A3    ; Get buffer pointers
            MOVE.W   D4,-(SP)           ; preserve these registers
            MOVE.W   Bufsize(A2),D4
            MOVE.W   BufIndex(A2),D1    ; get BufIndex and BufOutdex
```

```
InLoop      BTST      #RXBF,(A0)        ; Rx char available?
            BEQ.S     InLoop            ; delay if not

            MOVE.B    SCCData(A0),D3    ; get the byte
            MOVE.B    D3,D2
            ANDI.B    #$F0,D3           ; rotate in channel data
            ORI.B     #$80,D3           ; set upper bits
            MOVE.B    D3,0(A3,D1.W)     ; stash the byte

            ADDQ.W    #1,D1             ; update BufIndex
            CMP.W     D4,D1
            BNE.S     @20               ; br if not at the end
            MOVEQ     #0,D1             ; otherwise, reset to 0

@20         ANDI.B    #$0F,D2
            MOVE.B    D2,0(A3,D1.W)     ; stash the byte

            ADDQ.W    #1,D1             ; update BufIndex
            CMP.W     D4,D1
            BNE.S     @15               ; br if not at the end
            MOVEQ     #0,D1             ; otherwise, reset to 0

@15         BTST      #RXBF,(A0)        ; Rx char available?
            BEQ.S     @15               ; delay if not

            MOVE.B    SCCData(A0),0(A3,D1.W); stash the low byte

            ADDQ.W    #1,D1             ; update BufIndex
            CMP.W     D4,D1
            BNE.S     @16               ; br if not at the end
            MOVEQ     #0,D1             ; otherwise, reset to 0

@16         SUBQ.W    #2,D0             ; Decrement counter
            BNE.S     InLoop            ; branch if not zero

@10         MOVE.W    D1,BufIndex(A2)   ; update index
            MOVE.W    (SP)+,D4
            BRA       FinInput          ; and exit

NotData     CMP.B     #$6F,D0           ; cycle counter data to input?
            BNE.S     @6                ; if not, branch
            BSR       StashByte         ; Stash to inform Pascal
            BRA       FinInput          ; and exit
```

```
@6          CMP.B    #$60,D0          ; A Sargen terminated character?
            BNE.S    @1               ; if not, branch
            BSR      StashByte        ; Stash to inform Pascal
            BRA      FinInput         ; and exit

@1          CMP.B    #18,D0           ; how about an XOFF?
            BNE.S    @7               ; if not, branch
            ST       XOFFlag(A2)      ; if so, then note it
            BSR      StashByte        ; Stash to inform Pascal
            BRA      FinInput         ; and exit

@7          CMP.B    #17,D0           ; how about an XON?
            BNE.S    FinInput         ; if not terminate
            BSR      StashByte        ; Stash to inform Pascal
            BRA      FinInput         ; and exit

StashByte   MOVE.L   InBufPtr(A2),A3 ; Get buffer pointers
            MOVE.W   BufSize(A2),D3
            MOVEM.W BufIndex(A2),D1-D2      ; get BufIndex and BufOutdex

SBData      MOVE.B   D0,0(A3,D1.W)    ; stash the byte

            ADDQ.W   #1,D1            ; update BufIndex
            CMP.W    D3,D1
            BNE.S    @12              ; br if not at the end
            MOVEQ    #0,D1            ; otherwise, reset to 0

@12         CMP.W    D2,D1            ; hit the output index?
            BNE.S    @13              ; br if not
            BSET     #SoftOR,AsyncErr(A2)   ; note the soft overrun
            SUBQ.W   #1,D1            ; reset Index

@13         MOVE.W   D1,BufIndex(A2)  ; update index
            RTS

FinInput    MOVE.B   #$20,(A1)
            MOVEM.L  (SP)+,A0-A3
            RTS

;_____________________________________________________________
;
; Routine:    SCIntHnd
;
```

```
; Arguments:      A0 (input)  -- port A/B control read address
;                 A1 (input)  -- port A/B control write address
;
; Function:     This routine handles SCC special condition interrupts:
;               these occur when an input character is received that has
;               a parity error, framing error, or causes an overrun.
;               If the option is set to abort on the error, the character
;               is discarded and the input request (if any) aborted; otherwise,
;               the error is noted and the character buffered as usual.
;
;_______________________________________________________________________

SCBIntHnd               LEA     PortBVars,A3
        ; get appropriate variables (port A)

SCIntHnd        MOVE.B  #1,(A1)             ; point to error reg
                MOVE.L  (A3)+,A2            ; get local variables pointer
                MOVE.L  (A3),A3             ; and DCE pointer (delay, too)
                MOVE.B  (A0),D1             ; read the error condition

                MOVEQ   #$70,D3             ; form $70 mask
                AND.B   D3,D1               ; isolate error bits
                OR.B    D1,AsyncErr(A2)     ; accumulate errors (delay, too)
                MOVE.B  SCCData(A0),D0      ; get the data byte

                MOVE.B  Options(A2),D2      ; get abort options
                MOVE.B  #$30,(A1)           ; reset the error flag
                AND.B   D1,D2               ; check abort options
                BEQ.S   InputRTS            ; then just discard the character

                TST.B   ReadCmd(A2)         ; if we have no pending read command
                BEQ.S   InputRTS            ; then just discard the character

                MOVEQ   #RcvrErr,D0         ; otherwise, note the error

RdReqDone       MOVE.L  A3,A1               ; DCE pointer
                CLR.B   ReadCmd(A2)         ; no longer a read request pending
                BRA     toIODone            ; and go to IODone (A1 must point
                                            ;  to the DCE and D0 = IOResult)
InputRTS        RTS

;_______________________________________________________________________
;
; Routine:      ExtIntHnd
```

```
;
; Arguments:      A0 (input)   -- port A/B control read address
;                 A1 (input)   -- port A/B control write address
;                 D0 (input)   -- SCC read reg 0 value
;                 D1 (input)   -- SCC read reg 0 changed bits
;
; Function:       This routine handles SCC external/status interrupts for
;                 both ports; mouse (DCD) interrupts are passed along to the
;                 mouse interrupt handler in CrsrCore. Only Break/Abort and CTS
;                 external interrupts are enabled (besides DCD).
;
;                 Note that CTS low in read reg 0 currently means that the
;                 hardware handshake line is asserted which means 'ok to transmit'.
;_______________________________________________________________________________

ExtBIntHnd  LEA       PortBVars,A3        ; get appropriate variables - port A

ExtIntHnd   MOVE.L    (A3)+,A2            ; get pointer to local variables
            MOVE.L    (A3),A3             ; and DCE ptr in case of break abort

            MOVE.B    D1,D2               ; changed bits
            AND.B     postOptions(A2),D2; post this change?
            BEQ.S     @0                  ; br if not

            MOVEM.L   D0/A0,-(SP)         ; preserve these registers
            MOVE.W    #IODrvrEvt,A0
            ASL.W     #8,D0               ; make room for 'changed' values
            MOVE.B    D1,D0
            SWAP      D0                  ; make room for driver refnum
            MOVE.W    DCtlRefnum(A3),D0
            _PostEvent                    ; and post the event
            MOVEM.L   (SP)+,D0/A0

@0          TST.B     D1                  ; see if it's a change in break status
            BMI.S     @1                  ; branch if it was a break interrupt
            LSL.B     #2,D0               ; must be CTS change
            SMI       CTSFlag(A2)         ; set flags according to CTS
            BPL       ContOut1            ; if freshly asserted, continue output
            RTS                           ; if not, exit for now

@1          TST.B     D0                  ; check break level
            BMI.S     @2                  ; if it's asserted, terminate any input
            MOVE.B    SCCData(A0),D0
```

```
        ; otherwise (end of break), discard null
                RTS                                 ; and return

@2              MOVEQ   #BreakRecd,D0               ; note the break
                TST.B   ReadCmd(A2)                 ; read request pending?
                BNE.S   RdReqDone                   ; if there is one, jump to IODone
ExtIntRTS       RTS                                 ; otherwise, just return

                .END


@1              TST.B   D0                          ; check break level
                BMI.S   @2                          ; if it's asserted, terminate any input
                MOVE.B  SCCData(A0),D0              ; otherwise (end of break), discard null
                RTS                                 ; and return

@2              MOVEQ   #BreakRecd,D0               ; note the break
                TST.B   ReadCmd(A2)                 ; read request pending?
                BNE.S   RdReqDone                   ; if there is one, jump to IODone
ExtIntRTS       RTS                                 ; otherwise, just return

                .END
```