DEVELOPMENT AND INTEGRATION OF NOVEL SMART SACRIFICIAL SENSORS INTO CRITICAL WEAR SITUATIONS

STUART YAXLEY

A thesis submitted in partial fulfilment
of the requirements of De Montfort University
for the degree of Doctor of Philosophy

May 1997

De Montfort University

ABSTRACT

A novel smart wear sensor is developed for use within the quarrying industry. Aggregate is produced by crushing the mined rock using jaw and cone crushers. Jaw crushers are used initially, followed by cone crushers for the secondary and tertiary stages of the mill. Cone crushers use the principle of grinding rock between an eccentrically rotating mantle and a stationary concave liner. The rock is fed into the top of the crusher and the output diameter is set by the discharge setting, or nip angle at the lower half of the crusher. During crushing, the manganese liners are susceptible to a high degree of wear. A system to monitor the performance of the cone crushers is described, with specifically designed sacrificial sensors which are embedded into the liners.

A series of wear sensors are developed which use resistive, capacitive and conductive (discrete level) techniques. The sensors are housed in specifically designed steel inserts, with an epoxy resin used to cement the sensor into the housing. Laboratory strength and grinding tests simulate the typical quarrying environment. Field tests of the sensors embedded in the crusher are performed to verify the laboratory results. Integrating an electronic interface to the sensors with signal conditioning enables a more flexible sensor to be developed. The addition of a microcontroller with a serial bus interface allows for a network of smart wear sensors. Embedding a series of smart wear sensors enables the contours of the liners to be analysed during the crushing process, and thus the suitability of the crusher set up. Furthermore, remote smart wear sensors are developed which include a telemetry link for the rotating liner.

The smart wear sensors are integrated into a full condition monitoring system which includes a Knowledge Based System, a wear model specific to the quarry being mined and a distributed control and acquisition system. By implementing such a system, distinct benefits can be noted.

- ♦ Accurate knowledge of the liner wear allows the mill manager to extensively run the crusher leaving the minimum amount of manganese before replacement.
- ♦ Corrective measures to improve the shape and diameter of the output can be made automatically whist the crusher is in operation.

♦ An improvement in quarry performance by reduced downtimes, reduced recirculation and a decrease in the power consumed.

ACKNOWLEDGEMENTS

NOTATIONS

L	Length of Surface Mount Device
W	Width of Surface Mount Device
T	Thickness of Surface Mount Device
t	Width of Surface Mount Device Contact
l_m	Length of Multiple Resistive Sensor
r_a	Resistance of Single Surface Mount Resistor
n	Number of Surface Mount Resistors
R_1	Potential Divider Resistor Value - Surface Mount Resistive Sensor
r_m	Resistance of Surface Mount Resistive Sensor
V_s	Voltage Supply
V_{out}	Voltage Output
l_w	Length of Welwyn Single Resistive Sensor
r_w	Resistance of Welwyn Single Resistive Sensor
r_{init}	Start Resistance of Welwyn Single Resistive Sensor
m	Single Resistive Sensor Multiplication Factor
f	Frequency
R_A	555 Timer Circuit Timing Resistance
R_B	555 Timer Circuit Timing Resistance
C_s	Capacitive Sensor Capacitance
CT10	Microcontroller Timer Register
p	Set Number of Samples
C_{s_ave}	Average Capacitive Sensor Capacitance
n_p	Total Number of Surface Mount Capacitors
C_{smd}	Average Capacitance for Each Surface Mount Capacitor
n_{smd}	Number of Surface Mount Capacitors
l_s	Length of Capacitive Sensor
t_{smd}	Thickness of Surface Mount Capacitor
f_{CLK}	Microcontroller Internal Clock Frequency
C_k	Capacitive Sensor Capacitance, Modified Martin Oscillator
R	Feedback Resistance, Modified Martin Oscillator

C_{off} Offset Capacitance, Modified Martin Oscillator

t_d Switching Time Delay, Modified Martin Oscillator

T_m Output Period, Modified Martin Oscillator

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PLATES

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

1.1 The Growth of the Sensor Industry

The importance of data acquisition systems within industry has vastly increased over the last decade. The quantity of real time information available not only improves the reliability of modern day complex systems, but also gives the user more knowledge of how the system is operating. The development of application specific sensors has advanced the production of novel sensors, providing industry with standard data acquisition tools for use within a wide variety of situations.

During the 90's, the sensor industry is estimated to have the potential for growth from $5 billion in 1990 to around $13 billion by the end of the decade. Figure 1.1 shows the distribution of this market throughout the world, with the US, Japan and Europe with similar divisions of the market (1995).
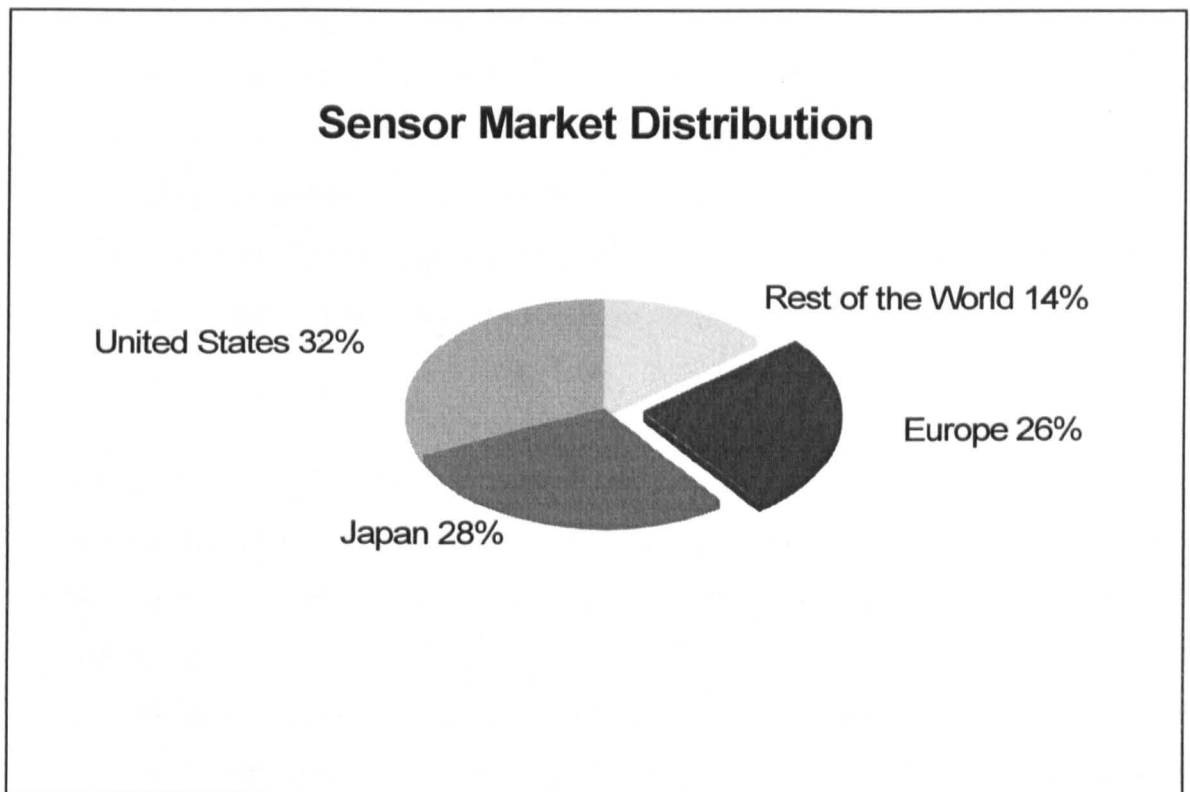
Figure 1.1 - Sensor Market Distribution

The rapid growth in the sensor industry is mainly due to the development of silicon sensors, in particular the utilisation within the automotive industry which holds 46% of the total sensor market. Due to the need for more informative and safer vehicles, intelligent automotive sensors have been developed for a wide variety of parameters including pressure, acceleration, velocity, temperature and position. These have provided the customer with sensors for devices such as air bags, temperature control and exhaust emission monitoring.

1.2 Sensor Definition

A sensor can be defined as "..a device for sensing a physical variable of a physical system or an environment" [Iyengar et al, 1995]. The sensor is an interface between the physical properties of the real world, and the electrical parts of the developed system. This interface converts a change in the physical parameters into a predictable electrical signal. Six domains [Lion, 1969] cover the complete range of physical and chemical quantities measured. These are:-

- ◆ Thermal (temperature, heat, heat flow...)
- ◆ Mechanical (force, pressure, velocity, acceleration, position, wear...)
- ◆ Chemical (concentration of a certain material, composition, reaction rate...)
- ◆ Magnetic (magnetic field intensity, flux density, magnetisation...)
- ◆ Radiant (electromagnetic wave intensity, wavelength, polarisation, phase...)
- ◆ Electrical (voltage, current, charge...)

1.3 Integration of Sensors and Electronics

One of the main features of newly developed sensors, has been the complexity of the designs, allowing for a more compact transducer. The miniaturisation of certain sensors has allowed analysis in previously un-reachable areas. Throughout the sixties to date, electronic components have dramatically reduced in size, the major contributing factor being the introduction of surface mount technology for a whole host of electronic components. However, the development of the microprocessor has been the forerunner in the electronics market, with the total amount of transistors per area of silicon increasing exponentially with

time. The complexity and miniaturisation of the microprocessor, has not only forwarded the developments in the computer world, but has also allowed the integration with sensors to provide a new intelligent sensor. This has been 'tagged' the smart sensor.

1.4 Smart Sensors and Distributed Systems

The evolution of sensor design has enabled systems with intelligent nodes to be created. In the first instance, industrial sensor signals were carried back to a central core in which a fault indicator is lit. This notified the user of the problem, and manual changes with human interaction were made. The next step involved a central processor in which all of the sensors in the system report back to a central processing unit. This unit can inform the user of the state of the system and control actuators to complete the feedback system. This works fine for simplistic systems, however, if signal verification, accuracy and reliability are necessary then distribution of control and data acquisition must be employed. This is often termed a Distributed Sensor Network (DSN). A DSN is described [Iyengar et al, 1995] as ".. a set of spatially scattered, intelligent sensors designed to derive data from the environment, abstract relevant information from the data gathered, and to derive appropriate inferences from the information gained".

This has a number of advantages over the traditional 'single processor to multiple sensor' architecture. By integrating a processor with the sensor, the sensor node becomes more flexible. Installation is one of the immediate advantages. A two wire bus runs through the system, and any smart sensor can be clipped directly onto the bus. By using a single processor system, each sensor would use two or three wires trailing back to the central processor.

The flow of information is now bi-directional in that data can be received from the sensor and commands can be sent to the sensor. These may be sampling times, accuracy of measurement, calibration or a change in algorithm to extract the relevant information. This may be necessary if another sensor within the system is not functioning correctly.

Compensating sensors [Powner and Yalcinkaya, 1995] on each node can be used to finely tune the data before it is returned to the master processor. For example, temperature may

effect the sensor output, thus integrating a temperature sensor into the smart sensor enables the data to be automatically corrected.

Finally, DSN's are more fail-safe, in that control of the system can be passed to any node. This allows the system to continue running if a controller fails, giving the servicemen time to repair the damaged node.

1.5 Statement of the Problem

The work carried out in this thesis is particular to the quarrying industry, with the emphasis on cone crushers. The most significant problems that exist are:-

♦ The wear of the crushing members of a cone crusher is not accurately monitored.

♦ There is no method to automatically adjust the system for wear on the liners. The crusher is manually reset at timed intervals. Thus, irregular diameter aggregate is often output from the crusher.

♦ The total lifespan of the liners is not anticipated with regular service intervals used instead of wear knowledge to change the liners.

1.6 Objectives of this Work

The objectives of the research carried out in this thesis are:-

1. To develop a sensor capable of detecting real-time wear on the manganese liners of the cone crusher. An improvement of the current accuracy of 1mm is desired. The wear sensors will be developed into 'smart' sensors by way of microcontroller integration.

2. To introduce a distributed control and data acquisition system into the quarrying industry. The system must allow for direct expansion, and link the current sensors and actuators with the newly developed smart wear sensors through a serial bus.

3. To develop a full condition monitoring system by way of a Knowledge Based System (KBS). The developed KBS will be flexible in design, and will be able to link to any quarrying system. The KBS will provide total control of the cone crusher, and will notify the mill manager of the current condition including the state of wear inherent on the manganese liners. Furthermore, a fault diagnosis and on-line manual KBS will also be available.

1.7 Organisation of the Thesis

The thesis analyses the needs of the quarrying industry, and of similar industries in which this work would be a benefit. Also examined are system structures already utilised in alternative industries, with emphasis on the sensors used and methods for data acquisition and control. Comparisons with previously developed wear sensors are made, and the results of the developed wear sensor tests are described. With this in mind, the developed surrounding electronics and distributed control system are described. To enable remote data to be accessed, telemetry systems are analysed and a prototype system is developed which integrates the sensors, the microcontroller unit, telemetry and serial bus protocol.

Chapter 2 contains a review of previous work in the variety of related fields. The chapter reviews condition monitoring systems and related research. Wear sensing and measurement techniques from different industries are also analysed. This incorporates direct and indirect measurement, and on and off-line strategies. Finally, the chapter analyses the role and organisation of today's smart sensor, and looks at the individual modules, the fabrication and the typical environment of the smart sensor.

Chapter 3 details the environment in which the sensors are to be used, and looks at the particular machinery that the project is centred on, namely the cone crusher. Typical problems that occur in the production of aggregate are described.

In Chapter 4, the designs of each of the different wear sensors are described. The design constraints on the sensor are detailed with information on the sensor modules, signal conditioning circuits and the fabrication process.

The results of a variety of tests on each of the wear sensors are presented in Chapter 5. Compressive Strength and grinding laboratory tests are carried out on each of the sensors to simulate typical forces inherent in the cone crusher. Stability and temperature tests are used to simulate possible conditions to which the sensors may be subjected. Field tests followed, in which the sensors were embedded in the liners of the cone crusher. Data from the wear sensor was taken at regular intervals throughout the lifespan of the manganese liner.

Chapter 6 contains a detailed description of the developed system. Described is the interaction of the protocol between the smart sacrificial wear sensors and the distributed control and acquisition system. The chapter also examines the method for retrieving wear sensor information from the rotating liner using a telemetry link. The interaction with the modules to the Knowledge Based System provides a comprehensive man-machine interface. The construction of a smart sensor prototype with serial bus and telemetry interface is outlined.

In Chapter 7, the results are discussed and in Chapter 8, conclusions on the thesis are formed.

In Chapter 9, recommendations are given for further research..

The appendices contain software listings for the smart wear sensors and the serial bus development, as well as wiring diagrams for the developed systems.

CHAPTER 2
REVIEW OF PREVIOUS WORK

2. REVIEW OF PREVIOUS WORK

In reviewing the related previous work, this chapter examines research that is particular to each of the modules of the developed system, namely,

- Condition Monitoring and Knowledge Based Systems
- Wear Sensing and Measurement Techniques
- Smart Sensors
- Serial Bus/Distributed Control Systems
- Telemetry Systems

2.1 Condition Monitoring and Knowledge Based Systems

Condition monitoring is the measurement of a series of parameters within a system at a certain sample rate. Basic systems rely upon human periodic inspection of core components within the plant. Though this is cheap, it is dependent upon the skill of the inspector and poor inspections could produce catastrophic failures with large plant downtimes. The primary aim of condition monitoring systems is to detect early stages of deterioration of particular components within a system, and to analyse the consequences of component failure. Using these techniques, a more efficient method of determining when maintenance is necessary is provided. There are, however, times when a condition monitoring system is undesirable. This is typically when the failure of key components cannot be measured or predicted, and when the expense of the system cannot be justified against possible savings in maintenance labour [Armstrong, 1996].

Developed condition monitoring systems are very much application specific [Charbonneau & Plouffe, 1993][Grimmelius et al, 1995][Mechefske et al, 1994]. The complexity ranges from a multi-sensor condition monitoring and control system with integral neural networks and expert systems to a single sensor with alarm. A high volume of systems are designed with one application in mind, and in the majority of cases for single plants. There are, however, modules within a condition monitoring system that are common.

1. Data Acquisition - Information from a series of sensors provide the system with knowledge regarding the state of the system being monitored. Data is fed back, usually in analogue form, from the sensors with a difference in the measurand providing a change in voltage or current. The change can be amplified, enhanced and noise reduced by signal conditioning circuits.

2. Central Processing System - Sensor data is input into the central processor, and control signals are provided. This is in the form of a PLC, or a microcontroller. Often these are linked to provide a distributed control and data acquisition system.

3. Man-Machine Interface - The man-machine interface allows the user to analyse the information regarding the state of the system, and control the related inputs. This is usually in the form of mimic table with switches and lights, or a PC graphical screen.

4. Knowledge Based System (KBS) or Expert System - This is often used to interpret the information retrieved from the sensors and data or commands fed in by the user. Neural networks, fuzzy logic and other mathematical techniques can be employed to determine the state of the system.

Typical areas in which condition monitoring is used is fault prevention and fault diagnosis. System optimisation is often provided when using an expert system or KBS and allowing the system to be controlled by the resultant decisions. One of the main reasons for implementing condition monitoring systems is to prevent unexpected equipment failure, and thus reduce downtime whilst maintenance operations are undertaken. Furthermore, the forecasting of plant requirements can severely reduce maintenance needs [Swindells et al, 1994].

Condition monitoring of machine tools is an area in which there has been a considerable amount of research. A wide variety of strategies have been formulated using different monitoring techniques. Acoustic emission analysis and chip-form identification have been used [Kakade, 1994] to predict tool wear in an on-line condition monitoring system. A data acquisition system [Drake et al, 1995] developed in a machine tool condition monitoring system is quoted as *more than just a data logger and an array of sensors*. By encompassing a relational database and a methodology for analysing the data logging

requirements, the data acquisition system can be re-configured to operate with different machines. Artificial neural networks [Das, 1996] and fuzzy neural networks [Li & Elbestawi, 1996] are being employed to predict tool wear in on-line tool condition monitoring systems. In the latter system, force, vibration and spindle motor power signals are used as some of the signal inputs into the system to recognise the state of the tool. In tests using tools from sharp to severely worn, success rates of between 80%[1] and 94%[2] were produced.

Vibration analysis can be used in condition monitoring systems for various industries. An on-line condition monitoring system is described [Peck & Burrows, 1994] which analyses the state of rotating equipment used for the mining and petrochemical industry. Vibration data from the compressors, pumps and electric motors are used to identify patterns and trends within the vibration signals of the rotating components of the system. From this, on-line predictions of component wear and thus preventative maintenance is possible. A condition monitoring system which is now installed at Pirelli [Hills, 1996] analyses the vibration spectrum to detect a defective rolling element bearing. Using these condition monitoring techniques, direct savings of $55,000 are quoted, with an increase in machine availability from 335 days to 350 per annum.

The addition of qualitative information supplied by the supervisors of the process is an important part in the design of a condition monitoring system. This is often input in the form of a Knowledge Based System. Component failure is often determined by experienced plant supervisors. The utilisation of this information as well as on-line data acquisition and off-line data management of analytical laboratory results have provided waste water treatment plants with a knowledge based condition monitoring system [Serra et al, 1997].

Knowledge Based Systems have been developed for plants within the comminution sector since 1983. Starfield et al, [1983] developed a KBS which was used as a teaching aid for undergraduate engineering students. The utilisation of KBS in this and similar industries

---

[1] different samples used for learning and classification

[2] same samples used for learning and classification

was then enhanced by the Canadian CANMET symposium in 1987 (Knowledge Based Systems in Mineral Processing). Typical applications included a diagnostic programme for a cement clinker grinding circuit [Vanderstichelen, 1987], and an advisory system for operators at a zinc refinery [Ghibu and Dupuis, 1987]. Since CANMET, there has been a variety of Knowledge Based Systems developed using different KBS shells within the minerals industry.

Meech [1990] used the COMDALE /X KBS development shell from Comdale Technologies to develop a KBS for plant simulation and for the assessment of rock mass rating.

A real-time KBS with input/output, graphics and database utilities was developed for Brenda Mines [Spring and Edwards, 1989]. This used the SUPERINTENDENT KBS shell from Heuristics Ltd.

Interaction with a distributed control system was demonstrated at Brunswick Mining and Smelting [Anon, 1990]. The G2 real time KBS developed by Gensym Corporation was used to control a grinding circuit.

The implementation of a KBS using the RTEXPERT shell from Control International has produced substantial savings in the control of a flotation circuit [Hales, 1989]. Furthermore, the KBS was developed in only two months, a lead time considerably shorter than those associated with traditional control systems.

An aid to designing and simulating rock crushing circuits has been developed [Bearman et al, 1990] which uses the CRYSTAL shell from Intelligent Environments Ltd. CRYSTAL is based on the C language rather than early systems based around LISP and is a popular choice of KBS development shell.

A developed KBS is used as a diagnostic tool for the analysis of coal shearing equipment [DTI, 1990]. The consultative KBS, known as SHEARER, uses the SAVOIR shell supplied by ISI and is used by British Coal.

Underground extraction of ore is an area in which professional judgement is often necessary to determine the best mining procedure. A prototype KBS has been developed [Mutagwaba and Terezopoulos, 1994] called MINEXPERT, which analyses the factors which effect the choice of method implemented.

The choice of KBS shell used is often determined by the ease of interfacing with conventional software packages. Batanov et al [1993], uses the EXSYS shell for

maintenance management of a rotary screw type air compressor. Interaction with dBase III Plus and Turbo Pascal version 6.0 provides the KBS with an increased flexibility.

2.2 Wear Sensing and Measurement Techniques

The development of a dedicated wear sensor to analyse the state of a system component throughout its particular stages of wear is not a new concept. Previous research in this field is divided into direct and indirect, on-line and off-line measurement. It is not uncommon for systems to estimate the inherent wear in abrasive areas using mathematical modelling and surrounding sensors (indirect measurement). There are, however, a few examples in which the state of wear is critical to the user and patented sensors are being utilised.

2.2.1 Sensor Development and Wear Analysis for Automotive Braking Systems

The automotive industry is an industry which has enjoyed significant growth in terms of sensor integration. Due to the increase in integrated electronics systems, the sensor content of an average passenger car is expected to grow from $70 per car in 1990 to $300 per car in the year 2000 [Smith, 1993].

One of the most critical automotive parts, is the braking system. The determination of wear for braking systems has been subjected to a high degree of progress in the last decade. Patents have been administered for a wide variety of wear sensor systems.

Traditional methods to analyse brake liner wear is to remove the wheel and physically examine the liner. In the last decade, systems have been developed to inform the driver of a worn brake liner. A simple discrete level system is described [Vasilow et al, 1986] in which a signal is provided when the liner wears beyond a certain threshold. Figure 2.1 shows a schematic of the system developed. The wear sensor consists of a metal contact which is attached to the brake lining support plate. This has an electrical contact to ground through the support plate. Resistance is provided by a semiconductive ceramic bushing. When the brake liner becomes worn, the rotor makes contact with the sensor and a connection to ground is made through the rotor. A circuit is connected to the sensor which notifies the driver through a tri-colour LED.
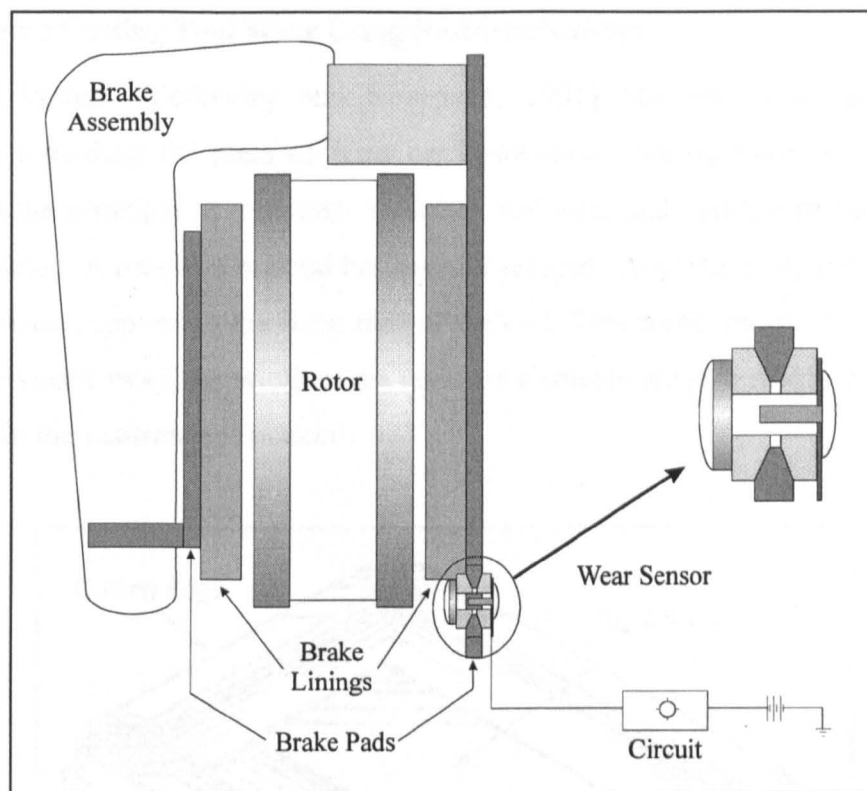
Figure 2.1 - Patented Brake Wear Sensor

Vogt Electronic AG in Germany have developed a sensor for measuring drum brake shoe wear by means of an oscillator coil on a ferrite U-core [Wimmer, 1990]. The wear is determined by a resistance variation on the compensation coil. Kilian et al. (1995) have developed a brake pressure control system that uses a series of brake wear sensors. The output signals from the wear sensors are fed into control electronics, and on light braking the pressure is equally distributed across the braking system. Brakes with a large amount of wear are susceptible to a reduced braking pressure. Similar work [Klein & Fischer, 1995] has used a control system involving wear sensors to equalise out the wear apparent in opposing brake liners.

A smart wear sensor integrated into the brake liners would provide the driver with accurate data regarding the state of wear. This information will also be useful in diagnostics for analysing the quality of the braking system. If, for example, one set of liners is wearing at a faster rate than the others on the automobile, problems of alignment could be diagnosed immediately.

2.2.2 Machine Cutting Tool Wear Using Resistive Sensors

A patented system [Yellowley and Hosepyan, 1991] has been developed in which information regarding the state of wear on throw-away cutting tools is available. The system uses the principle of resistivity to detect tool wear and thus the remaining tool life can be predicted. A previous method has been developed [Aoyama et al, 1987] which uses thin film resistors applied to the flank face of the tool. This works on the theory that as the cutting edge wears away, the width of the resistive elements are also slowly worn away and an increase in the resistance is noticed.
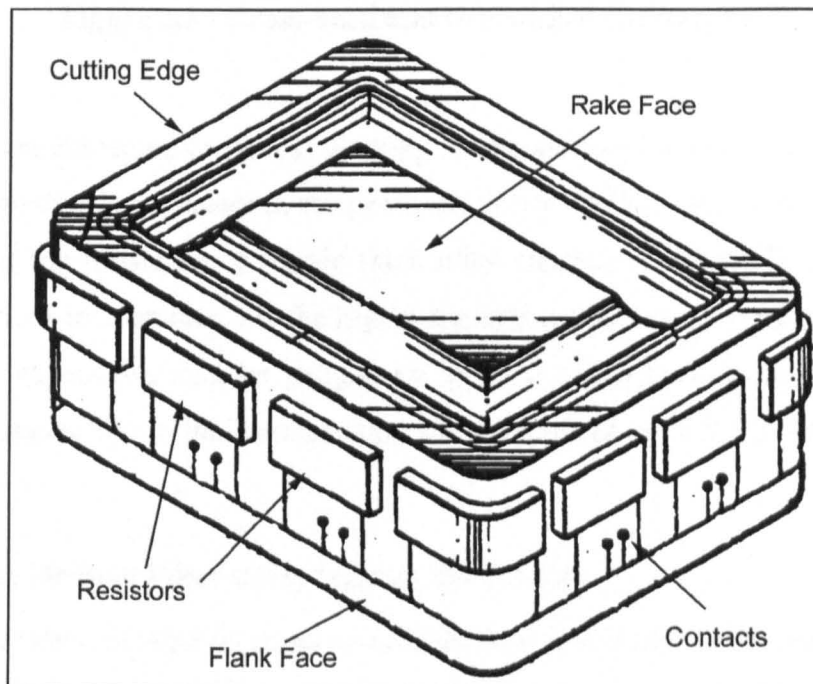
Figure 2.2 - Drawing of Cutting Tool With Resistors Placed Around the Flank Faces

Figure 2.2 and Figure 2.3 show drawings of the patented tool wear detector. Thick film resistors are embedded around the flank faces adjacent to the cutting edge using screen printing techniques. The resistors are 200-400µm wide with a thickness of 10-20µm. These are placed 400-800µm away from the cutting edge. A series of contacts are made available for each of the resistive elements.
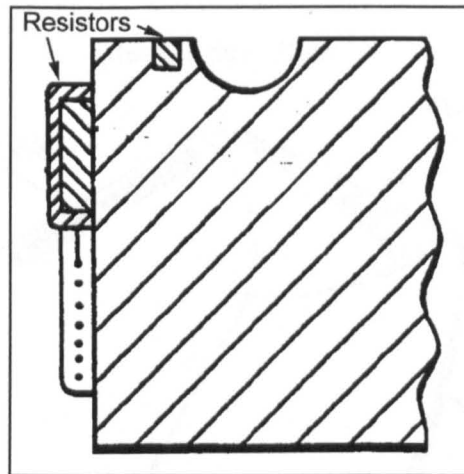
Figure 2.3 - Cross-Sectional Drawing of Cutting Tool

The resistors are subjected to typical working temperatures of between 0 and 500°C. The resistance dramatically increases at temperatures above 300°C. Above this temperature, it was found that the resistance of certain thick films changes permanently. As wear rate is related to the tool temperature, i.e. the higher the tool temperature the faster the wear rate, the tools life expectancy can be gauged by thermal stress techniques. The permanent change in resistance allows the thermal stress to be measured on or off the machine.

2.2.3 Machine Drilling Wear Using Indirect Techniques

Measuring the extent of wear on small twist drills is an area in which direct measurement is extremely difficult.

A wide variety of different type of wear can cause a twist drill to become less reliable. These are detailed in Figure 2.4. A system has been developed [Brinksmeier, 1990] in which a sensor to measure the torque of the rotating shank can provide an early warning system for tool fracture.

A developed eddy current sensor is placed between 0.5mm and 1mm away from the shank on the drill bit. The changes in the magnetic properties of the material due to torsional stress whilst drilling can be measured by the sensor. By using a 10Hz low pass filter, tool fracture can be detected prior to tool failure. This is manifested in an increase in tension, thus an increase in torque on the drill shaft.
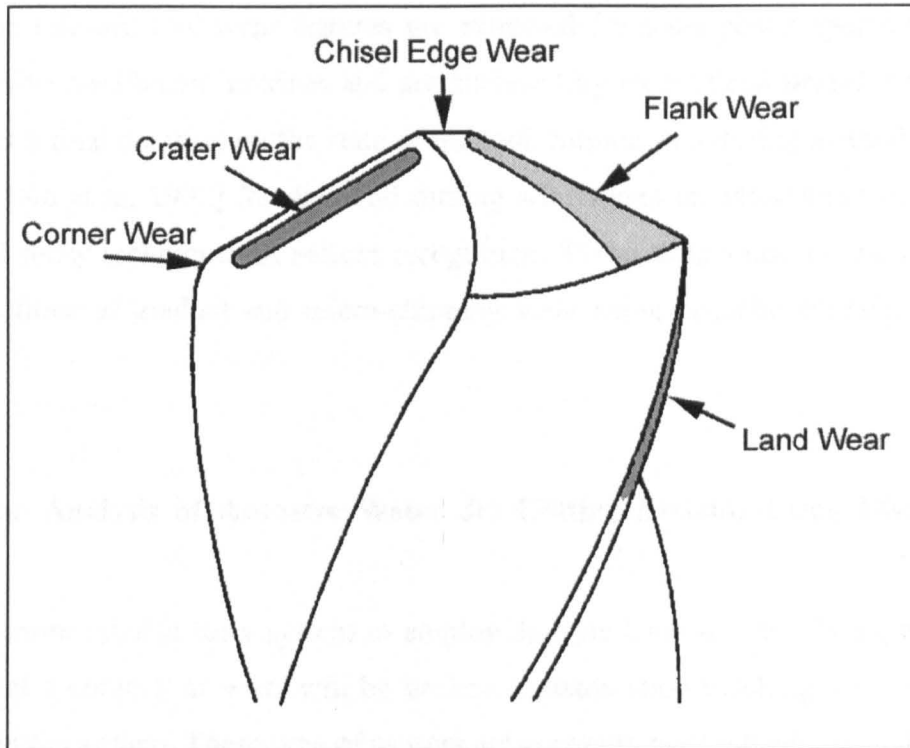
Figure 2.4 - Areas of Wear on a Machine Drill Bit

One of the distinct advantages of this system is that the measurements are taken on-line, without any mechanical contact to the drill. Methods to access any signal from the rotating drill are not necessary, and the sensor is quick and easy to install.

Incipient wear on twist drills used to drill vias in printed circuit boards is determined by using accelerometers [Barker and Hinich, 1994]. Extra indirect sensors are used to measure shaft speed, displacement and vibration. However the use of the Z-axis accelerometer provides the data for statistical monitoring of the drill wear using cumulant spectral analysis. Tool fracture and ultimately tool failure are the main concerns of this system, and due to the accuracy needed for this application, inferior drilling from a worn drill must be detected early.

The use of a suite of sensors to determine the state of a system is called sensor fusion. A third tool wear monitoring example uses sensor fusion techniques coupled with neural networks [Dornfeld, 1990]. The signals from acoustic emission, force and current sensors are processed using fast Fourier Transforms (FFT) or a multichannel autoregressive series

model. The relevant tool wear features are extracted from the power spectral density or autoregressive coefficient matrices and are analysed by an artificial neural network. This then makes a final decision on the state of the tool. Similar monitoring methods have been employed [Ko et al, 1992] for diamond turning which uses an autoregressive time series model and fuzzy techniques in pattern recognition. These techniques are used to monitor small quantities of gradual and micro-chipping wear using acoustic emission, force and vibration.

2.2.4 Wear Analysis of Abrasive Water Jet Cutting Systems Using Discrete Level Sensors

It is often more reliable for a system to employ discrete level sensors. These work on the premise that a contact, or wire, will be broken or made upon reaching a certain point of wear within the system. These type of sensors are generally very reliable, with the accuracy dependent upon the initial production and installation. Special manufacturing methods need to be employed for multiple discrete level sensors with a small spacing between levels.

Machining using abrasive waterjet (AWJ) cutting is an expanding area, and is being used to cut very hard materials such as superalloys, composites, titanium, ceramics, carbon and stainless steel. One of the problems, however, is the wear apparent on the nozzle of the cutter. The tungsten carbide nozzle is effected by abrasive and erosive wear, and the accuracy is decreased as the nozzle wears. A typical life span for the nozzle is 4 hours. It is therefore, of the utmost importance that the computer controlled system is notified of the extent of the nozzle wear. A variety of methods have been investigated including optical tracking, ultrasonics, vibration monitoring and noise monitoring. However, a low cost and highly accurate wear probe has been developed [Kovacevic and Evizi, 1990]. A single conductive loop is embedded in the tip of the nozzle. When the nozzle wears to a pre-determined maximum internal diameter, the conductive loop breaks and a pneumatic valve shuts off the system. With this method proving to be the most effective, further developments of this sensor [Kovacevic, 1991] have been introduced in which multiple levels of conductive loops are integrated into the nozzle tip. These are accurately spaced at 0.05mm apart, and provide the user with a more informative system.

2.2.5 Ultrasonic Thickness Measurement

Ultrasonic non destructive testing (NDT) has been a technology which has endured a high degree of research. Typical applications have included the analysis of the multilayered structure of concrete [Popovics & Rose, 1994]. By analysing the thickness and bulk properties of the concrete layers, the deterioration of building structures can be predicted. Flaws and cracks in pipes within the electric-power industry are now being detected using ultrasonic pulse-echo techniques [Case, 1996].

However, the use of ultrasonic techniques are now being used entirely to measure the thickness of a material. Ultrasonic thickness gauges are available which can measure up to 2.5 metres thickness of most materials at an accuracy of up to ±0.01mm for some applications [Fowler et al, 1996]. The device is based on a piezo electric crystal that converts pulses of electrical energy in to ultrasonic sound waves. The thickness of material is calculated by the time taken for the sound wave to travel through the material and reflect back off the opposing surface. i.e.

$$d = \frac{Vt}{2}$$ **Eq 2.1**

where

d = thickness of the test piece

V = velocity of sound wave in the material

t = measured time from transmit to receive

Frequencies of operation range from 500kHz to 100MHz, with the lower frequencies used for thick material and the higher frequencies, thin material. Typically, hand held devices have been developed which are used for one off measurements. An application of a liquid couplant such as glycerin or propylene glycol is used to ensure the maximum transference of sound energy.

The use of ultrasonic thickness gauges are being used extensively in the shipping industry. Surveys to determine the thickness of tankers and bulk carriers are regularly carried out [Porter, 1997] to meet international standards. New classification methods have been adopted using ultrasonic thickness gauges which are a significant improvement on the previous method of drilling holes in the tanker.

Research has been carried out [DiGuinta et al, 1996] to determine the thickness of underwater off shore structures. However, problems of encrusting layers of up to 5mm covering the steel walls can provide erroneous measurements. The feasibility study uses simulated encrustation, and it is believed that using ad hoc procedures for backwall echo identification, the thickness of the wall can be measured without removing the calcareous layers.

The use of ultrasonic thickness gauges for continuous on-line measurements is possible, however, the accuracy of thickness measurement could be compromised. A major manufacturer of ultrasonic thickness gauges stated that '*vibration is a problem when taking ultrasonic measurements*', and the measurement of the manganese wear liners whilst the cone crusher is running is '*probably going to be difficult to achieve with our equipment*' [Smith, 1997]. Problems of inaccuracies are also caused by cable length in excess of 1 metre between the transducer and the gauge. Reflections from the cable can cause spurious signals and distortion of the results. The thickness of couplant is critical which must remain consistent throughout the monitoring of the measured material. Furthermore, the pressure in which the transducer is applied to the material must be kept constant and any tilting of the transducer will cause inaccurate readings. The accuracy of measurement can also be degraded by the structure of the material under measurement. Coarse grained metal castings and composite materials produce multiple sound echoes, and clean signals from the backwall may not be possible.

2.3 Smart Sensors

Smart sensors have endured a phenomenal growth since their initial inception over ten years ago. Growth rate figures of 22% per year are typical, with expected sales to reach the order of $1.1 billion by 1997 [MIRC, 1991]. The extent to which a sensor is smart is under some discussion, however, the main structure of a smart sensor is globally agreed. A smart sensor is defined as "..a system that contains the functions of sensing, signal conditioning, A/D [Analogue to Digital] conversion and a bus output. It may also have on-chip calibration and self test" [Huijsing et al, 1994]. Other definitions have taken the idea of smart sensing one step further. "A smart sensor should include all required analogue signal-processing functions - amplification, impedance transformation, level translation, filtering and multiplexing. It should also digitise analogue transducer outputs, have an

external bi-directional bus for communications, contain a specific address for user access and identification, and be able to execute commands or logical functions received over a digital bus. Other desirable features include compensation of transducer outputs, self-calibration, autoranging, programmability, self-testing or self-diagnosis, fault tolerance, and failure prediction" [Leonard, 1993]. This is a fairly comprehensive list of attributes. However, a further dimension to the scope of the smart sensor can be added in the form of remote control and data retrieval by way of a telemetry link. Within each of the various modules of the smart sensor, there is a high volume of different solutions to achieve a competitive product. This not only applies to the physical construction of the electronic components, but also the embedded algorithms [Zhang et al, 1994] to extrapolate the results from the sensor data.

2.3.1 Smart Sensor Fabrication Strategies

There are several methods of converting a sensor into a smart sensor [Heintz and Zabler, 1989]. If the components of the smart sensor are to be part of a single piece of silicon, either a custom processor is specifically developed, or the smart sensor is developed by processor manufacturers. The utilisation of present microcontrollers for smart sensor development offers the advantage of a smaller time to production as the central processing unit is already established. A separate microcontroller offers greater flexibility, as well as the elimination of undesirable effects such as electromagnetic interference and self heating. One of the problems, however, is the link between the sensor modules and the processing unit can cause a degradation in performance, as well as the likelihood of a physically larger smart sensor being developed.

Figure 2.5 details some of the different approaches adopted when developing smart sensors.

Smart Sensor #1 shows a system that incorporates the sensor(s) and signal conditioning onto a single chip, the shaded area showing the single chip development. Generally, a micromachined component with a mechanical attribute such as pressure or displacement is formed in silicon along with the necessary signal conditioning.
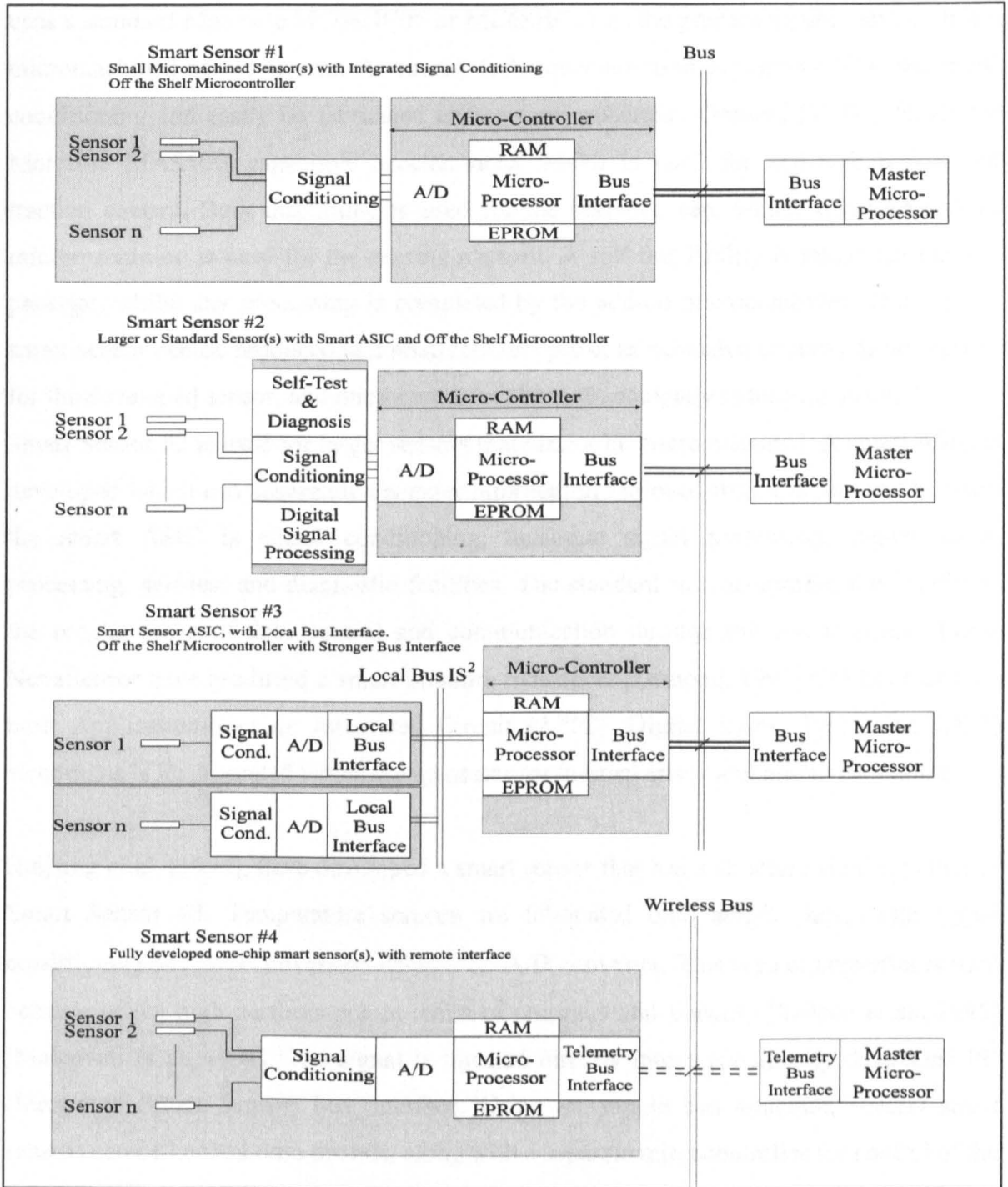
Figure 2.5 - Smart Sensor Fabrication Options

In the case of smart sensor #1, the signal is utilised by a standard microcontroller, and a hybrid circuit produced. A typical example of this is a micromachined capacitive pressure sensor [Hughes, 1995] which is produced using the sacrificial oxide process. Also, a smart capacitive sensor has been developed for a vehicle air-bag system [Ormond, 1993] which

uses a standard Motorola MC68HC05 or MC68HC11 as the processing unit, and a surface micromachined accelerometer. As similar techniques are used to produce IC's, the signal conditioning can easily be fabricated on the same substrate. Ormond [1993], details the Motorola MAS50G capacitive accelerometer which is used for anti-lock brakes and traction control. Bulk machining is used for the hermetic cap, whilst advanced surface micromachining is used for the sensing element. A self test facility is integrated into the package, whilst any processing is completed by the add-on microcontroller. This type of smart sensor can be produced at a relatively low price, as extensive circuitry is not needed for the developed sensor, and microcontrollers have dramatically reduced in price.

Smart Sensor #2 is used for larger sensors that cannot be micromachined. A smart ASIC is developed which can accurately decipher information retrieved from the sensor. On board the smart ASIC is signal conditioning, analogue signal processing, digital signal processing, self-test and diagnostic facilities. The standard microcontroller then performs the regular tasks of data control and communication through the bus interface. Lucas NovaSensor have produced a smart pressure transducer [Ormond, 1993] that has a custom built Application-Specific Integrated Circuit (ASIC). Digital Signal Processing (DSP) electronics is incorporated which compensates for thermal errors and output calibration.

Huijsing et al. [1994], have developed a smart sensor that has a structure similar to that of Smart Sensor #3. Temperature sensors are fabricated onto single chips, with signal conditioning and A/D conversion using a $\Sigma\Delta$ A/D converter. This type of converter is used because of the high performance in terms of accuracy and linearity [Wilson et al., 1995] [Malcovati et al, 1994]. The signal is injected directly into a specifically developed IS^2 (Integrated Smart Sensor) bus interface. Using a simple bus structure, several smart sensors can be hooked onto the bus, along with a separate microcontroller for control of the sensor(s). The stronger well-developed bus interface on the microcontroller can then be used to interface with the outside world. A similar architecture has been adopted for the development of a flow and temperature measurement system that employs the I^{2}C bus as the internal bus [Hogenbirk et al., 1995].

Smart sensor #4 is the main aim for sensor developers, with the complete unit integrated onto a single chip. [Wilson et al., 1995] describes how a Universal Sensor Interface Chip

(USIC) is under development where a highly flexible sensor interfacing block supports a wide range of sensors. The system uses an 8-bit Reduced Instruction Set Computer (RISC) as the core of the USIC, along with 16 bit $\Sigma\Delta$ A/D converters, amplification, 24 bit counters and serial and parallel communications. Specific signal conditioning is provided on the USIC in the form of a synchronous demodulator that provides the user with a direct interface for capacitive measurement. Capacitances of up to 900pF can be measured at a resolution of 0.01pF.

This is still commercially unrecognised due to a variety of different reasons, the main concern being the final cost of the smart sensor. Other concerns are voiced [Olbrich et al., 1994] with regard to the modularity of single chip solutions, and the inherent development cost of specific single chip smart sensors. It is still the general consensus of opinion that multichip solutions on small printed circuit boards offer the most flexibility. By using methods such as surface mount and flip chip technology [Burkhart, 1991], the amount of printed circuit board used is kept to a minimum.

2.4 Serial Bus/Distributed Control Systems

Distributed control is used as a term for an integration of several modules that can work within an entire system, but can also retain enough intelligence to operate as a stand alone module. Distributed systems are connected via a two wire serial bus, and data and control commands can travel to and from any of the nodes on the distributed control system.

One of the great advantages of utilising a distributed system is the ability to set up the initial structure of the system, but allow for expansion in any direction. It is readily known that the secret to a good design, is in the modularity of the overall system. This is not only true for the structure of the hardware, but also in designing the software. It is imperative that individual modules can slot into the system with ease and be easily removed without disruption. Standard systems that are used today, particularly in the quarrying industry, are based around a single Programmable Logic Controller (PLC) with all inputs and outputs extending from this device to the controlled machine. Problems occur, however, when increased functionality of the machine is necessary in the form of additional sensors or actuators. This increase in peripheral count can often exceed the capabilities of the PLC,

and a complete program re-write is necessary on a PLC with increased capacity. Furthermore, the need for wiring between each sensor/actuator and the PLC can dramatically increase costs and system downtimes.

A typical example in which a standard system is converted to distributed control is the NODDATOR project [Uusijärvi & Törngren, 1994]. The introduction of distribution for hydraulics and computer control for a mobile tunnelling and forest harvester is described. With the development of the system, the amount of hydraulic hoses and electrical wiring is significantly reduced and thus reducing the system cost. Furthermore, an increase in functionality and reliability of the system has been noticed.

To aid the development of distributed control systems, smart sensors with integral serial bus connectivity (Section 2.3) are providing system developers with instantaneous compatible sensors. These are rapidly becoming commonplace in systems that we use. There is already considerable utilisation of this in the automotive industry, with devices such as Engine Management Systems [Chowanietz, 1995], anti-lock brake systems (ABS), air bags [Edwards, 1993], security and global positioning linked together over a single or multiple bus structure. Intelligent bus systems with a network of smart sensors allows information from any device to be stored in memory for future reference. Thus, the implementation of a 'black box' will enable more information to become available in the event of a road accident [Goldman et al, 1990]. The development of such a system inside the home is now gathering momentum [Huijsing et al., 1994], with sensors such as security, lighting, heating and smoke detection linked together over a single serial bus to produce a more controlled environment. With the integration of bus systems into smaller and smaller IC's, the cost of distributed smart sensors is providing the industry with a more economical solution [Middelhoek, 1994].

Specifying a particular bus structure for a system is not an easy task, with the multitude of bus systems around. One of the main design criteria is the reliability of the system.

Fieldbus	Country of Origin	Installed Nodes 1995	Comments
CAN	Germany US	4 000 000	Originally developed for automotive applications. Widely used in industry, principally as a sensor/actuator bus
SDS	US	5 000	Honeywell developed sensor/actuator bus based on CAN
DeviceNet	US	25 000 (1996)	Originally developed by Allen-Bradley to link sensors and actuators to PLC's using CAN.
ASI	US	10 000	Sensor/actuator bus used by Siemens and others to link low level devices into Profibus networks.
Interbus-S	Germany	750 000	German DIN standard. Sensor/actuator bus
P-NET	Denmark	45 000	Danish national standard incorporated into EN50170
LonWorks	US	1 500 000 (1996)	De facto standard developed by Echelon. Widely used in building automation.
HART	US	600 000	Interim solution providing digital communication with 'smart' transmitters over conventional 4-20mA circuits.
FIP WorldFIP Europe	France	30 000	French national standard, incorporated into EN50170. Linked via WorldFIP to Fieldbus Foundation.
Fieldbus Foundation	US		Merger of breakaway groups from original ISA/IEC standardisation process. Includes WorldFIP North America, Interoperable Systems Project (ISP). Has links with WorldFIP Europe and Profibus.
Profibus	Germany	650 000 (1996)	German (DIN) standard, incorporated into EN50170. Three versions: FMS for higher level comms; DP for device level comms; PA for process control with Intrinsic Safety.
IEC/SP50	US		The original and still to be finalised unified fieldbus standard-IEC1158. Many of the above have undertaken to migrate to the standard once it is complete.
EN50170	CENELEC		European interim standard - adopts Profibus, P-NET and FIP as parallel standards.

Table 2.1 - List of Fieldbus Technologies [FieldComms, 1995]

However, by analysing some of the systems more variable requirements such as speed of data retrieval, distance between nodes and cost, a particular bus structure can be chosen. Over the past years, a great deal of effort has been exerted in the form of development of

new serial systems, such as LONworks [Echelon, 1990][Jones, 1992][Black, 1994], I^2Cbus [Hogenbirk et al, 1995][Phillips, 1990], CANbus [Zabler et al, 1992], BITbus [Quin Systems, 1991][Shiffbauer, 1991], HART and Fieldbus [Reeve, 1993][Wilson et al, 1995]. The integration of the stronger buses such as Fieldbus and CANbus into a single chip smart sensor is still not fully realised for low cost solutions. Hybrid solutions, however, are under development [Huijsing et al., 1994]. Table 2.1 shows some of the current bus systems available with their countries of origin and the popularity of the bus.

2.5 Telemetry

In today's world of high precision machinery, it is becoming more and more critical that physical parameters on moving or rotating machine parts are monitored. By using radio telemetry systems, vital information can be transmitted reliably without any physical connections. The general definition of a telemetry system, as given by Strock [1983], is "any grouping of data measurements in a format that can be transmitted or stored on a single medium, received or retrieved from that medium, and separated into the original measurement components for observation". This definition, however, allows for an inclusion of a large amount of transmission medium. As the term 'telemetry' has been used over the years, it has become more of an unwritten law, that data transmitted via telemetry uses a wireless medium.

Two way control systems can not only utilise telemetry for data retrieval from a remote source, but can also use telecommand or telecontrol [Gómez et al, 1991] to initiate sequences, modify parameters or terminate systems within the remote source. Due to the increasing rate in sensor development, more focused data is becoming available. Therefore, it is imperative, that the telemetry system can interface with the new technology of sensors and the transmitting system can be inserted discretely into a constricted environment without impairing the performance of the system. Current telemetry systems in use include measurement of temperature [Chutatape, 1989], torque [Brown, 1990], force [Testem, 1992], pressure [Irish et al, 1990], acceleration and vibration on systems such as turbines, automobile shafts, pistons & connecting rods, rotors and many other moving and rotating machines.

An application [Testem, 1992] that shows the real flexibility of using telemetry modules within engineering, is the analysis of a turbine rotor disk, and the effect of the turbine blades as it rotates. Sixty strain gauges are wired to the centre of the shaft and record the deflection due to the vibration of the turbine blades. More importantly is the fact that the electronics can cope with the working temperature of 85°C and a turbine speed of 25800 revs/minute. A custom built inductive coil is used to power the transmitting unit with data transmitted via five scanner transmitters working in parallel.

As well as developments of telemetry systems based around plants, factories and general use, systems are now being developed for use in the medical profession. Even though these systems can make use of any of the exempt telemetry bands, the Radiocommunications Agency has allocated licensed bands, class 1,2 and 3, specifically for use in this profession. Class 1 devices operate within 300kHz and 30MHz and is used for ingestible radio pills. Class 2 and 3 operate between 173.7 and 174MHz and are used for narrow band or wide band applications.

A system developed in America with a custom built microprocessor [Fernald et al., 1989][Cook et al., 1990] combines data acquisition, actuator control and two way telemetry. The complete system is linked via a serial bus that allows for interchip communication within peripherals, and is hermetically sealed into a small implant capsule. The telemetry module is bi-directional, and can not only transmit data acquired from the transducers, but can also receive telecommands that can alter sample times, parameters, or even load a new program. The system allows transmissions on a 40MHz carrier of either one word commands, or 64 word data packets, both of which are controlled by the microprocessor. An impressive feature of this system is the ability to re-boot the microcontroller remotely via the telemetry link either prior to, or after the system implantation.

One of the emerging applications for the use of telemetry, is the integration of radio telemetry into Local Area Networks (LANs). The development of Fieldbus and similar technologies has enhanced the ability to utilise distributed systems from indoor factory automation to huge compounds spanning several kilometres. Factory integrated systems may use Pacsnet 3000, telemetry on a Fieldbus, Modbus or proprietary network that supports MPT1328/1329/1411 and runs on a dedicated channel, widely reducing the amount of wiring needed. An extensive

system can support up to 256 nodes over 20km, with a response time of 2 seconds [Knights, 1993].

2.6 Summary

Condition monitoring is implemented into a variety of different systems, with a differing degree of complexity. The quality of the condition monitoring system, however, is only as good as the sensor inputs. One of the main areas in which condition monitoring systems have been designed is that of fault diagnosis and fault prevention. This provides the user with instantaneous savings in the form of predicting maintenance downtimes and preventing unexpected failures. There are further benefits of condition monitoring in that of system optimisation. Using historical data and trending techniques, the optimum set-up of a system can increase performance of the system in terms of increased production rates, improved energy consumption and an enhancement of quality.

A series of different Knowledge Based Systems have been described, each using a different development shell. When developing a KBS, it is important to identify the factors which influence the choice of the KBS shell. Real-time operation is often a major factor for on-line control systems, although not so important for consultative KBS. The Knowledge Base language should be easy to learn, as low development times are often required. An increased flexibility is provided with the links to conventional software. These are often in the form of databases, high level languages such as 'C' and spreadsheets. Algorithm development is often provided with provisions for fuzzy logic and neural network development. Finally, the man-machine interface should be clear with graphical displays to mimic the functionality of the system being monitored.

A wide variety of wear measurement techniques have been analysed, in a variety of different industries. It is apparent that wear monitoring of crucial components within the developed systems is vital for accuracy and reliability. Of the examples described, methods to investigate wear directly using sensors positioned in the area of critical wear are more desirable. Problems of reliability of the sensors are apparent as they are often subjected to severe working conditions.

Indirect wear measurement is often more realistic, with the use of readily available sensors in alternate areas of the system providing data for wear analysis. Modelling techniques are used extensively, with case studies examined for the different sensor readings. A typical example of this is described [Dauw, 1986] in which the wear inherent on Electro-Discharge Machining (EDM) tools can be calculated by examining the pulse parameters used in operation. One of the main problems of this, however, is that the measurements are not solely influenced by tool wear, but also from the various process parameters surrounding the tool. These include variations on the work material, cutting conditions and possible changes in the geometry of the tool.

Due to the nature of these industries, a pre-requisite of any wear system developed is that the monitoring system must be able to provide wear data on-line. Systems that can only measure wear off-line cause delays in production, and external equipment is often required to produce a wear measurement. The sensors developed in this thesis provide direct low cost on-line measurement in real time.

Upon analysing the market for smart sensors, it has become apparent that this is a high growth area. The conversion from sensor to smart sensor increases the flexibility of the sensor unit in several ways. Sensor calibration and signal processing increase the functionality of the sensor, whilst the addition of a serial bus allows a new smart sensor to be added to the data acquisition system with a reduced amount of wiring and easier installation. A series of smart sensor fabrication techniques have been described which vary from an off-the-shelf microcontroller with hybrid sensor and signal conditioning, to fully integrated smart sensor with all components fabricated on the same chip. The decision as to the complexity of the smart sensor fabrication is usually with the application the sensor is designed for, and the number of units to be commercially produced.

It is important that these smart sensors can easily link to a distributed control and data acquisition system via a serial bus. Distributed systems offer a greater flexibility and ease of expansion over traditional control systems, such as PLC's. A wide variety of serial bus systems are now commercially available, with the constraints of the application often deciding the type of bus structure employed (i.e. speed, distance between nodes, maximum number of nodes etc.). Serial buses are also available at a lower level of complexity and are

provided on chip with some microcontrollers. The NEC 78K3 series microcontrollers contain an on-chip SBIbus which links NEC microcontrollers over a two-wire bus.

Condition monitoring systems often have to employ a telemetry link to enable signals to be sent and received from the sensors or control units. The implementation of telemetry into distributed systems provides local wireless nodes which can receive data from moving elements, and control inaccessible actuators.

CHAPTER 3
THE QUARRYING INDUSTRY

3. THE QUARRYING INDUSTRY

3.1 Industry Overview

In 1991, 246 Million Tonnes of aggregate was produced in the UK alone, which involved an energy consumption of some 6.25 TWh/yr [Bearman and Parkin, 1992a], using a significant proportion of all electricity generated in the UK. This figure is expected to rise to nearly 400MTonnes per year by the year 2011 [Bearman and Parkin, 1992b]. Figure 3.1 and Figure 3.2 show the distribution of European and World aggregate production. Over half of the aggregate produced in the world is concentrated throughout Europe with the greatest proportion of European rock crushed in Germany.

European Aggregate Production
Annual Figures for 1991

Finland 58   Spain 259

Italy 280   Sweden 100
Austria 92
Belgium 39

Switzerland 55

U.K. 246

Germany 700

France 380

Denmark 39

Figures in Millions of Tonnes

Source: BACMI, European Aggregates Association, TSGB
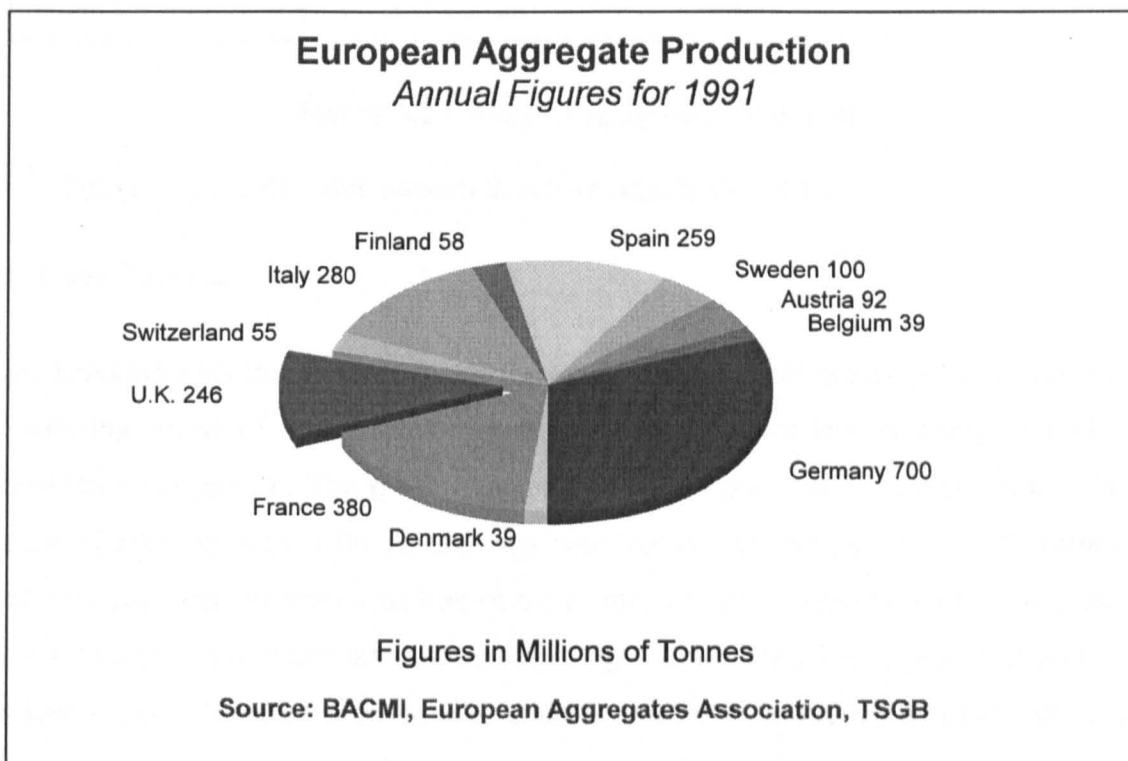
Figure 3.1 - European Aggregate Production

*Fig3.1 Not including Eastern European figures.

World Aggregate Production
Annual Figures for 1991

United States 1470

Europe 3100

Africa 170
South America 200

Asia & Australasia 600

Figures in Millions of Tonnes

Source: BACMI, European Aggregates Association, TSGB
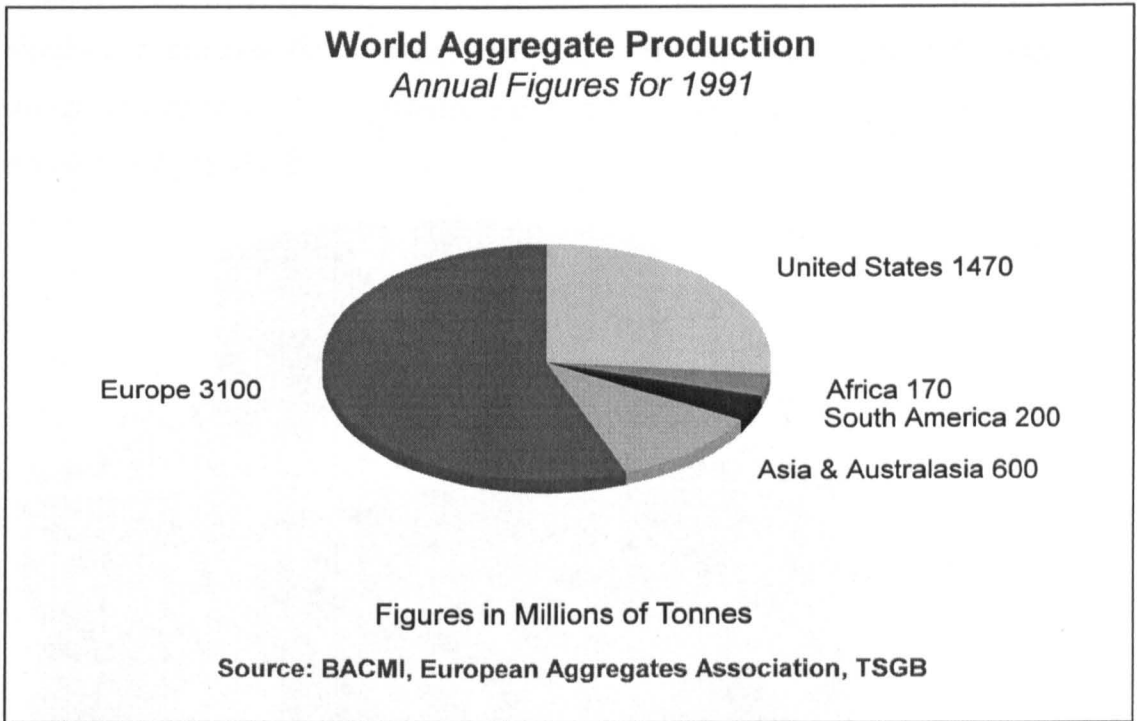
Figure 3.2 - World Aggregate Production

*Fig 3.2 Europe figures includes Eastern European aggregate production.

3.2 Cone Crushers

Cone crushers are manufactured by all the leading mining and quarry plant manufacturers to differing levels of sophistication, and are used for secondary crushing of rock into roadstone or aggregate. The profit generated from the production of aggregate is in the region of £450 million, with the average cone crusher producing up to 100 Tonnes of roadstone per hour. Primary crushing of the as blasted rock is carried out by jaw crushers. Cones then form the secondary and tertiary stages of the mill. Large pieces of rock enter the feed hopper above the cone crusher, and are crushed by the rotary and gyratory motion of a conical wear liner attached to a central main shaft. This rotates in an eccentric motion at around 6 revolutions per minute loaded. The rock is crushed between the rotating wear liner, or mantle, and the stationary outer liner, or concave. Both of these are one piece castings made from high grade wear resistant manganese steel, and typical sizes for U.K. cone crushers measure up to 3.05 metres in diameter. The desired output diameter is set by the discharge setting, or nip angle, and is usually hydraulically controlled by double acting hydraulic cylinders mounted externally to the crusher. In the case of an incorrect feed such

as tramp iron, an increase in pressure will generate a signal to instruct the hydraulic cylinders to separate the crushing members. This will then allow tramp iron[3] to drop through the crusher. More detailed information about the operation of the crusher is provided in Appendix E.
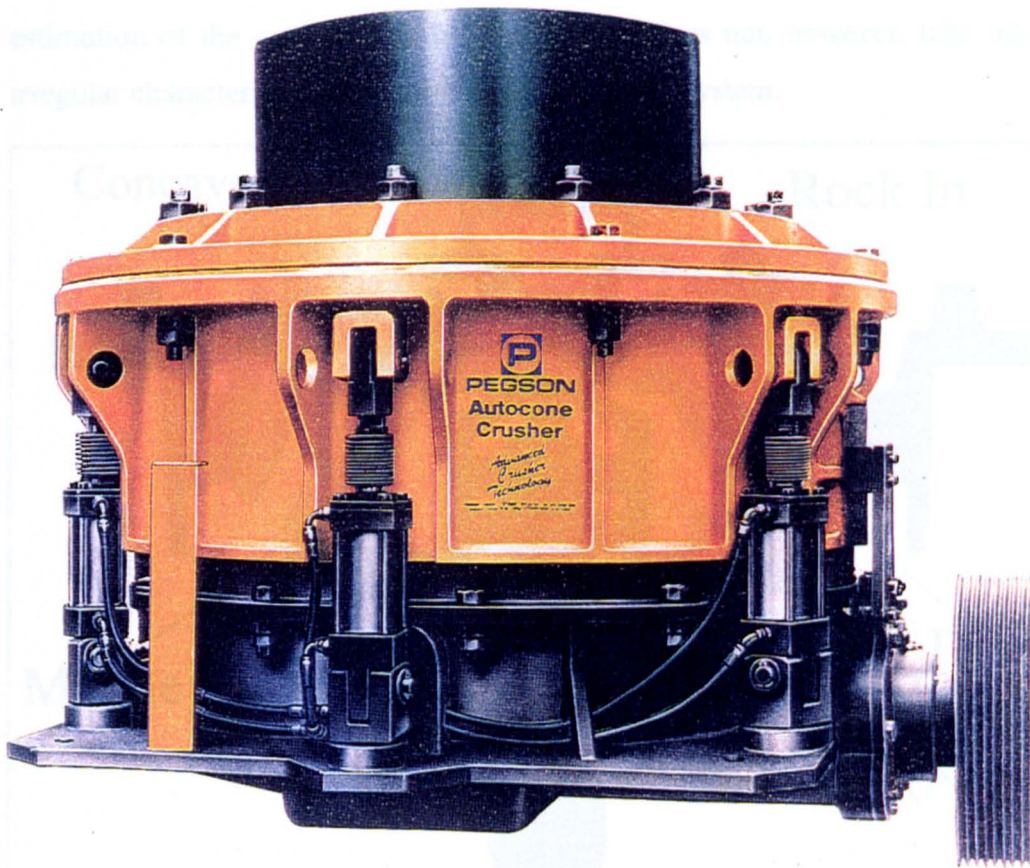
Plate 3.1 - Pegson Autocone

The Pegson Autocone (Plate 3.1), which the project members used for experimentation purposes, uses hydraulic means for setting of the nip angle and overload protection. However, early cone crushers have used systems with adjustments via screw threads and spring loaded overload protection [Moshgbar et al, 1995a]. Due to the abrasive action of the rock, the crushing members can wear at a rate of up to 1mm per hour [Bearman 1991], with an average life expectancy of two weeks. With the expense of the wear liners, therefore, it is imperative that the mill manager can utilise the majority of the manganese

---

[3] a term used to describe unexpected hard material fed into the crusher

before replacement. It is also a necessity that the discharge setting remains constant to deliver aggregate with a similar diameter, including compensation for wear on each of the manganese liners. The system currently employed for the determination of wear only considers the expected wear rate and the amount of hours crushed. From these figures, an estimation of the wear pattern is formed. This does not, however, take into account the irregular characteristics of an eccentrically rotating system.
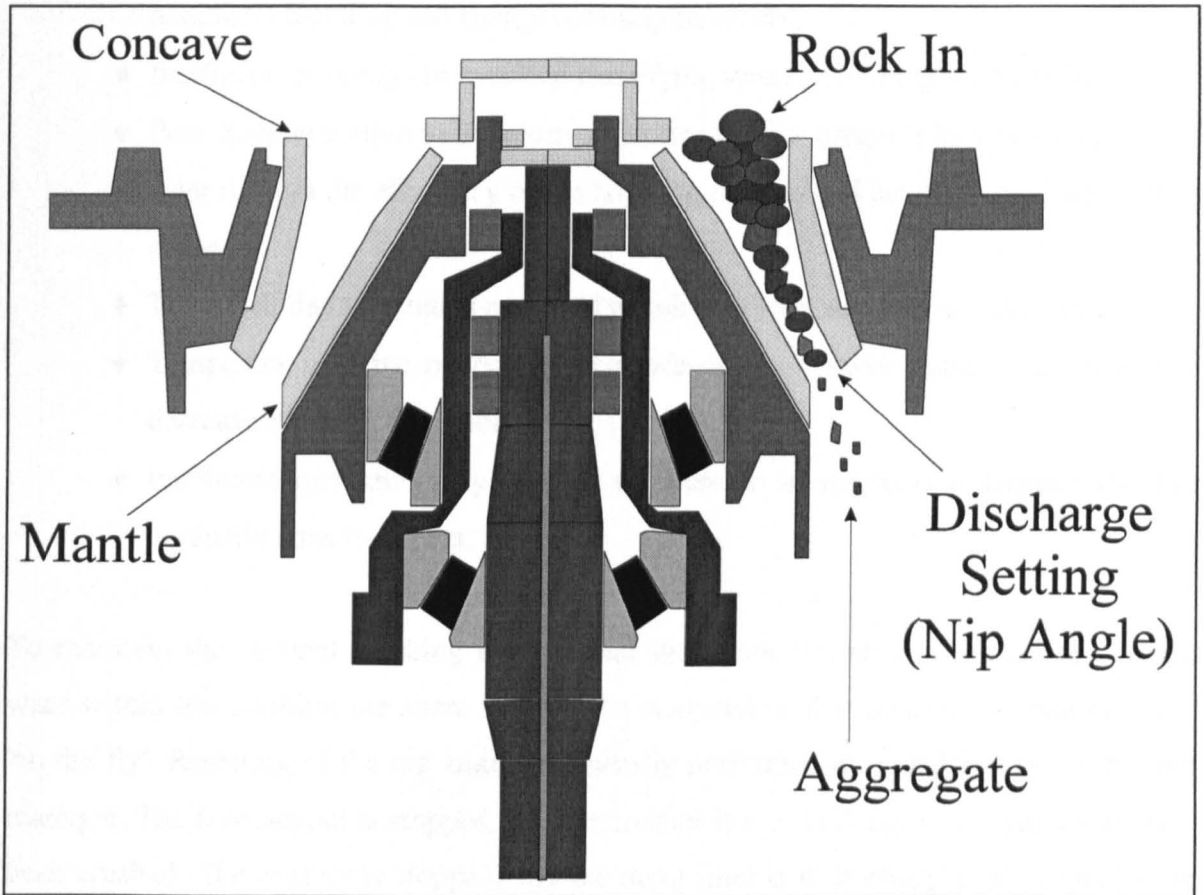
Figure 3.3 - Cone Crusher Schematic

Figure 3.3 shows a schematic of a cone crusher, detailing the stationary concave liner, and the rotating mantle liner which rotates to grind the rock into aggregate.

3.2.1 Problems Associated With Irregular Wear

One of the most important parameters regarded by the quarrying industry, is the production of uniform aggregate. The discharge setting (or nip angle) of the cone crusher specifies the aggregate diameter produced. The production of a consistent aggregate diameter reduces

the need for re-circulating rock into the cone crusher, thus keeping production costs to a minimum.

If cone crushers are not operating efficiently the following effects are noted:

- ♦ Product does not reach size specification and is either used for inferior (lower value products) or is blended back into the crushing cycle, thus incurring additional handling and energy consumption costs
- ♦ Inefficient crushing can overload classifying screens reducing their efficiency
- ♦ Poor feed regulation to the crushers increases wear costs drastically and increased wear reduces the efficiency of the crushing process and produces particles of poor shape
- ♦ Too much fine material is produced which may represent non-saleable material
- ♦ Temporary machine overloads occur which can become cyclic, thus causing a decrease in production rates
- ♦ Inefficient operation may lead to an increase in mechanical failures and lost production due to repairs.

To maintain the optimal crushing regime, real time data regarding the state of inherent wear within the crushing members needs to be analysed so that adjustments can be made 'on the fly'. Resetting of the nip angle is generally performed on a daily basis by the mill manager. The feed hopper is stopped, and the crusher is run until any remaining rocks have been crushed. The crusher is stopped, and the outer liner is then brought into contact with the inner liner. Contact is recognised by an increase in pressure on the hydraulic rams. The outer liner is then hydraulically withdrawn and the nip angle set. The reading is supposedly accurate to within 1mm and is determined by a Linear Voltage Displacement Transducer (LVDT) attached between the upper and lower frame. However, there are several possibilities of inaccuracies inherent in this system. The mill manager is uncertain of an empty crusher, and stray rocks or tramp iron could increase the hydraulic pressure without the liners touching.

3.2.2 Wear Profiles

Tests on the wear liners [Moshgbar et al., 1995a] show that the wear on the manganese liners is non-linear, with certain areas more prone to wear than others. Figure 1.4 shows a

schematic of a cone crusher detailing two types of wear profiles, parallel and bell shaped. In both cases the majority of the wear is towards the main crushing zone or nip angle.
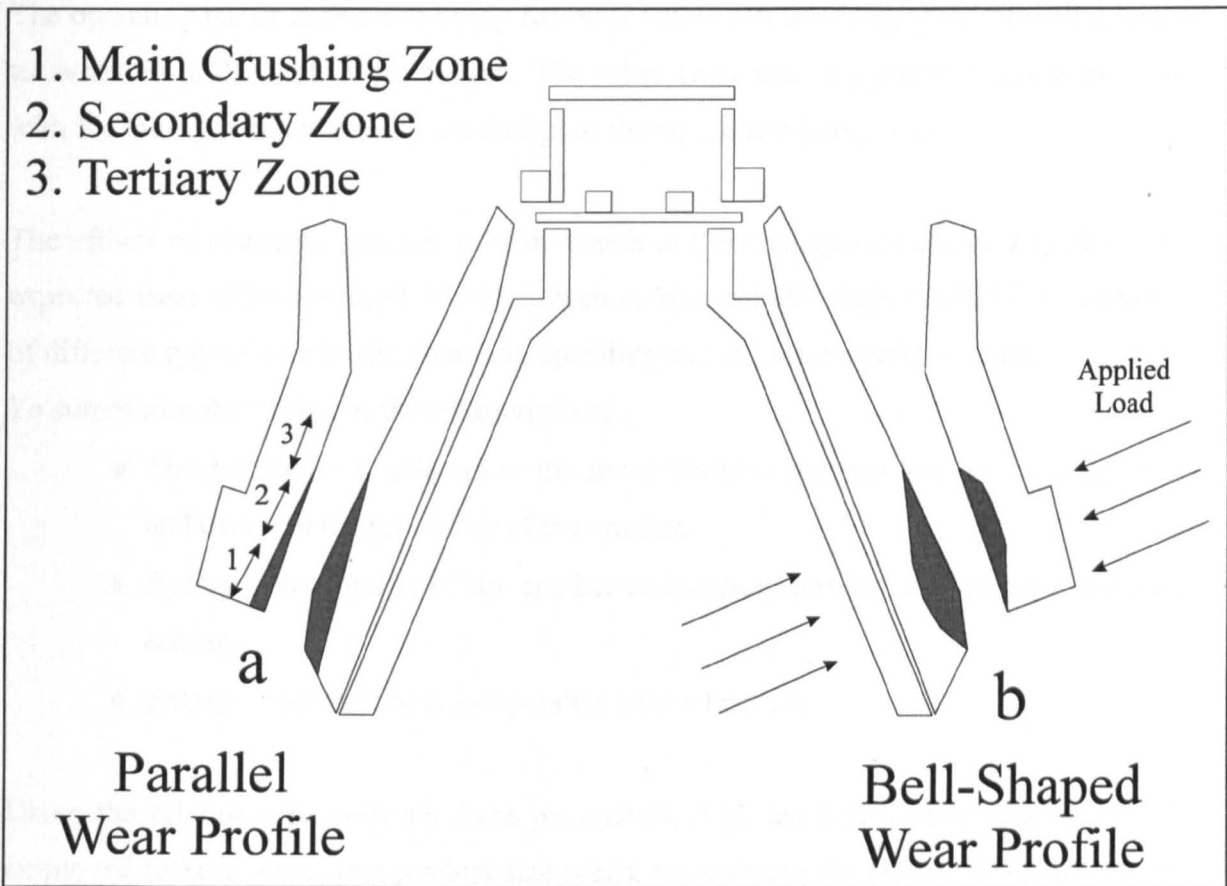
Figure 3.4 - Cone Crusher Wear Profiles

In the case of a badly worn set of liners, an undesirable gap within the main crushing zone will appear when the liners are bought into contact. This contact will be in the secondary or tertiary crushing zone. An increase in hydraulic pressure will follow which signals the liners touching, and an accurate calibrated discharge setting will not be achieved.

The wear patterns that emerge from the used liners are dependent upon a wide variety of factors. These are split into 5 categories:-

1. Machine Design Variables
2. Rock Properties (physical, chemical and mechanical)
3. Operating Parameters (crusher setting, feed size)

4. Environmental Parameters (moisture content)

5. Liner Properties (hardness, micro-structure)

The operating parameters are the only category which can be changed by the mill manager to avoid an undesirable wear pattern. The other categories are either inherent problems with the quarry being mined, or the design of the equipment being used.

The effects of changing parameters within each of these categories allows a model of the expected wear to be produced. This has been carried out [Moshgbar, 1995a] on a number of different types of rock with changing operating and environmental parameters.

To summarise the results of these experiments:-

♦ The presence of moisture in the rocks tends to increase the wear on the liners and decrease the efficiency of the crusher.

♦ Energy consumption of the crusher is inversely proportional to the closed-side setting.

♦ Energy consumption is independent of the feed size.

Using the relationships between these parameters, a closed loop control strategy can be employed to keep a constant product size whilst maintaining the efficiency of the crusher. This is to be implemented using an adaptive controller (inputs taken from the crusher settings), a wear model (parameters of the rock crushed) and a wear measurement system (smart sacrificial wear sensors). This is explained in more detail in Chapter 6.

CHAPTER 4
WEAR SENSOR DESIGNS

4. WEAR SENSOR DESIGNS

The development of the Novel Sacrificial Wear Sensor is initially for the use within the Quarrying industry, most notably within the Cone Crusher liners. For this purpose, the physical designs were designed specifically for this application. However, minor modifications could allow the sensor to be introduced into some of the industries mentioned throughout Chapter 2. A series of prototype sacrificial wear sensors have been developed which use capacitive, resistive and conductive principles. The sensors are inserted into several key areas of the manganese liners, and a signal from the sensor is provided which is related to the length of the sensor. The sensors wear away at a rate equivalent to that of the manganese liners, thus giving an accurate representation of the liner wear. Due to the environment of the quarrying industry, the sensors are housed in steel inserts of similar hardness to the surrounding material.

4.1 Design Requirements

In order to provide the quarrying industry with a system capable of detailing the wear on the cone crusher liners, several aspects needed to be addressed. A table of design requirements is shown in Table 4.1.

FACTORS	CONSTRAINT
Initial Sensor Length	52mm outer liner (concave) 53mm inner liner (mantle)
Sensor Width	No constraint. - If the sensor is excessively wide, multiple sensors could weaken the manganese liners
Accuracy	Sub 1mm. Present accuracy is ±1mm
Expected Lifespan	Varies dependent upon quarry, rock crushed and crushing strategy. With extra fine grinding the lifespan can be upto 24 weeks.
Temperature	Varies. Not thought to be excessive (above 70°C). The crusher rotates at a low speed of 6 revs/min whilst crushing. Crushing is performed throughout the year in various climates.
Moisture Levels	Varies dependent upon the moisture conditions of the rock crushed. From high degree of moisture to very dry.
Environmental Factors	Dust

Table 4.1 - Design Constraints

4.1.1 Size Constraints

The most influential factor on the design of the sensors is the extent of wear to be measured, and the resolution. Liner depths vary depending upon the model of cone crusher used. This is not a problem for the development of the sensor, however, general lengths of sensor were made in line with the Pegson 900 Autocone Mk II. From the diagram (Figure 4.1) it is necessary that 52mm and 53mm sensors are designed to fit perpendicular to the surface of the liner.

The width of the sensor also plays a critical part in the design. From a mathematical analysis of used wear liners [Moshgbar et al, 1995c], the wear is typically non-linear. This provides us with further complications to the development of a measuring system. It is imperative for the mill manager to know the irregularity of the wear, as this can play a critical role in the quality of output produced. For this reason, it is necessary that more than one sensor is used in the cone crusher liners.
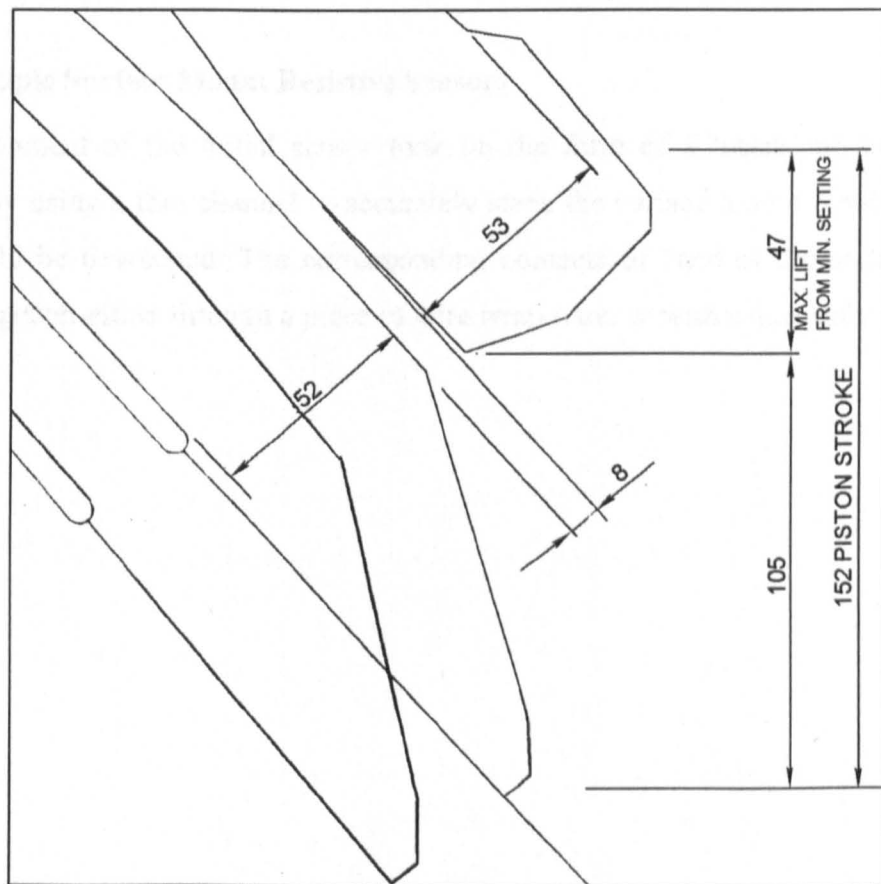
Figure 4.1- Pegson Autocone 900 Mk II Liner Separation

Therefore, it is important that a series of sensors, inserted into the wear liners, will not impede on the quality of the crushing members. A further constraint on the width of the sensors is imposed by the mechanism used to produce the hole. Drilling through manganese steel is not ideal due to the hardness of the material, so methods such as electro discharge machining need to be employed.

The resolution of the sensor is of paramount importance to the mill manager, as this is what determines the accuracy of the aggregate diameter. The system initially used was accurate to 1mm, therefore an improvement on this is essential.

4.2 Resistive Sensor Design

One of the most traditional methods of sensor development have been to look at the change in resistance of a certain material due to changing parameters in the surrounding area.

4.2.1 Multiple Surface Mount Resistive Sensors

The development of the initial sensor took on the form of a 'stack' of surface mount resistors. By using a thin channel to accurately stack the surface mount resistors, a 50mm sensor could be developed. The corresponding contacts of each of the sensors are then soldered together either through a piece of wire wrap wire, or with solder only (Figure 4.2).
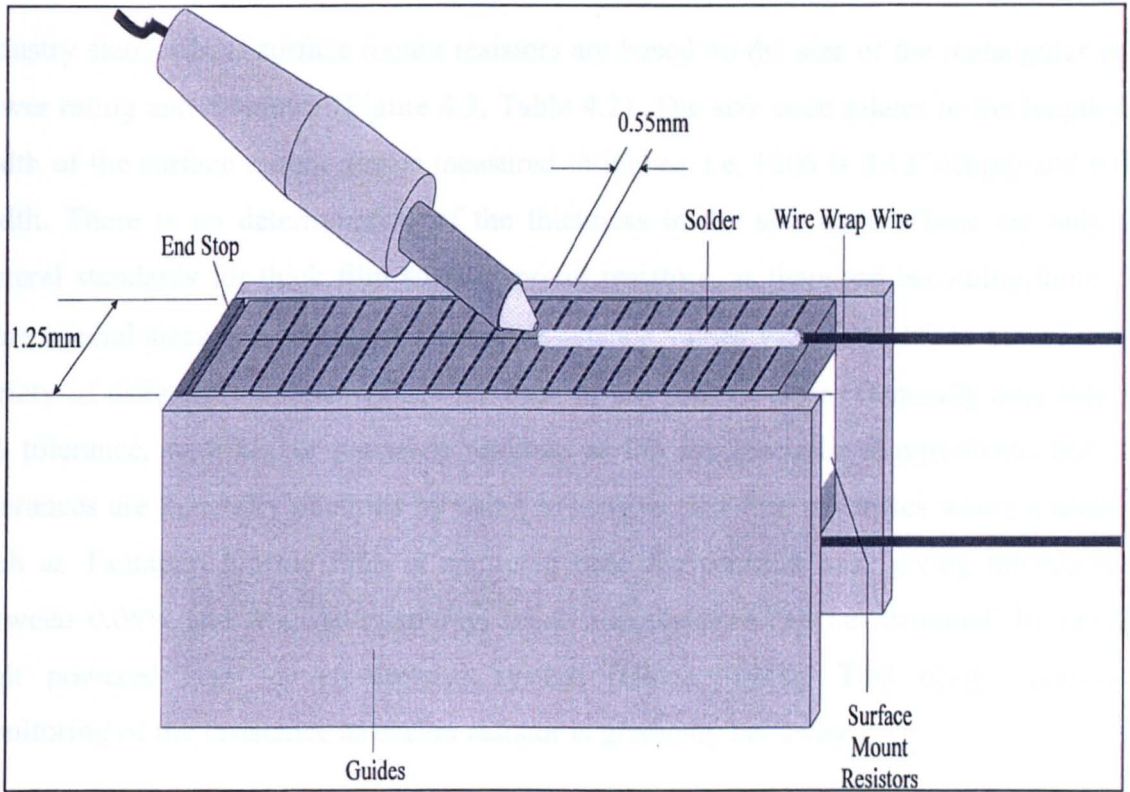
Figure 4.2 - Construction of Multiple Surface Mount Sensor
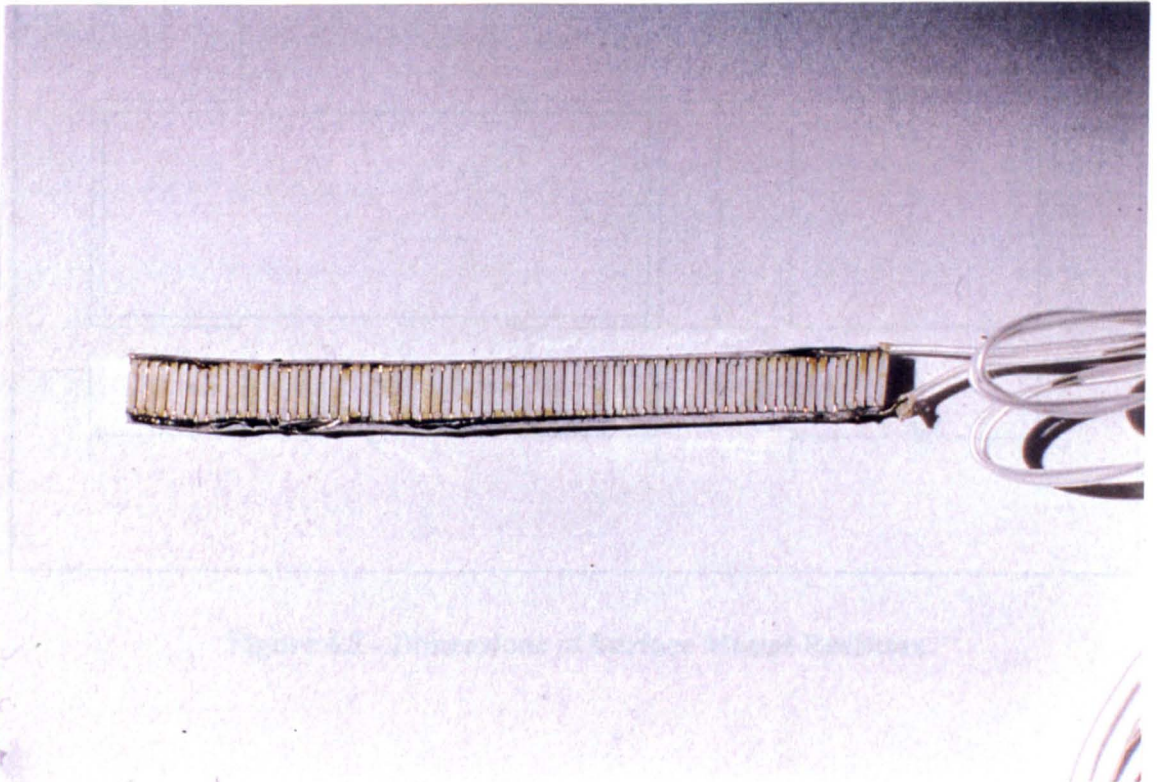
Plate 4.1 - Photograph of Multiple Surface Mount Resistive Sensor

Industry standards of surface mount resistors are based on the size of the rectangular chip, power rating and tolerance (Figure 4.3, Table 4.2). The size code relates to the length and width of the surface mount device measured in inches, i.e. 1206 is 0.12" length and 0.06" width. There is no determination of the thickness in the size code. These are only the general standards for thick film surface mount resistors, as there are becoming more and more special sizes and packages. General resistance values vary from 1Ω to $50G\Omega$ with a variety of different tolerance values for each of the resistor sizes. Generally available are 5% tolerance, with higher precision resistors at 1% for specialist requirements. Sub 1% tolerances are generally obtained by using expensive thin film processes where a material such as Tantalum Nitride Film is sputtered onto the ceramic base giving tolerances of between 0.05% and 2%. Alternatively, thick film resistors can be 'trimmed' by using a high powered laser or air-abrasive system [Hinch, 1988]. This needs continuous monitoring of the resistance as excess resistor is gradually cut away.
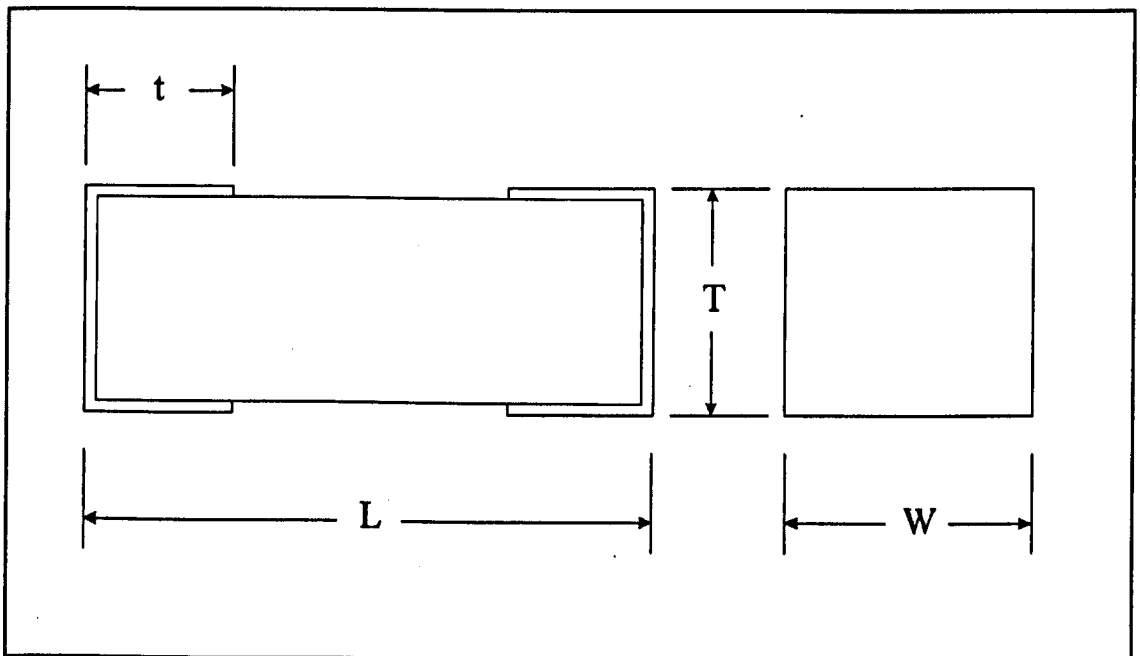
Figure 4.3 - Dimensions of Surface Mount Resistors

Size Code	L (mm)	W (mm)	T (mm)	t (mm)	Rated Wattage (W)	Max. Working Voltage (V)
0402	1.00	0.50	0.35	0.19	0.063	50
0603	1.60	0.80	0.50	0.25	0.0625	50
0805	2.00	1.25	0.50	0.40	0.1	100
1206	3.20	1.60	0.50	0.50	0.125	200
1210	3.20	2.60	0.50	0.50	0.250	200

Table 4.2 - Dimensions of Surface Mount Resistors

Table 4.2 shows the general dimensions for surface mount resistors. The thickness, however, often has a tolerance of ±0.05mm, and differs with the manufacturers and chip type. The development of the sensors designed for the cone crusher tests used generally available $100k\Omega$ 1206 and 0805 resistors. As the sensors are stacked, the thickness (T) of the surface mount resistors is the length tolerance of the sensor. In the case of the 1206 and 0805, this is 0.55mm. For a 49.5mm sensor, a total of 90 surface mount resistors are stacked together, and joined by the solder strip on either side of the sensor.

4.2.1.1 Electrical Considerations

With resistive sensors, conversion of wear to a voltage or current is easier than for most types of sensor. By using a simple potential divider circuit, (Figure 4.4), a voltage relative to the length of the sensor is output.
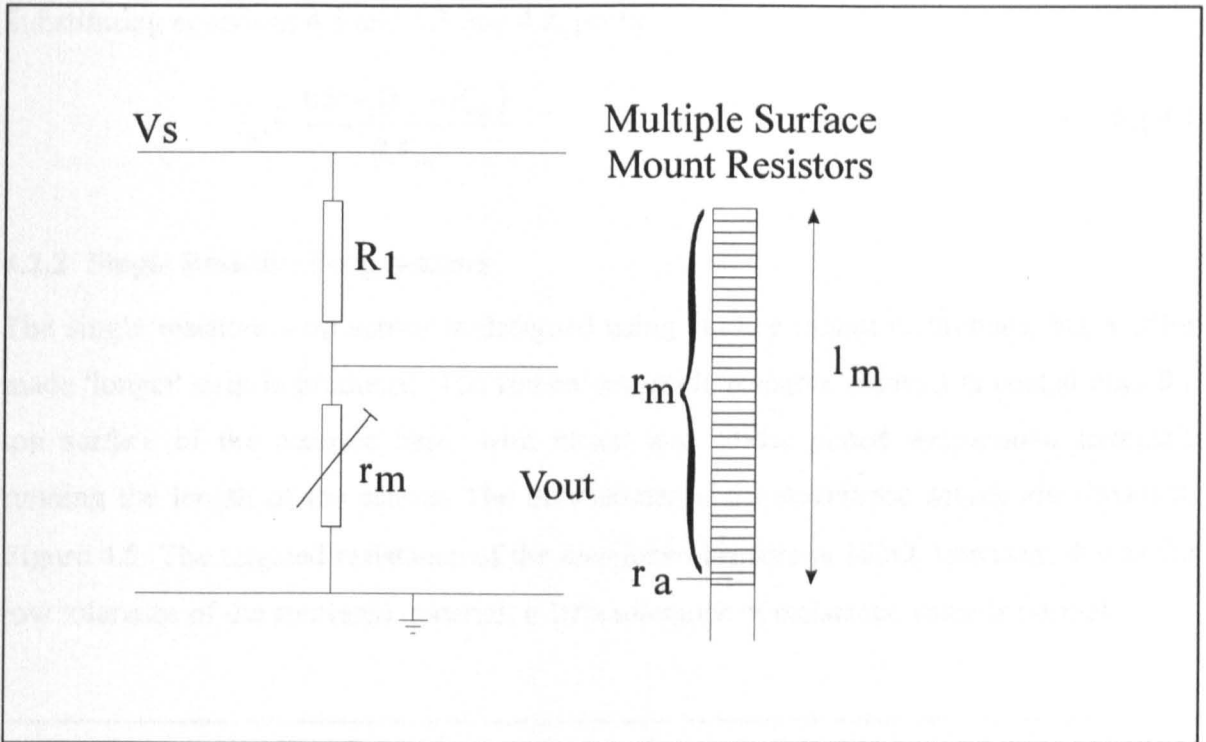
Figure 4.4 - Potential Divider Circuit

The equation for l_m is as follows:-

As each surface mount resistor is 0.55mm thick

$$0.55n = l_m \qquad \text{Eq 4.1}$$

where n is the number of surface mount resistors and l_m is the length of the sensor in millimetres.

Using the equation for the potential divider,

$$V_{out} = V_s \left(\frac{r_m}{r_m + R_1} \right) \qquad \text{Eq 4.2}$$

where V_{out} and V_s are the output and supply voltage respectively and R_1 is the potential divider resistor value.

The resistance of the sensor, r_m in a parallel configuration is,

$$r_m = \frac{r_a}{n} \qquad \text{Eq 4.3}$$

where r_a is the resistance of a single surface mount resistor.

Substituting equations 4.1 and 4.3 into 4.2, gives

$$l_m = \frac{0.55 r_a \left(V_s - V_{out} \right)}{R_1 V_{out}}$$

Eq 4.4

4.2.2 Single Resistive Strip Sensors

The single resistive strip sensor is designed using surface mount techniques, but a tailor made 'longer' strip is produced. The ruthenium oxide resistive element is coated onto the top surface of the ceramic base, with nickel and solder plated wraparound terminals running the length of the sensor. The dimensions of the developed sensor are shown in Figure 4.5. The targeted resistance of the developed sensors is 160Ω, however, due to the low tolerance of the sputtered material, a 10% tolerance in resistance value is normal.

Wraparound

50.5mm

Resistance Applied to Top Face

0.65mm

3mm
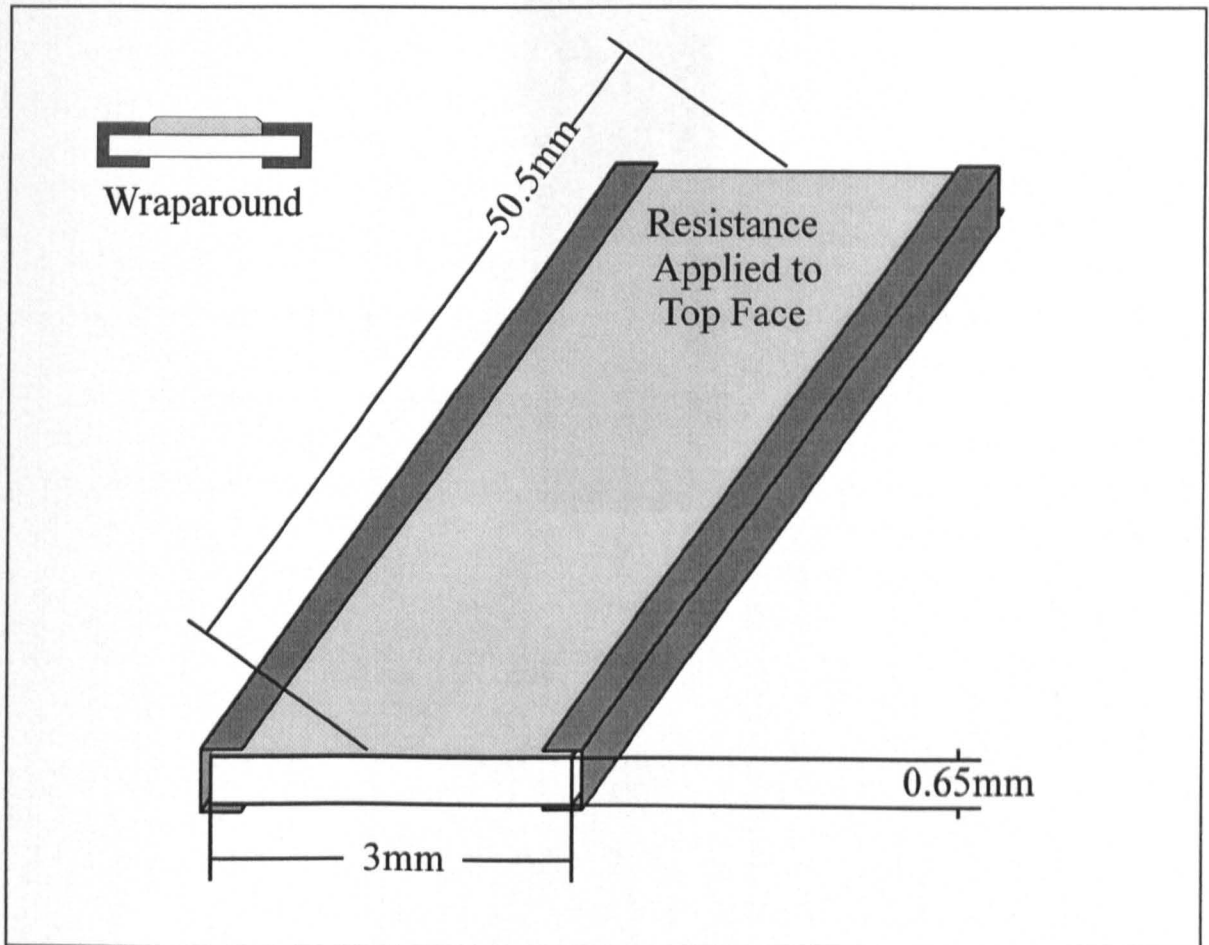
Figure 4.5 - Single Strip Surface Mount Sensor

Further complications for a single strip surface mount sensor are produced with the problem of even over-burden. The production of "off the shelf" surface mount resistors does not have this problem as the final resistance is the only criterion. A more accurate single resistive strip could be produced by using the expensive thin production techniques and a material such as Tantalum Nitride Film for the resistive element.

4.2.2.1 Wear Resistance Profile

To determine the length of the sensor from a certain resistance and the profile of the resistance as the sensor wears away it is necessary, using known lengths of the strip and measuring the resistance, a graph can be plotted of the length to resistance. Figure 4.6 shows this relationship with the data using the values read from a vernier calliper and a resistance meter. The line of the best fit curve which results in an equation based on a power function fits the measured data, and value [5] for the regression equation is

$$ L = ? ? ? ? ? ? ? ? $$

where L is the sensor length in millimetres and the sensor resistance in ohms.

Plate 4.2 - Photograph of Single Strip Resistive Sensor

Further complications for a single strip surface mount sensor are produced with the problem of even distribution. The production of 'off the shelf' surface mount resistors does not have this problem as the final resistance is the only concern. A more accurate single resistive strip could be produced by using the expensive thin production techniques and a material such as Tantalum Nitride Film for the resistive element.

4.2.2.1 Wear Resistance Profile

To determine the length of the sensor 'in situ' it is imperative that the profile of the resistance as the sensor wears away is known. By cutting known lengths of the sensor and measuring the resistance, a graph can be drawn relating length to sensor resistance. Figure 4.6 shows this relationship with the points indicating the values read using a vernier calliper and a resistance meter. The line is the best fit curve which relates to an equation based on a power function. For the tested sensor with start value 172.3Ω, the equivalent equation is:-

$$l_w = 7279.1 r_w^{-0.9659}$$

Eq 4.5

where l_w is the Sensor length in millimetres, and r_w is the sensor resistance in ohms.
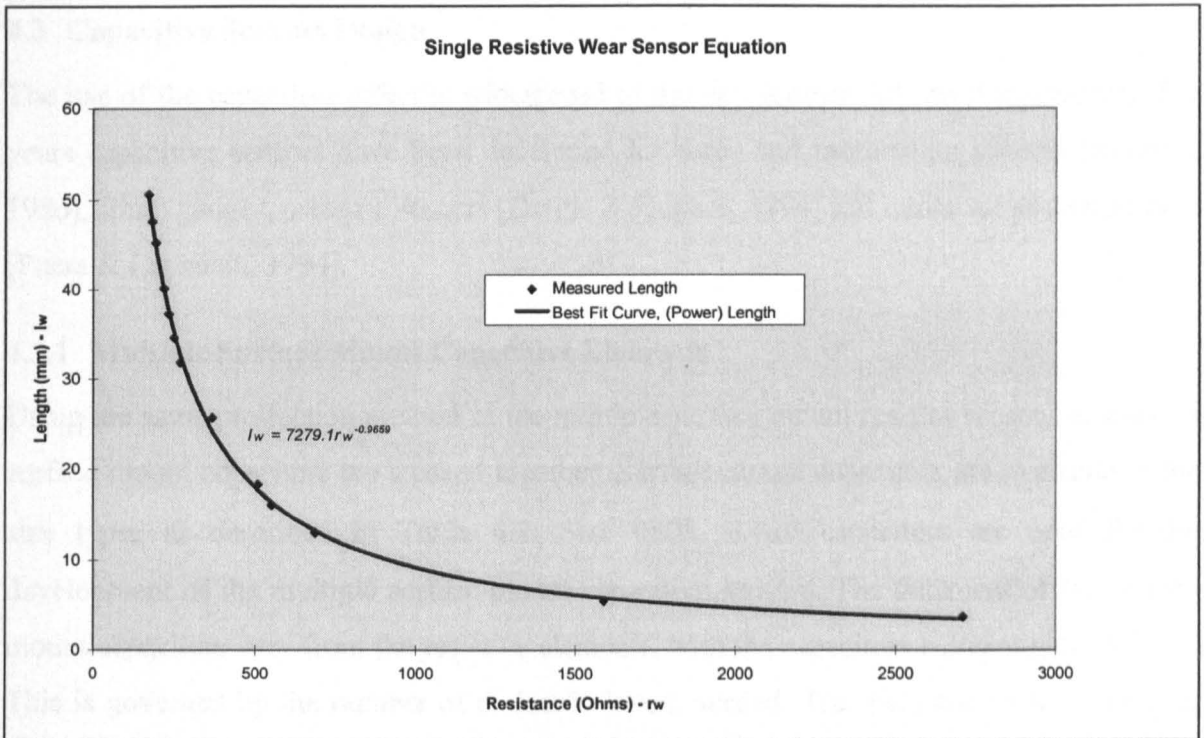
Figure 4.6 - Graph of Resistance versus Sensor Length

Due to the 10% tolerance on the single resistive wear sensors, this equation cannot immediately be applied to determine the length of the sensor. The relationship is similar for each of the sensors, however, the curve shifts for different start resistances. The multiplication factor is determined by Equation 4.6.

$$\frac{r_{init}}{m} = \frac{172.3}{7279.1}$$
Eq 4.6

where r_{init} is the start resistance of the sensor, and m is the multiplication factor in Equation 4.7.

$$l_w = mr_w^{-0.9659}$$
Eq 4.7

Substituting Equation 4.6 into 4.7, the equation of the curve specific to that sensor is:-

$$l_w = 42.247r_{init}r_w^{-0.9659}$$
Eq 4.8

4.3 Capacitive Sensors Design

The use of the capacitive effect is widespread in the development of novel transducers. For years capacitive sensors have been developed for force and momentum sensors [Heerens, 1986], strain gauges, pressure sensors [Dargie & Hughes, 1994] and miniature accelerometers [Puers & Lapadatu, 1994].

4.3.1 Multiple Surface Mount Capacitive Elements

Using the same production method as the multiple surface mount resistor sensor, an array of surface mount capacitors are stacked together. Surface mount capacitors are available in the size types as described in Table 4.2. Size 0805, $0.1\mu F$ capacitors are used for the development of the multiple surface mount capacitive sensors. The thickness of the surface mount capacitors vary from the resistive elements, with the capacitors being slightly thicker. This is governed by the number of dielectric layers needed. The dielectric material used in ceramic capacitors plays a large part in the characteristics of the device. The three main ceramic dielectric materials are COG, X7R and Z5U. Capacitance change with temperature, dielectric constants and capacitance range vary in relation to the dielectric material and size of capacitor, Table 4.3 [Hinch, 1988].

Property	Dielectric Type		
	COG(NPO)	X7R	Z5U
Temperature Range	-55°C to +125°C	-55°C to +125°C	+10°C to +85°C
Capacitance Change over Specified Temperature	±30ppm/°C	±15%	+22% to -56%
Dielectric Constant (K)	60-80	2200-3000	5000

Table 4.3 – Ceramic Capacitor Dielectric Properties
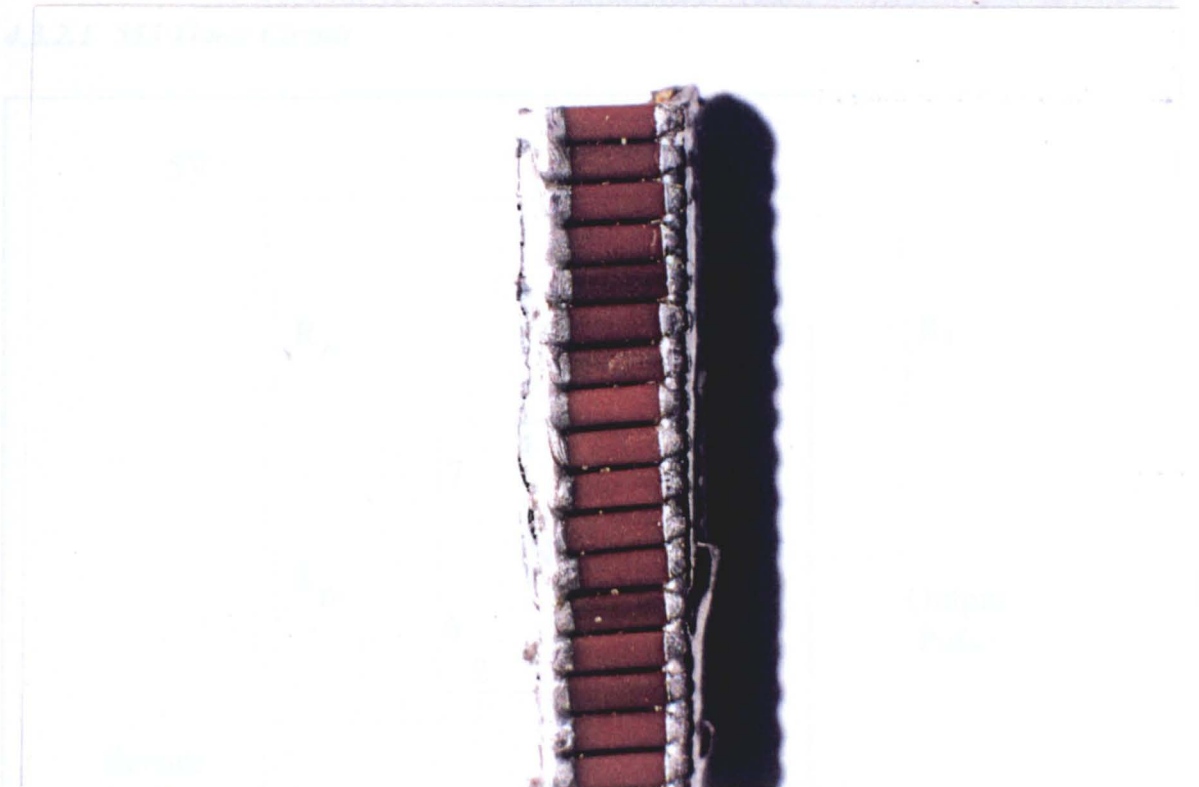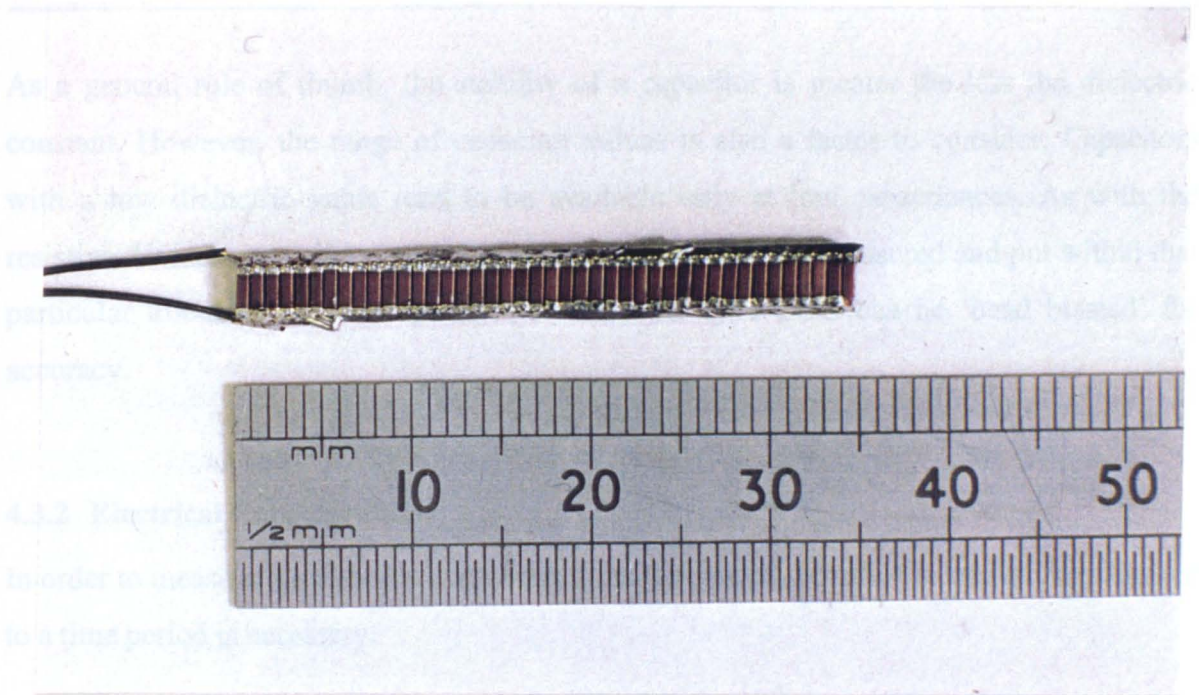
Plate 4.3 - Photograph of Surface Mount Capacitive Sensor

As a general rule of thumb, the stability of a capacitor is greater the less the dielectric constant. However, the range of capacitor values is also a factor to consider. Capacitors with a low dielectric value tend to be available only at low capacitances. As with the resistive devices, capacitors are not trimmed. Instead they are measured and put within that particular tolerance range. Capacitors of dielectric type COG can be 'bead blasted' for accuracy.

4.3.2 Electrical Considerations

In order to measure the distance worn away from the sensor, a method to convert capacitance to a time period is necessary.

4.3.2.1 555 Timer Circuit
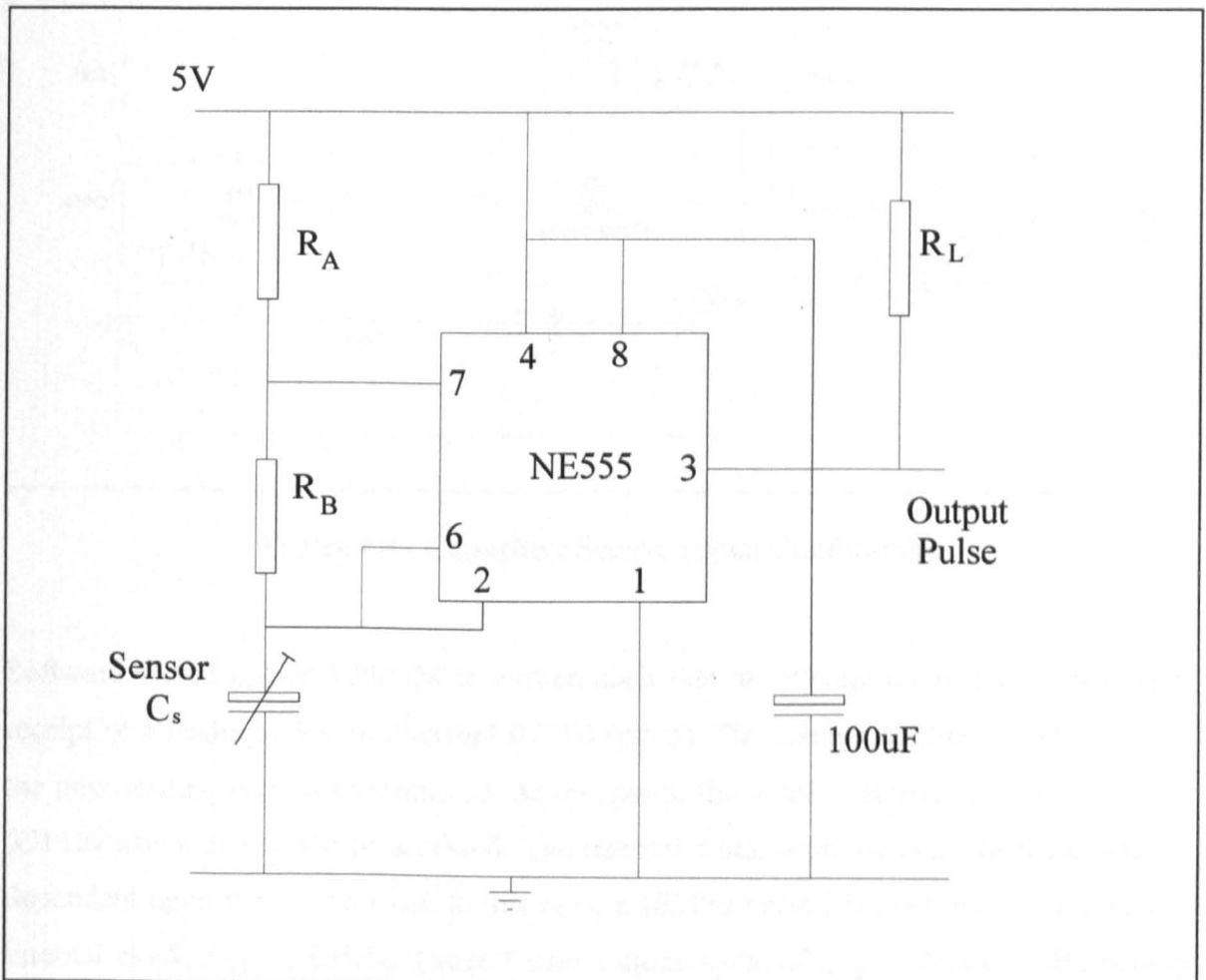
Figure 4.7 - 555 Timer Circuit

Figure 4.7 shows the circuit diagram of the 555 circuit. The NE555 is oscillating at frequency governed by the resistors R_A and R_B, and the capacitive sensor C_s. The frequency f, is given by the equation

$$f = \frac{1.44}{\left[(R_A + 2R_B)C_s\right]}$$

Eq 4.9

By using figures of $1K\Omega$ for R_A and 470Ω for R_B, the output frequency will cover the range of 74Hz to 7.4kHz for a sensor change from 10uF to 0.1uF respectively. In turn, to read this frequency and convert it to a length, a microcontroller is used with an integrated counter/timer (Figure 4.8).
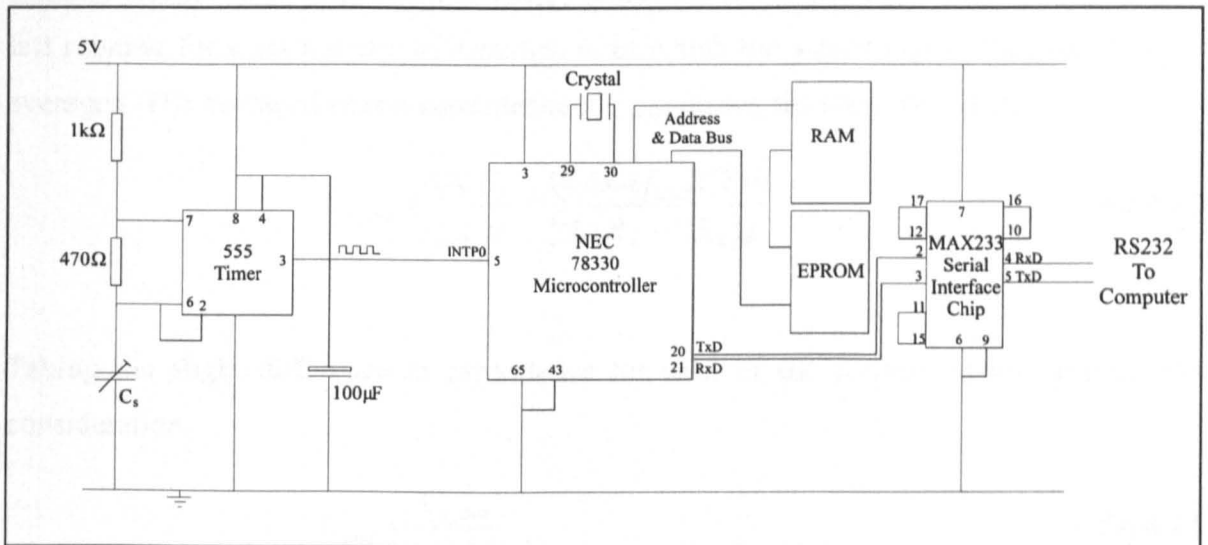
Figure 4.8 - Capacitive Sensor Signal Conditioning

Software stored on the EPROM is written such that an internal counter is started upon receipt of a leading edge on interrupt INTP0 (pin 5). The counter continues counting until the next leading edge is encountered. At this point, the value is stored in a 16 bit register (CT10) which can easily be accessed. The internal clock which is used for the counter is dependent upon the crystal used. In this case, a 16MHz crystal is used, which produces an internal clock, f_{CLK} of 8MHz. Timer 1 uses a clock cycle of $f_{CLK}/8$, thus a 1MHz counter cycle is used to calculate the interval between the interrupt pulses.

$$f = \frac{1}{f_{CLK} CT10} \qquad \text{Eq 4.10}$$

From Equation 4.9 the sensor capacitance C_s is,

$$C_s = \frac{1.44}{(R_A + 2R_B)f} \qquad \text{Eq 4.11}$$

Inserting Equation 4.10 into 4.11,

$$C_s = \frac{1.44 f_{CLK} CT10}{R_A + 2R_B} \qquad \text{Eq 4.12}$$

In order to encompass the stability of the capacitors and the surrounding electronics, a stability test provides an accurate starting position for the sensor capacitance. The sensor is left running for a set number of samples, p, in which the sensor capacitance can then be averaged. The averaged sensor capacitance (C_{s_ave}) during the stability test is,

$$C_{s\_ave} = \sum_{x=0}^{x=p} \frac{C_s}{p} = \sum_{x=0}^{x=p} \frac{1.44 f_{CLK} CT10}{(R_A + 2R_B)p} \qquad \text{Eq 4.13}$$

Taking the slight difference in capacitance for each of the surface mount devices into consideration.

$$C_{smd} = \frac{C_{s\_ave}}{n_T} \qquad \text{Eq 4.14}$$

where C_{smd} is the average capacitance for each surface mount capacitor and n_T is the total number of surface mount capacitors in the sensor prior to installation.

During operation, the number of surface mount devices, n_{smd}, is calculated by,

$$n_{smd} = \frac{C_s}{C_{smd}} = \frac{1.44 f_{CLK} CT10}{(R_A + 2R_B)C_{smd}} \qquad \text{Eq 4.15}$$

The length of the sensor, l_s, can be calculated by,

$$l_s = n_{smd} t_{smd} \qquad \text{Eq 4.16}$$

where t_{smd} is the thickness of a single surface mount capacitor.

Substituting Equation 4.15 into 4.16,

$$l_s = \frac{1.44 f_{CLK} CT10t_{smd}}{(R_A + 2R_B)C_{smd}}$$

Eq 4.17

For laboratory tests, the data from the 16 bit counter is transmitted to a serial interface chip, which in turn is sent down an RS232 cable to a Personal Computer. The data is sent every second as two following bytes, which is then re-installed as a 16 bit value by the PC software, and the length of the sensor calculated.

4.3.2.2 Modified Martin Oscillator

An alternative to the 555 timer circuit, is to use a modified martin oscillator. The circuit was initially conceived as a Voltage Controlled switched Capacitor Relaxation Oscillator [Martin, 1981]. Since its conception in 1981, the circuit has been redeveloped into a Capacitance controlled modified Martin oscillator [de Jong et al, 1994][Toth & Meijer, 1992b] for use with capacitance position sensors.
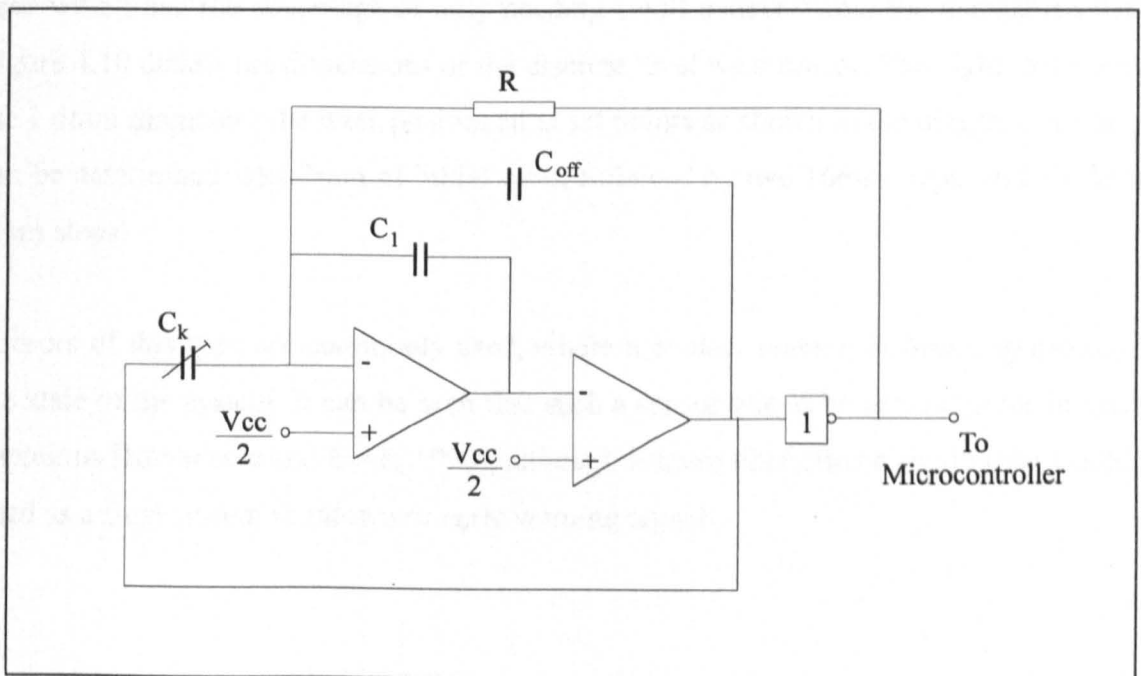
Figure 4.9 - Modified Martin Oscillator

The period T_m of the output signal is related to the sensor capacitance C_k by:

$$T_m = 4R(C_k + C_{off}) + t_d$$

Eq 4.18

where t_d is accounts for the switching time delay.

This circuit is generally used for capacitances in the lower picoFarad region, and where high accuracy in position is of paramount importance.

4.4 Discrete Level Sensor

A sensor has been developed which uses a more coarse measurement system than that of the resistive and capacitive sensors. The sensor comprises of a metal tube, filled with an epoxide pottant and 8 conductive wire loops which extend different distances along the sensor body. The sensor has a series of 8 discrete levels of wire, positioned within a 0.6mm I.D. tube. The corresponding sides of the discrete wire are taken from the pins at the base of the sensor. Depending upon which levels are broken will determine the distance the sensor has been ground down. One of the main disadvantages of this sensor, however, is the lack of accuracy (+/- 5mm), and the amount of wires that need to be inserted into the tube. Due to the width of the conducting wires, potted tubes with a small diameter could be used which had the advantage of only needing small inserts within the manganese liners. Figure 4.10 details the dimensions of the discrete level wear sensor. The eight loops within the 1.6mm diameter tube were positioned at set points as shown in the diagram. An output can be determined after 3mm of initial wear, followed by two 10mm steps, and six further 5mm steps.

Sensors of this type are commonly used, where a contact making or breaking determines the state of the system. It can be seen that such a sensor would be advantageous in certain situations [Kovacevic and Evizi, 1990], although it more likely that a single loop would be used as a final system shutdown or early warning signal.
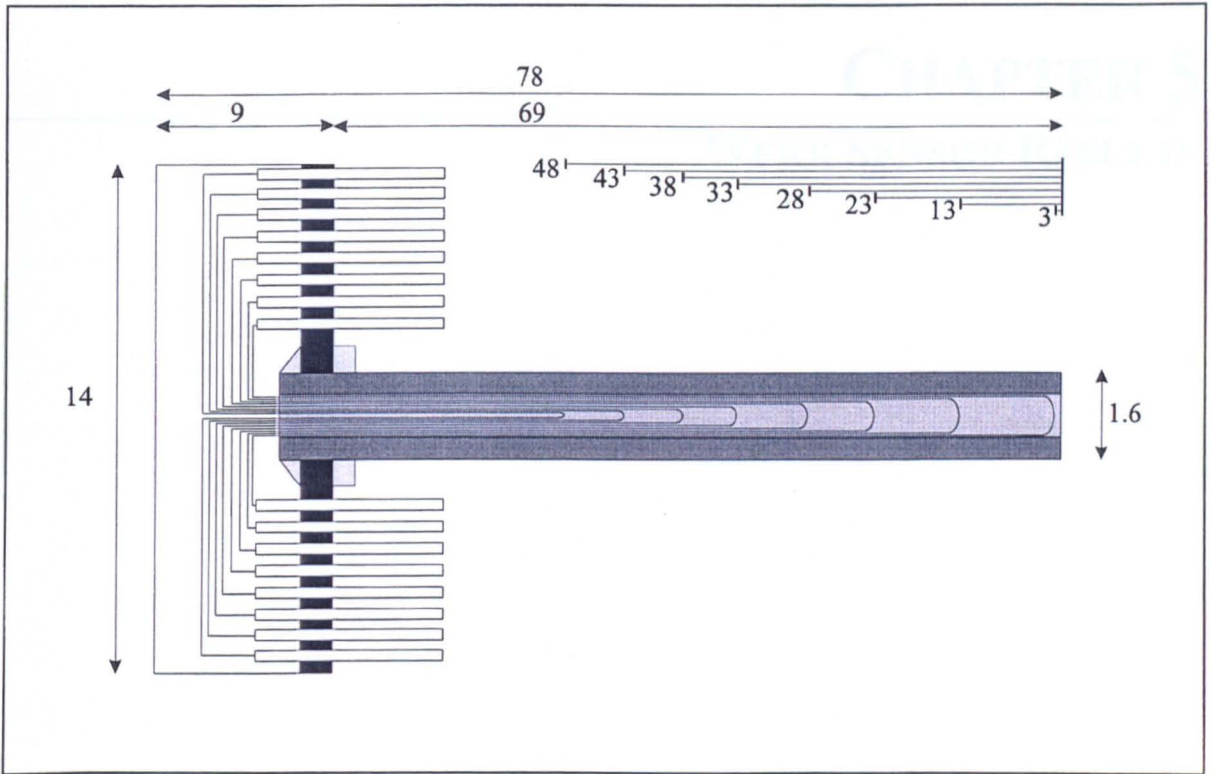
Figure 4.10- Discrete Level Wear Sensor Schematic
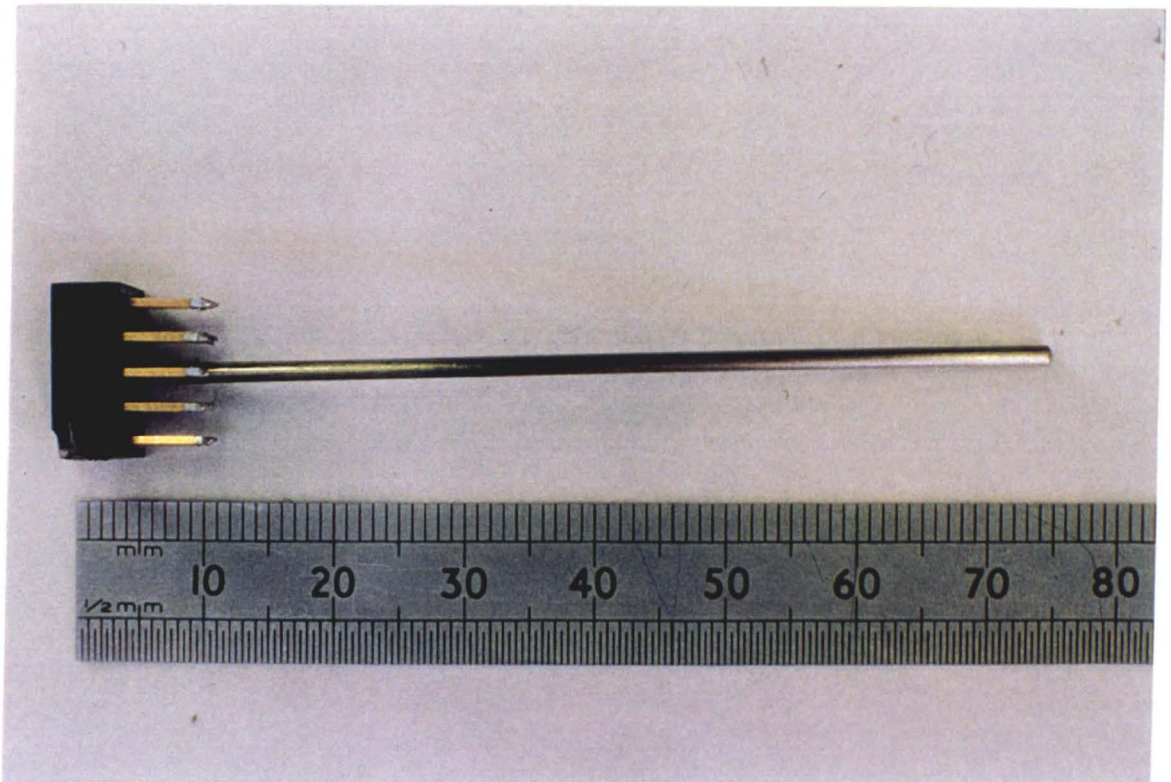
Plate 4.4 - Photograph of Discrete Level Sensor

CHAPTER 5
WEAR SENSOR RESULTS

5. WEAR SENSOR RESULTS

For each of the sensors, a series of tests were carried out to analyse the properties of the sensors during wear conditions. Analysis of the sensors in terms of temperature and initial stability were carried out to determine their suitability to the desired application. To simulate typical forces applied to the sensor, grinding and strength tests were carried out. Tests on the sensors used the Instrom, which could apply a force of up to 200kN, and the less accurate Denison, which could handle a force up to 300kN. Each time the data was recorded on a signal logger interfaced directly to a personal computer. A sampling time of 1 second was used. The most probable outcome from an undesirable sensor would be an open circuit at high compacting forces. This would imply that either the contacts to the resistive elements or the elements themselves failed due to cracking or a weak link in the solder contact. For the grinding tests, the sensors were gradually applied to a grinding element, and using the signal logger and computer, results for the sensors were recorded. It is important to analyse the amount of short circuits caused by the grinding effect on the tip of the sensor. If whilst grinding, a substantial amount of 'short circuits' occur, it may not be possible to determine the true results. In each case signal conditioning of the signal is necessary to provide the signal logger with appropriate values. A series of field tests were also carried out within a cone crusher installed at a quarrying plant.

5.1 Multiple Surface Mount Resistor Sensor

The sensors developed for the laboratory and field tests were constructed from 100kΩ surface mount resistors, size 1206 and 0805. Each sensor is formed into a parallel resistive stack and potted in a steel insert with epoxy resin (Plate 5.1).

Plate 5.1 - Photograph of Potted Sensor

5.1.1 Compressive Strength Test

It is necessary to determine the effect a compressive force will have on the sensor. This test is performed to analyse the effect a general compressive force has on the sensor in terms of unexpected failures. The sensor is housed and potted in a steel insert with a diameter of 24mm, and a centrally drilled 9mm hole to fit the sensor and epoxy resin. A steel mating half allows for a few small pieces of aggregate to be crushed down by the compacting force around the tip of the sensor (Figure 5.1).

This is not attempting to simulate all the forces that are likely to act upon the sensor during crushing. Localised forces on the sensor element in terms of Hertzian contact stresses [Roark, 1965] are likely to effect the output to a greater extent than any other forces. This is simulated by the inclusion of a small metal peg in the mating half of the test housing.
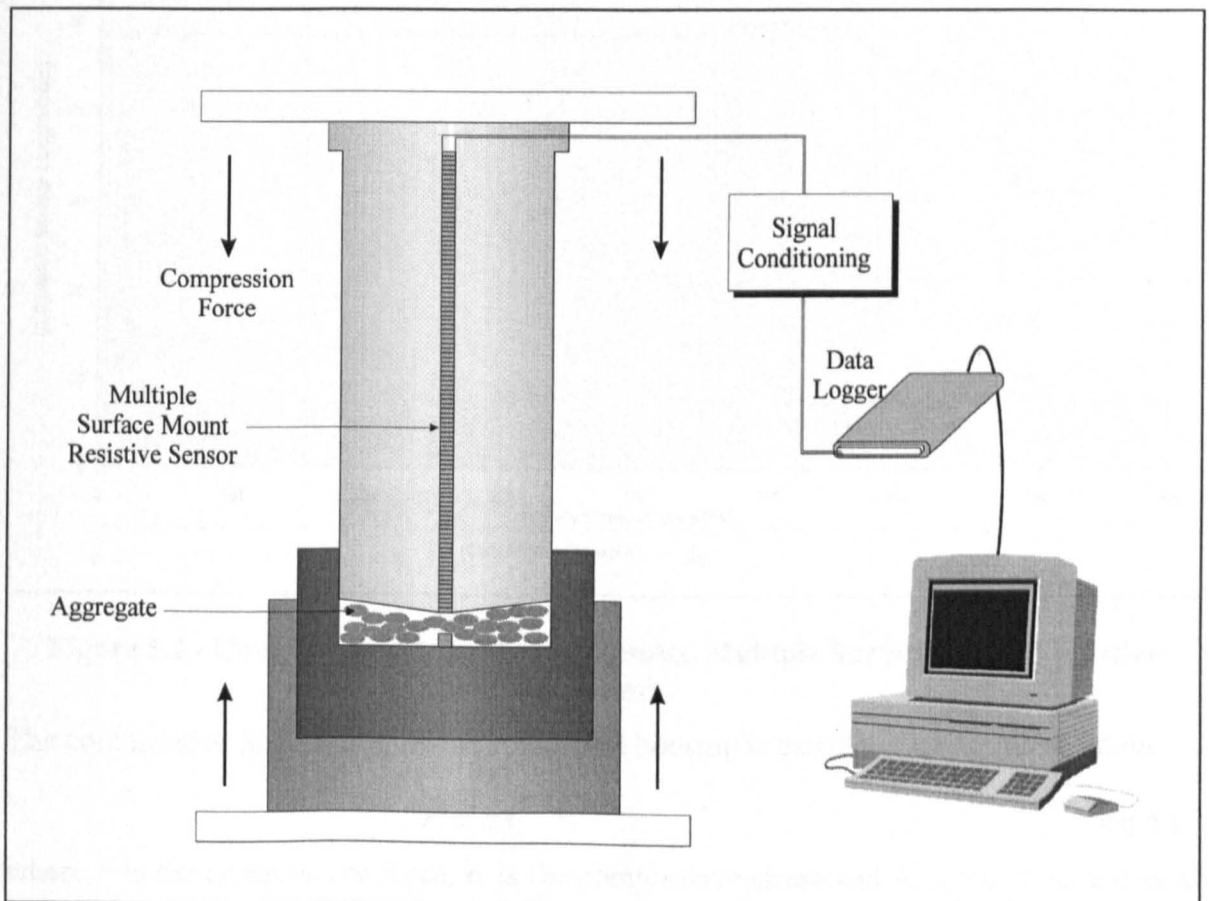
Figure 5.1 - Multiple Surface Mount Resistive Sensor, Compressive Strength Test

The compressive forces acting upon the housed sensor are taken up to the compressive strength of the steel employed. The test set-up does not take into account the increased strength the housed sensor would acquire when inserted into the manganese liners. A failure from the sensor is indicated as a short circuit, open circuit, or drift in the output during the compressive force.

The contacts for the sensor are taken through a machined groove which runs along the bottom of the sensor housing. Signal conditioning is provided by a potential divider circuit for signal input into the data logger. The results are shown in Figure 5.2 and show that the sensor did not weaken in any way from the compacting force.
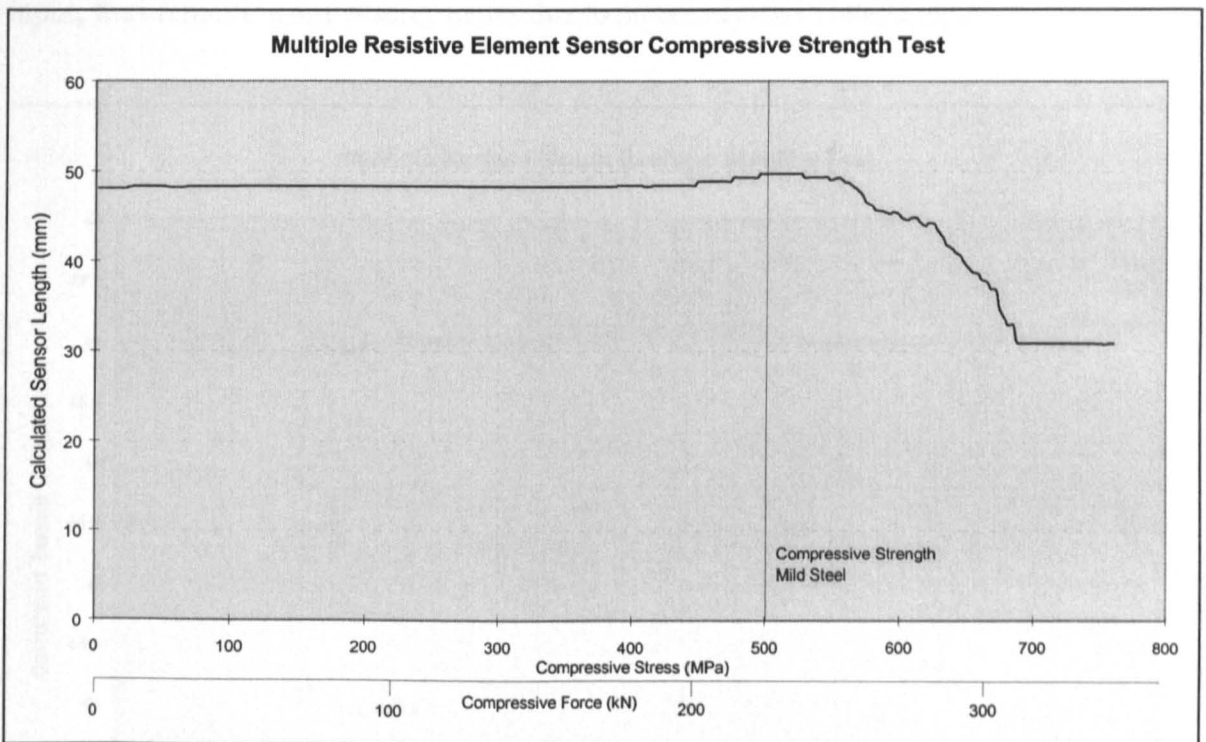
Figure 5.2 - Compressive Strength Test Results, Multiple Surface Mount Resistive Sensor

The compressive force acting on the sensor and housing is calculated using the equation

$$F = \sigma A$$

Eq 5.1

where F is the compressive force, σ is the compressive stress and A is the cross sectional area the force is acting upon.

From the results, it can be seen that there is no adverse affect on the sensor throughout the test. The output from the sensor remains steady up until the compressive strength of the

mild steel. After this the voltage output increases and the calculated sensor length decreases.

5.1.2 Stability Test

A test to determine the stability of the sensor whilst under no external forces or environmental conditions will provide an accuracy figure for that particular sensor. A series of 2400 samples taken at 1 second intervals are shown in Figure 5.3. The results show the calculated sensor length of the sensor taken from the voltage output from the potential divider circuit. Also measured and involved in the calculations is the voltage input, thus removing any discrepancies due to an unbalanced voltage input.
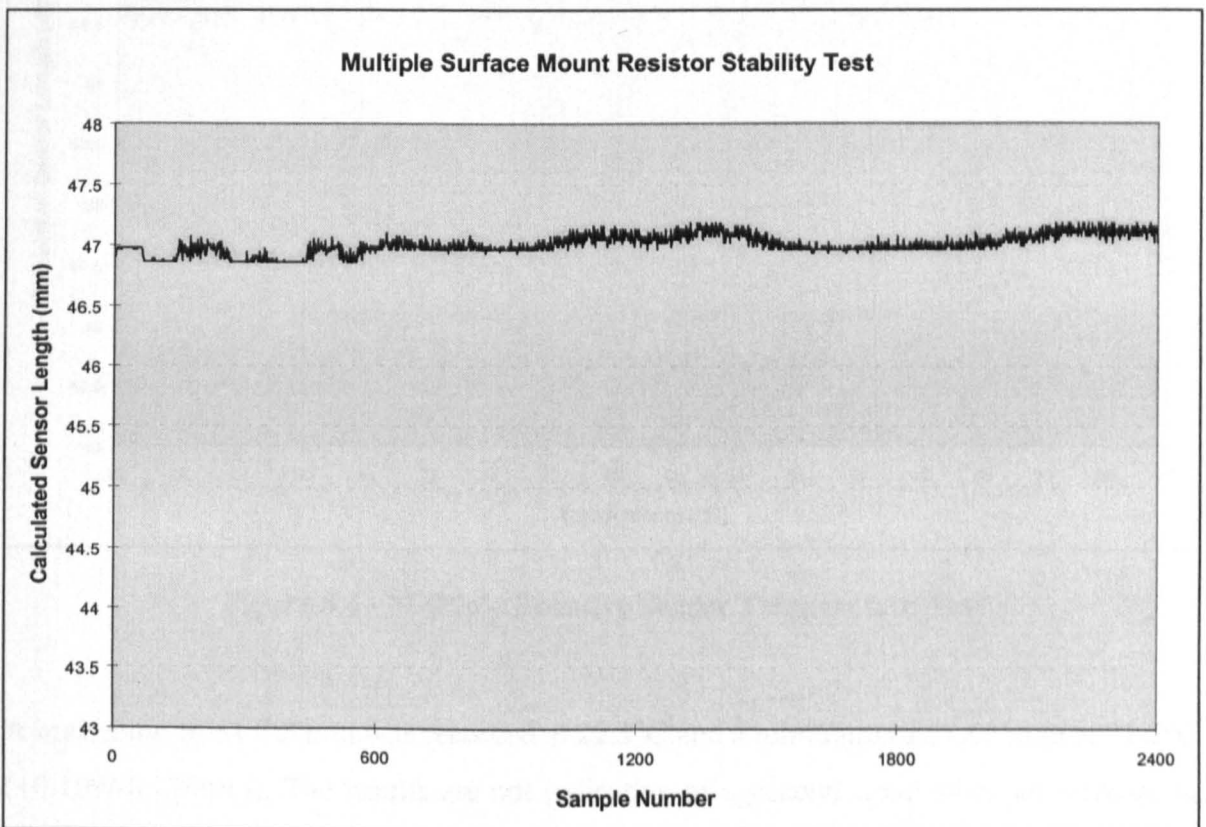
Figure 5.3 - Multiple Surface Mount Resistor Stability Test

A maximum calculated sensor length of 47.183mm was recorded, with a minimum of 46.865mm. The ideal length for this sensor is 47mm (94 surface mount resistors type 0805), thus a stability accuracy of +0.183/-0.135mm is noted.

5.1.3 Temperature Test

A temperature test to determine the suitability of a wear sensor to certain applications was performed (Figure 5.4). A stack of 94 type 0805 surface mount resistors were constructed into a sensor. Heat was directly applied to the sensor and parallel temperature probe.
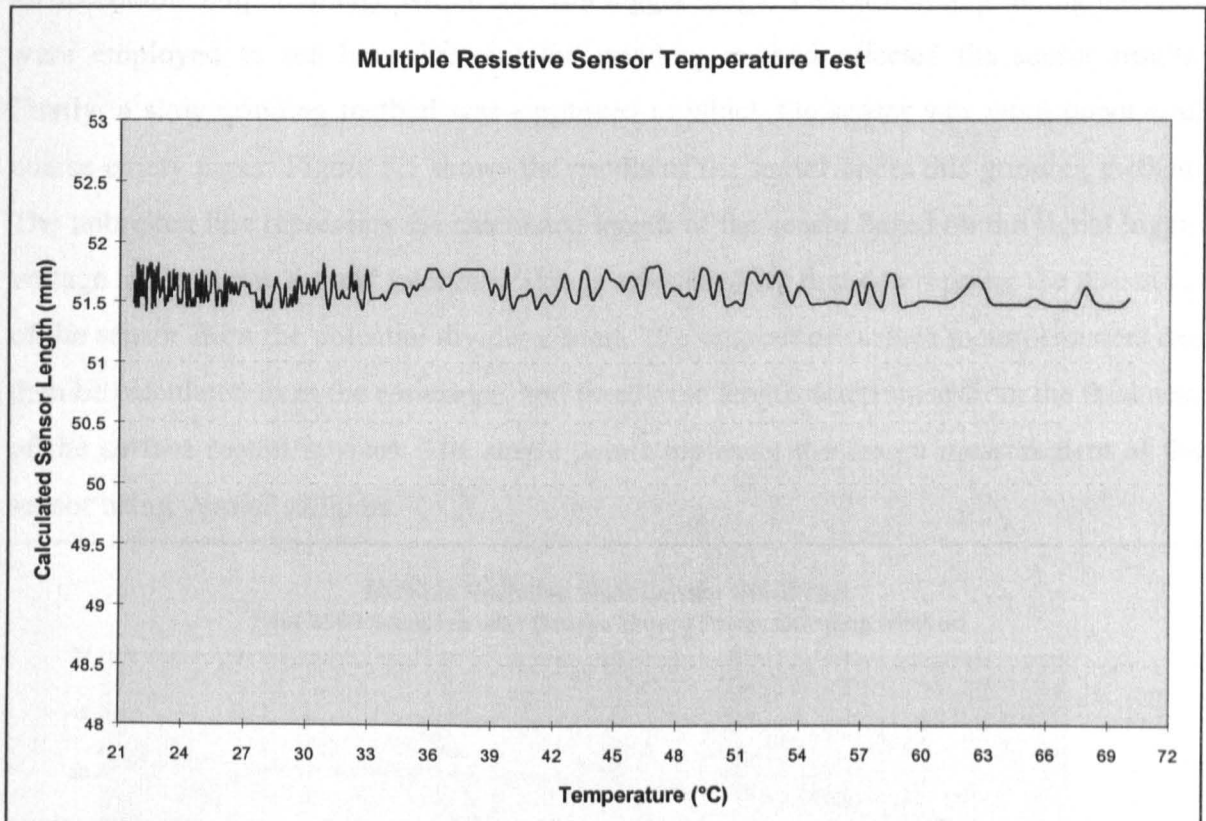
Figure 5.4 - Multiple Resistive Sensor Temperature Test

A maximum of 51.809mm was recorded at 22.5°C and a minimum of 51.421mm at 21.2°C (+0.109/-0.279mm). The results are not indicative of a general trend when an increase in temperature is applied. The calculated sensor length is generally erratic with an output similar to the stability test. From the manufacturers data given for the surface mount resistors, a change of 200ppm/°C is expected. This equates to just over 0.5mm change in calculated length when the sensor is subjected to 70°C.

5.1.4 Grind Test

A grind test was performed on a resistive stack of 75 surface mount resistors, size 1206, with a measured initial length of 41.7mm. The sensor is potted using epoxy resin in a plastic housing of diameter 8mm. Wire wrap wire is soldered down the entire length of each side of the sensor, and protrudes out of the back of the plastic housing to feed into the potential divider circuit. A 4.7kΩ resistor is used in the potential divider circuit to produce an acceptable output voltage within the data logger range. Two different grinding methods were employed to see how different the grinding method effected the sensor results. Firstly, a slow grinding method was employed in which the sensor was worn down with coarse emery paper. Figure 5.5 shows the results of the sensor under this grinding method. The unbroken line represents the calculated length of the sensor based on the signal logger voltage taken at one second intervals. This is calculated by first determining the resistance of the sensor from the potential divider circuit. The number of surface mount resistors can then be calculated from the resistance, and finally the length determined from the thickness of the surface mount devices. The single points represent the length measurement of the sensor using vernier callipers.
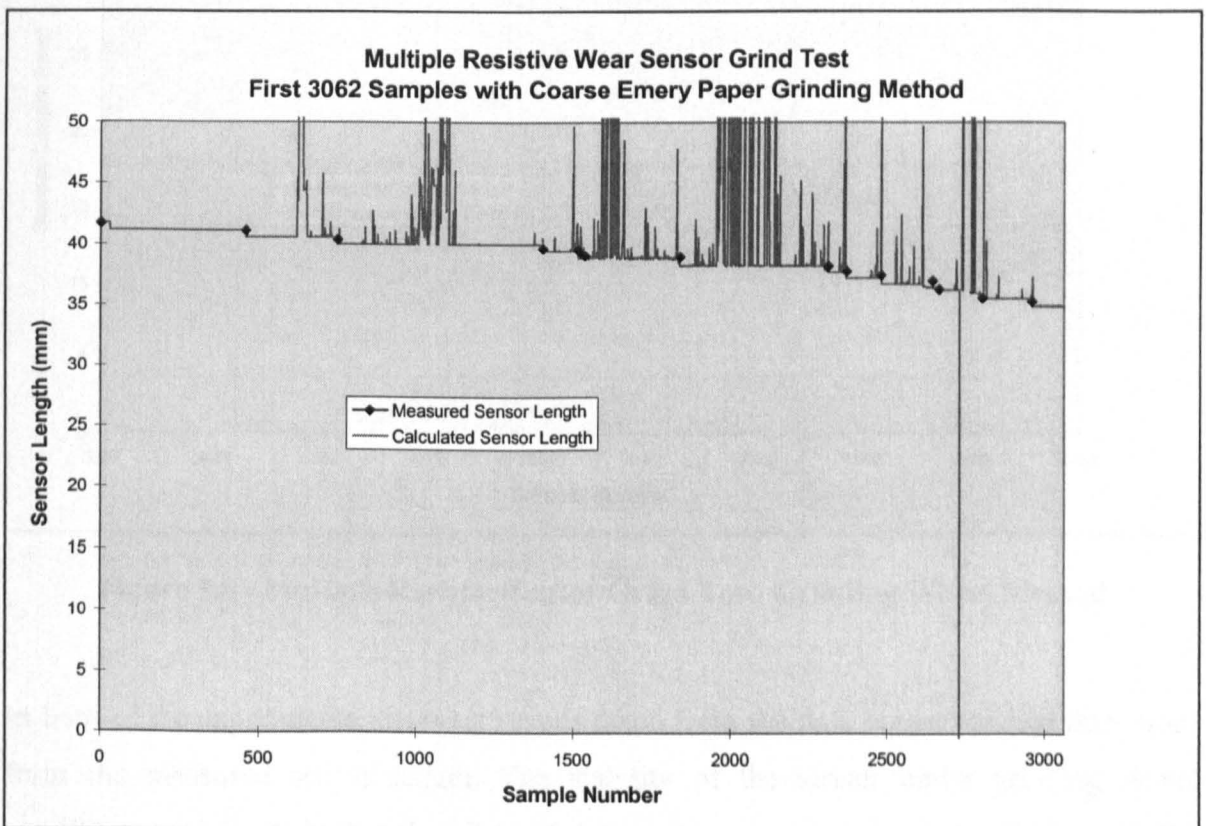
Figure 5.5 - Multiple Resistive Sensor Grind Test, Emery Paper Grinding Method

The results show that the sensor is accurate but unstable whilst under slow grinding movements on coarse emery paper. A high (above the initial sensor length), or infinite, sensor length is produced by a low data logger voltage. This means that a momentary short circuit across the sensor terminals is present.

The second grinding method was then implemented, in which the sensor is worn away using a fine grinding wheel. The results are shown in Figure 5.6, again with the unbroken line showing transposed lengths taken from information from the data logger, and the single points detailing the measured lengths with the vernier calliper.
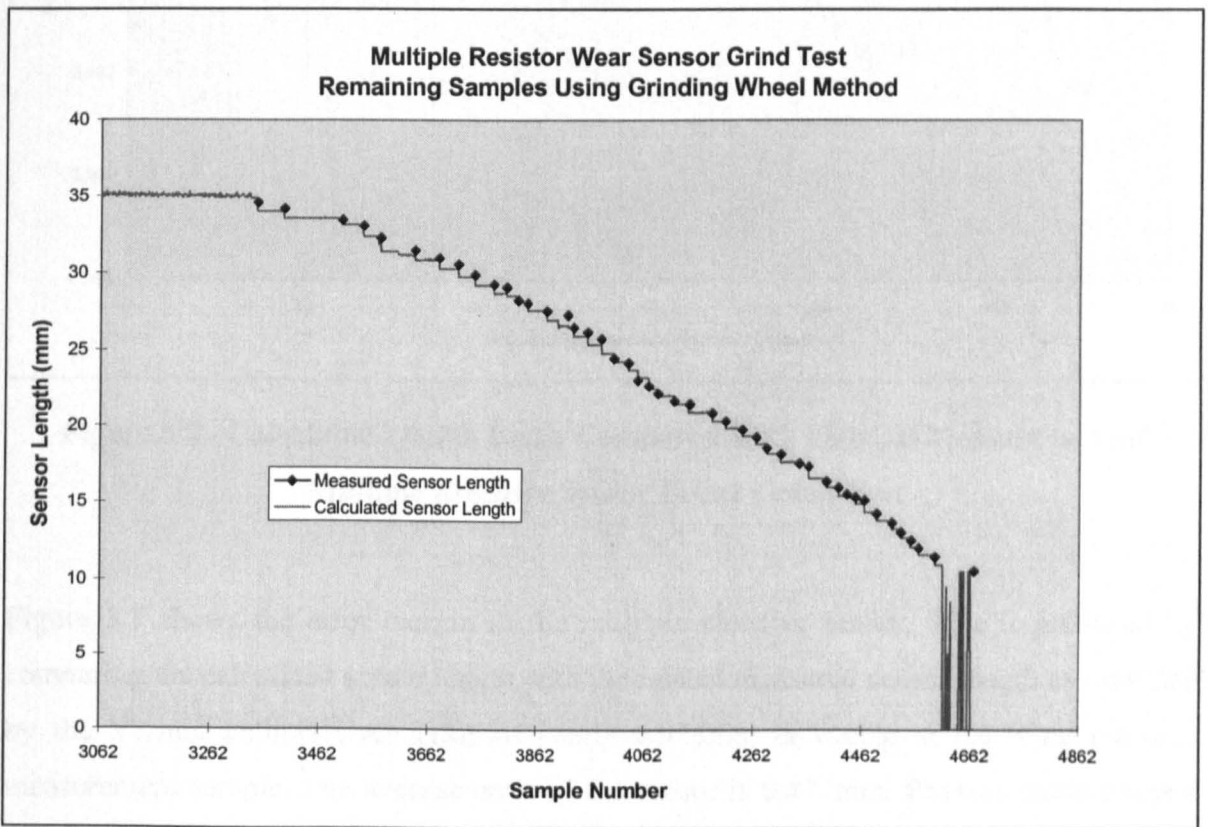
Figure 5.6 - Multiple Resistor Sensor Grind Test, Grinding Wheel Method

In both of the experiments, the wear results taken from the data logger are less than 1mm from the measured sensor length. The stability of the sensor under grinding wheel conditions was much higher than that of the emery paper, and proved reliable until the

sensor reached 12mm in length. At this point, momentary open circuits appeared, thus giving results of 0mm.

Calculated Length Error Compared With Physical Measurement of Multiple Resistive Sensor Under Grind Test
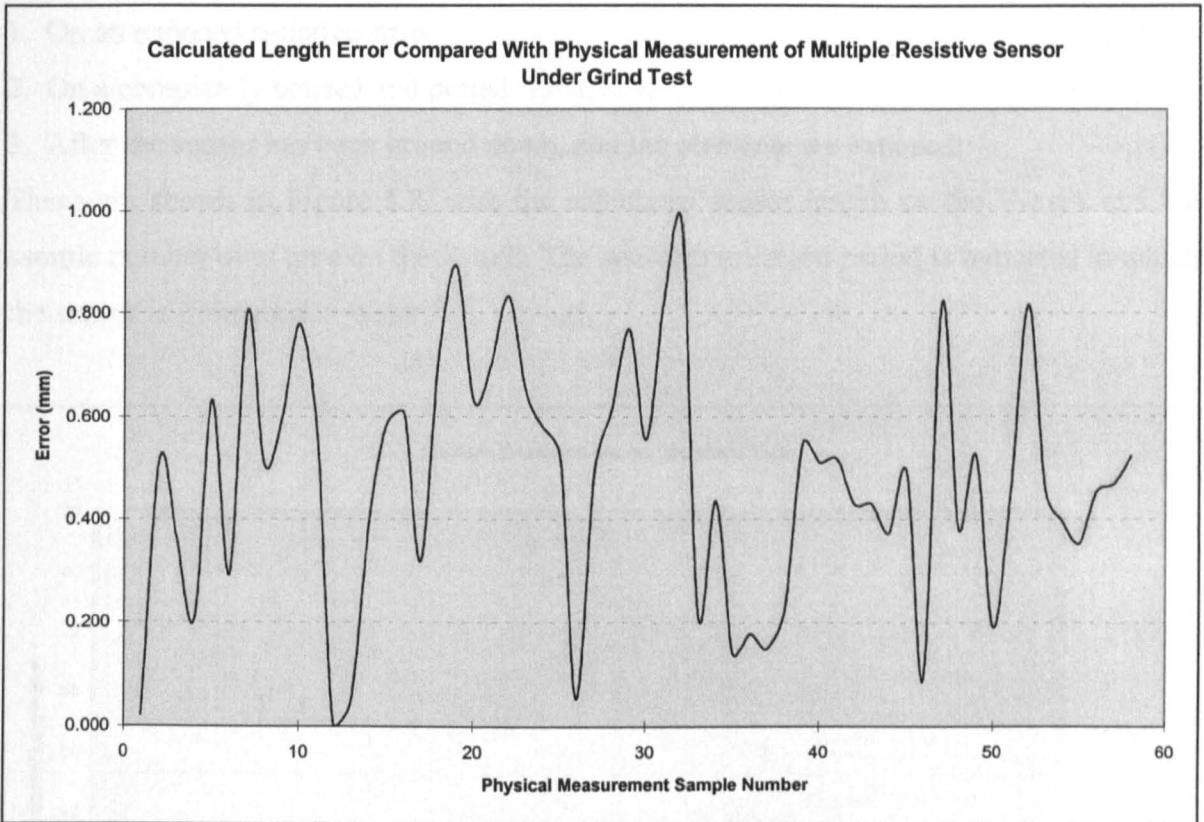
Figure 5.7 - Calculated Length Error Compared With Physical Measurement of Multiple Resistive Sensor Under Grind Test

Figure 5.7 shows the error margin of the multiple resistive sensor. This is produced by comparing the calculated sensor length with the related measured sensor length as provided by the Vernier callipers. An error of nearly 0.978mm is visible at the 32nd physical measurement sample. The average error of the sensor is 0.477mm. Physical measurement samples 1 to 15 are taken whilst the sensor is ground away using the emery paper, and 16 to 58 on the grinding wheel.

5.1.5 Moisture Test

A laboratory moisture test was performed on the multiple resistive sensor at three different stages of the sensor development.

1. On an exposed resistive strip.

2. On a completely housed and potted resistive strip.

3. After the sensor has been ground down, and the elements are exposed.

These are shown in Figure 5.8, with the calculated sensor length on the Y-axis, and the sample number over time on the X-axis. The sensor immersion period is indicated in which the sensor is immersed in water.
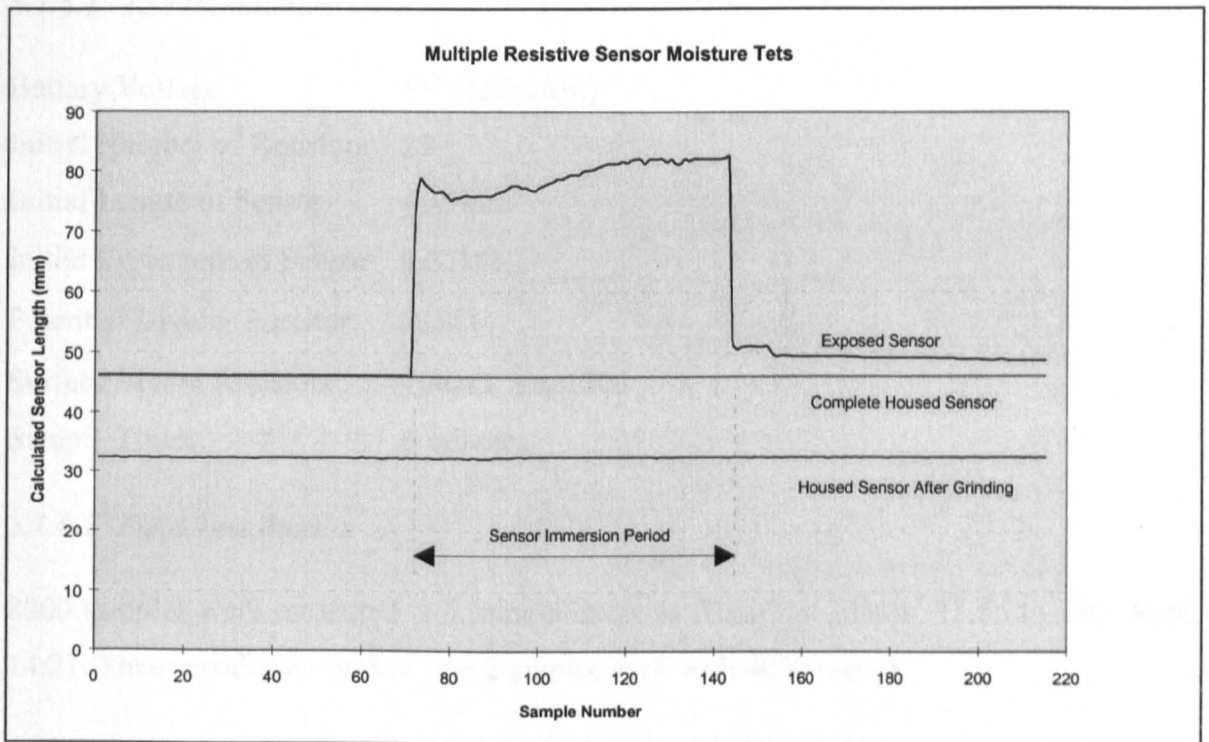
Figure 5.8 - Multiple Resistive Sensor Moisture Test

From the graph, it is noticeable that the exposed sensor reacted adversely during the immersion period, and that a larger sensor length is calculated. This, however, was not the case for the housed sensors. During the immersion period, the resistance of the sensor remains constant. This laboratory test proves that the resistive sensor will not be adversely effected by moisture when installed in a cone crusher.

5.1.6 Field Test

Three Pegson Autocones are used at Shardlow Quarry to grind the Quartzite gravel. The housed sensor was placed near the nip angle of the outer concave crushing member of the Autocone 900 XC. The width of an outer liner for an Autocone 900 XC (Extra Coarse) is 59mm. A data logger with 0 to 2V inputs was installed. To allow for an appropriate signal to be fed into the data logger, a potential divider circuit with related power was necessary. The harsh environment of the crushing part of the quarry meant that no possible power supply was available, thus a long life battery was used. A second channel on the data logger was used to measure the state of the battery during the field test.

5.1.6.1 Test Parameters

Battery Voltage:	9V (550mAhr)
Initial Number of Resistors:	82
Initial Length of Sensor:	40.5mm
Initial Resistance of Sensor:	1.37kΩ
Potential Divider Resistor:	22kΩ
Surface Mount Resistors:	100kΩ, Size1206
Sample Time:	5 minutes

5.1.6.2 Field Test Results

8000 samples were measured at 5 minute intervals from 21st March, 15:56 to 18th April, 14:21. These results are broken into 2 graphs, each with 4000 results.

Multiple Resistive Sensor - Field Test 1-4000
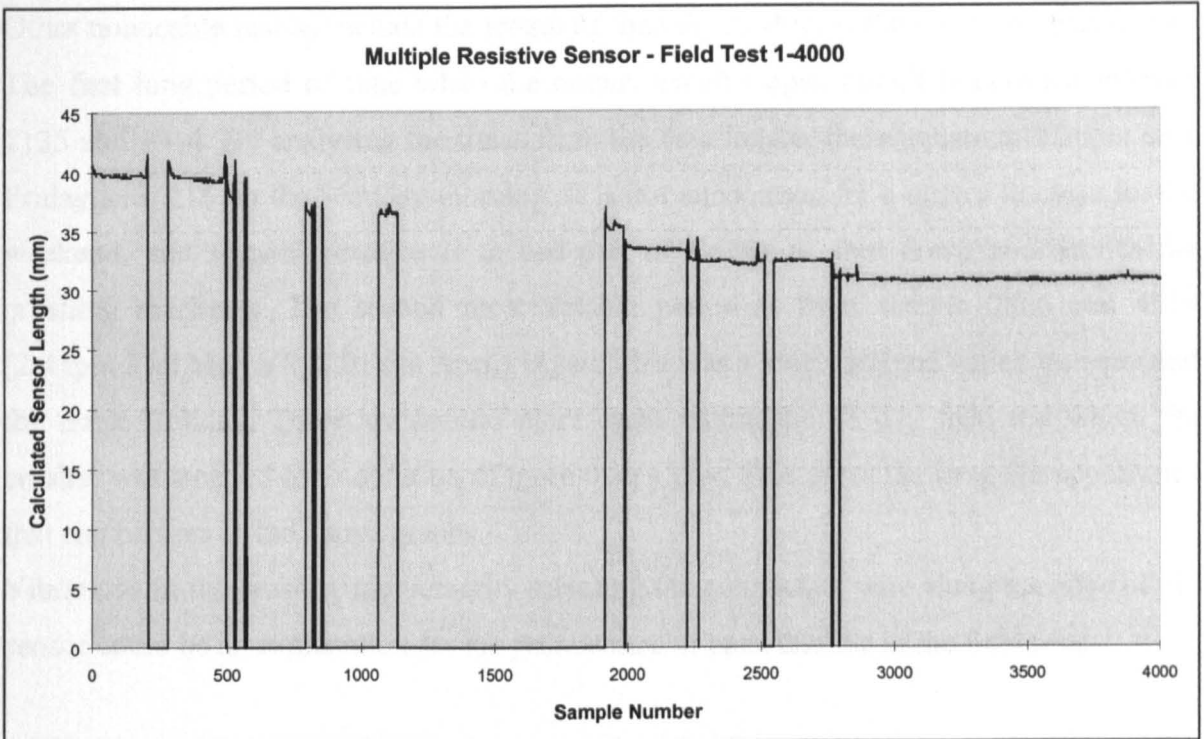
Figure 5.9 - Results 1-4000 Multiple Resistive Sensor Field Test

Multiple Resistive Sensor - Field Test 4001-8000
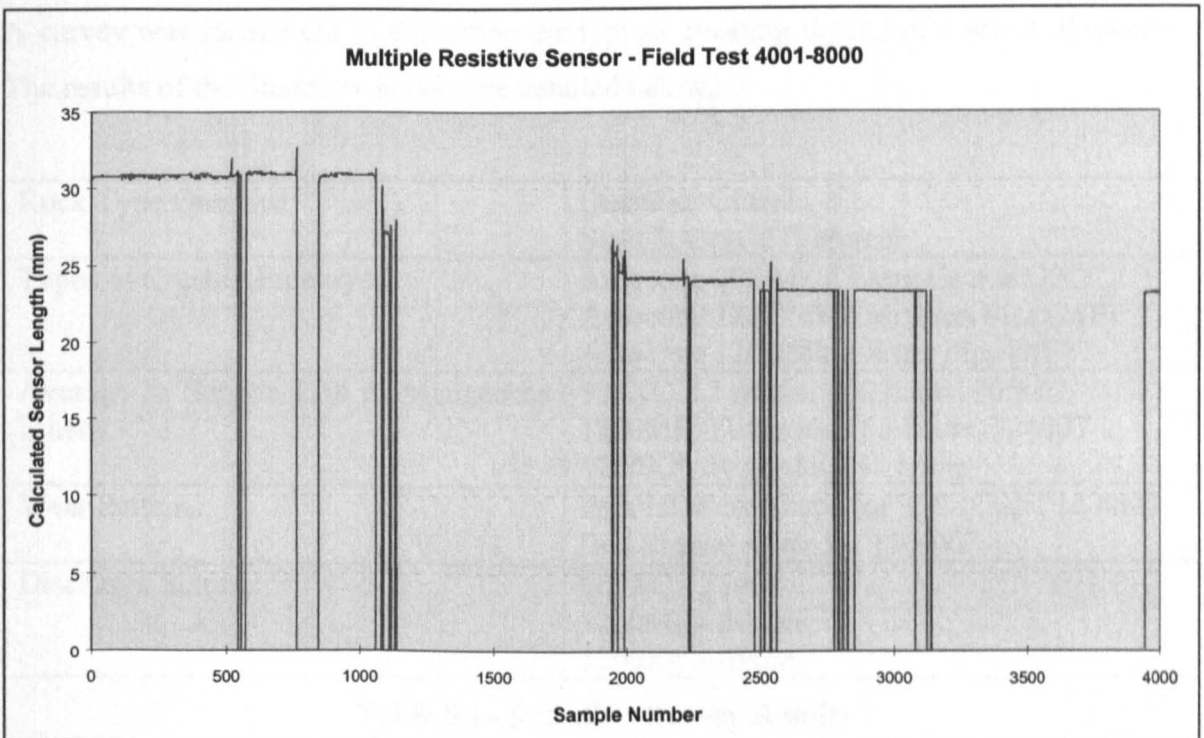
Figure 5.10 - Results 4001 - 8000 Multiple Resistive Sensor Field Test

What is most notable about the field tests, is the amount of open circuits (a length of 0mm). Other noticeable results include the length of time the reading is the same, or open circuit. The first long period of time when the output remains open circuit is between samples 1135 and 1914. By analysing the times from the data logger, these equate to 2:26pm on a Friday and 7:16 on the Monday morning. It is not uncommon for a quarry to close for the weekend, and general practice is to use part of Friday to shut down and service the crushing machines. The second most notable period is from sample 2866 and 4514 (2:41pm 31st March to 8:01 6th April). Again, this was a long weekend which incorporated the Bank Holiday. There are several more cases within the 28 day field test where the crusher was stopped for a duration of more than 1 day. This gives the irregular appearance that can be seen in the above graphs.

Vibrations in the crusher, momentarily releasing the connecting wire along the edge of the sensor could be an explanation for the prominence of open circuits in the field tests.

5.1.6.3 Shardlow Survey Results

A survey was carried out to determine the typical crushing regime of a series of quarries. The results of the Shardlow survey are detailed below.

Rock Type Quarried:	Quartzite Gravel Sand & Gravel (Terrace)
Types of Crusher Employed:	Autocone 900 Mk 2 Extra Coarse (XC) Autocone 1200 Mk 2 Medium Fine (MF) Autocone 1200 Mk 2 Extra Fine (XF)
Average In Service Life of Manganese Liners:	900XC, 12 weeks, 100 hours, 10500T 1200MF, 11 weeks, 185 hours, 32400T 1200XF, 24 weeks, 840 hours
Wear Patterns:	Parallel Wear Shape for 900XC and 1200MF Bell Shaped Wear for 1200XF
Discharge Setting:	900XC - 25mm 1200MF - 20mm 1200XF - 10mm

Table 5.1 - Shardlow Survey Results

5.1.6.4 Result Verification

8000 samples were taken at 5 minute intervals which lasted from late afternoon on the 21st March to 18th April (28 days). During this time, the sensor was worn down from 40.50mm to 23.36mm (17.14mm worn away). The wear sensor was installed in the 900 Autocone Extra Coarse (XC) crusher which is used approximately 100 hours in a 12 week period, thus an average of 1.19 crushing hours per day. During the 28 days the sensor was installed, a total of 33.32 hours of crushing took place. Relating this figure to the amount of sensor worn away, a wear rate of 0.51mm per hour is concluded. From the survey, a wear rate on the manganese can reach up to 0.75mm per hour. However, typical figures averaged over the life cycle of the manganese liner means that a figure closer to 0.52mm per hour is more likely (100 hours, 52mm wear depth). These values are estimates, with figures taken from questionnaires. It does, however, give similar results for the expected wear.

5.2 Single Resistive Element Sensor

A custom made 50.5mm single resistive strip was developed as mentioned in Chapter 4, with extensive laboratory and field tests carried out on the sensor.

5.2.1 Compressive Strength Test

An identical compressive strength test to that of the multiple resistive element sensor was implemented using similar housings to those detailed in Figure 5.1. By measuring the resistance of the sensor (180Ω), a resistor of similar value put into a potential divider circuit with the sensor, gives an output of 2.5V for a perfectly working sensor, and thus a calculated sensor length of 50.5mm. If the output drops to 0V, then the sensor is closed circuit. If the output is 5V, then the sensor is open circuit.
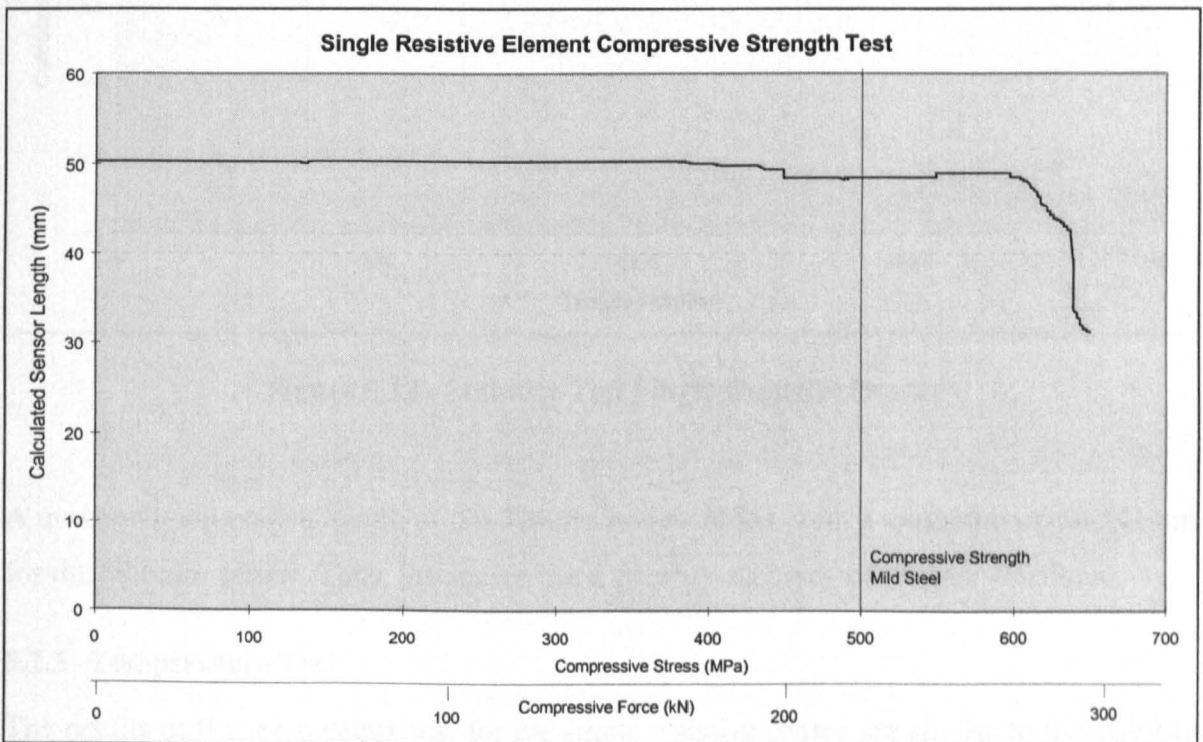
Figure 5.11 - Single Resistive Sensor Compressive Strength Test

Forces up to 200kN were provided by the Instrom, and forces greater than 200kN by the less accurate Denison. The sensor is steady up until a compressive stress of 430MPa. After this, the calculated sensor length decreases by almost 2mm. This is probably due to an early deformation of the mild steel housing.

5.2.2 Stability Test

Figure 5.12 shows the results for the stability test of the single resistive sensor.
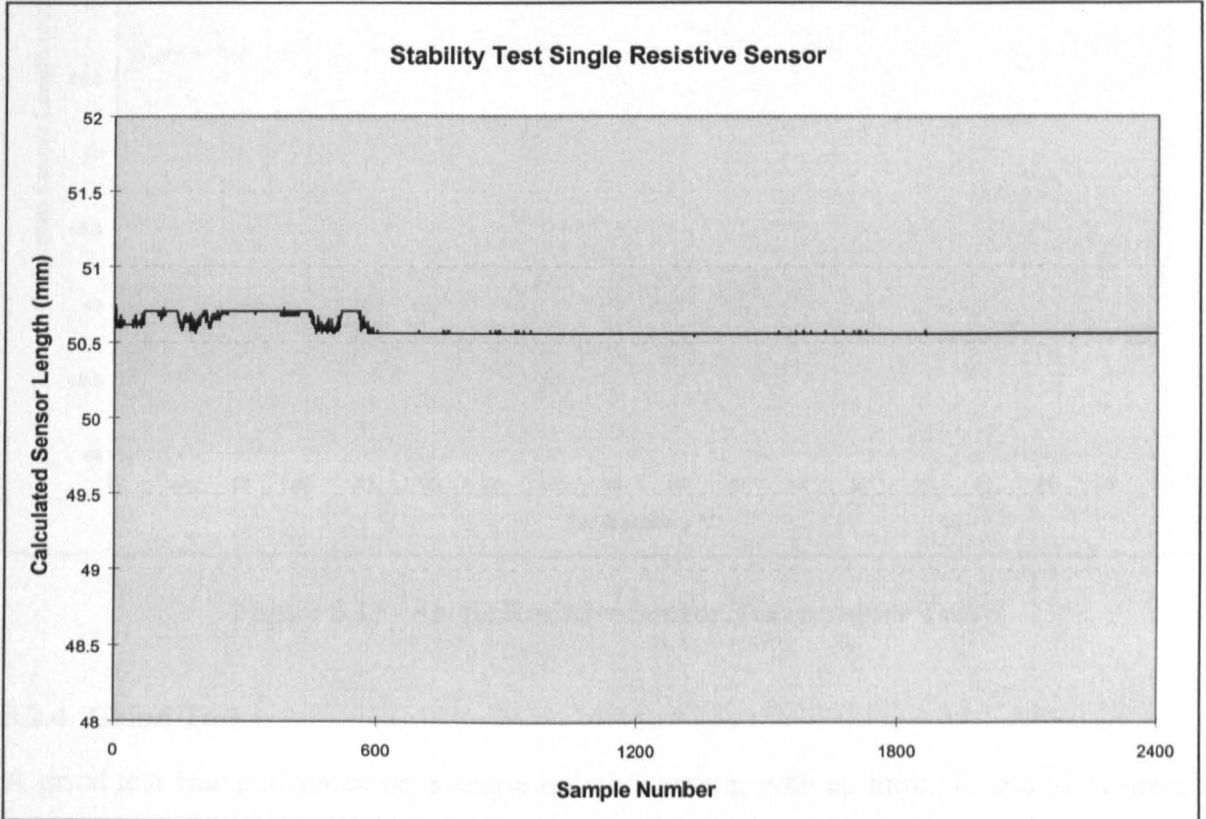
Figure 5.12 - Stability Test Single Resistive Sensor

A maximum equivalent length of 50.746mm was recorded, with a minimum of 50.541mm for the 50.6mm sensor. Thus, the sensor has a stability accuracy of +0.146/-0.059mm.

5.2.3 Temperature Test

The results of the temperature test for the single resistive sensor are similar to the stability test, i.e. the sensor did not change dramatically in resistance over the temperature range. Figure 5.13 shows the results of the temperature test, with a maximum of 50.850mm at 69.5°C and a minimum of 50.538mm at 19.7°C (+0.350mm). A gradual increase in the equivalent length is noticed at temperatures near 70°C.
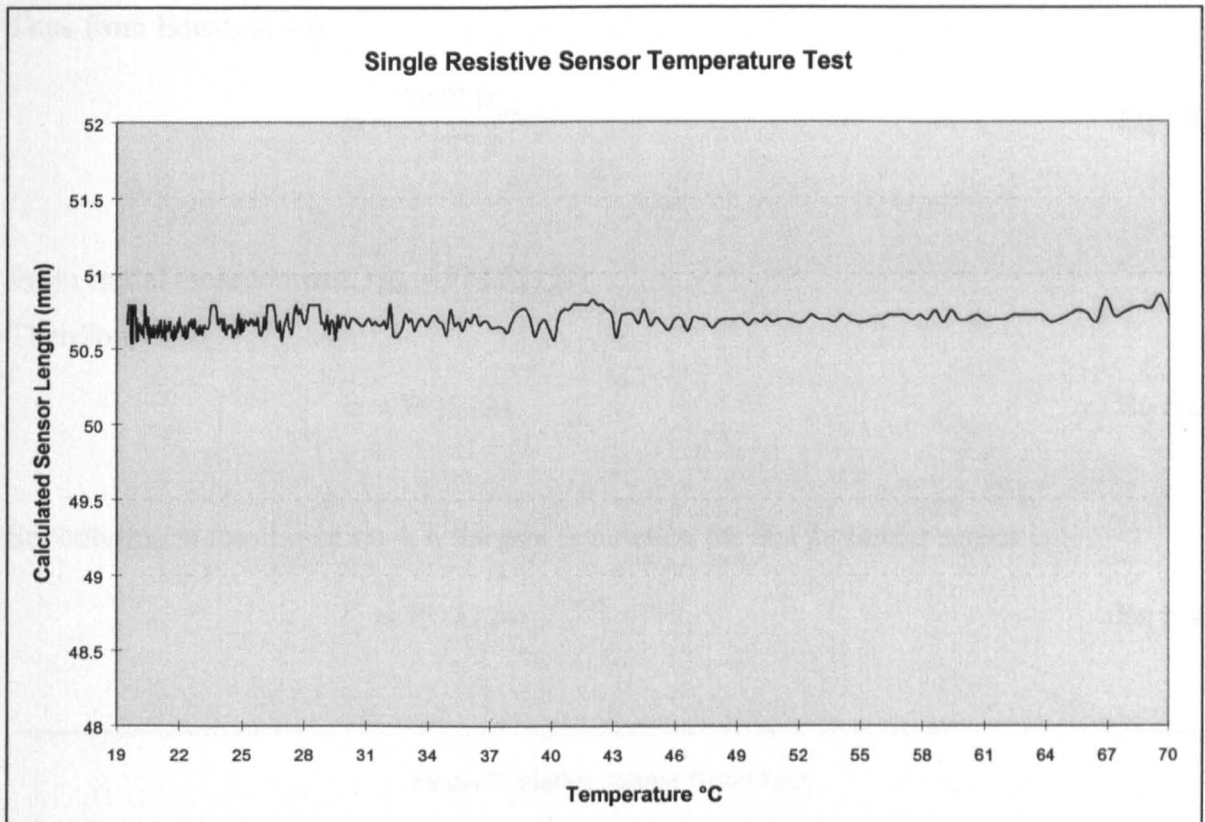
Figure 5.13 - Single Resistive Sensor Temperature Test

5.2.4 Grind Test

A grind test was performed on a single resistive sensor, with an initial length of 50.6mm. The sensor is potted using epoxy resin in a plastic housing of diameter 8mm. Wire wrap wire is soldered down the entire length of each side of the sensor, and protrudes out of the back of the plastic housing to feed into the potential divider circuit. A 1kΩ resistor is used in the potential divider circuit to produce an acceptable output voltage within the data logger range.

The grinding wheel was used to degrade the sensor. The results are shown in Figure 5.14, with the unbroken line showing transposed lengths taken from information from the data logger, and the single points detailing the measured lengths with the vernier calliper.

The length is determined by initially calculating the resistance of the sensor. The start resistance is used to calculate the multiplication factor within the power function.

Thus from Equation 4.6,

$$m = \frac{7279.1 r_{init}}{172.3}$$

Eq 5.2

From initial measurement, $r_{init} = 178.3115\Omega$.

Therefore,

$$m = 7533.124$$

Eq 5.3

Substituting m into Equation 4.7, the power function for this particular sensor is :-

$$l_w = 7533.124 r_w^{-0.9659}$$

Eq 5.4
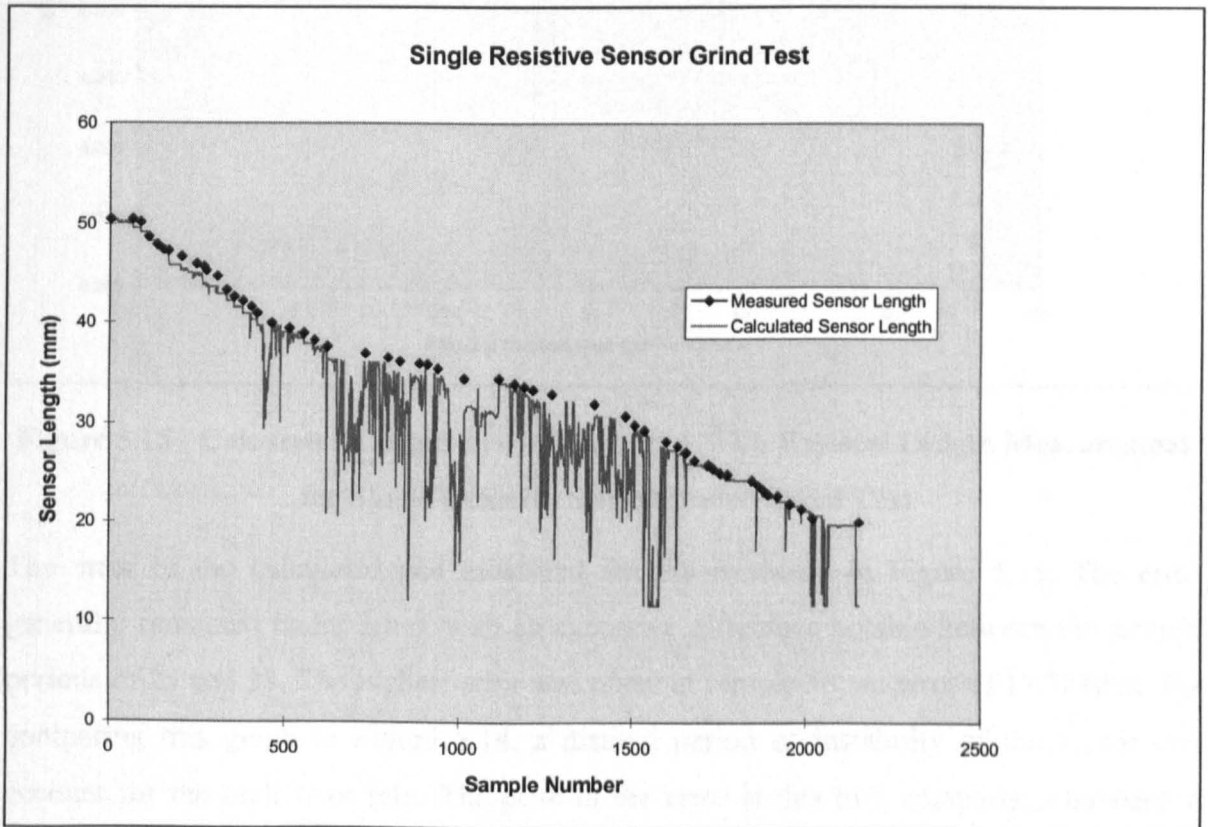
Figure 5.14 - Single Resistive Wear Sensor Grind Test, Grinding Wheel Method

The results show that the sensor output generally followed the measured length. An accurate wear pattern emerged for the first 10mm wear on the sensor. After that, the results

became erratic with a sudden length differentiation of up to 17mm. Due to the brittle structure of the single resistive sensors, it is possible that during the grinding test, the sensor 'cracked', and that a wear pattern perpendicular to the surface of the sensor was only present for the first 10mm of wear.
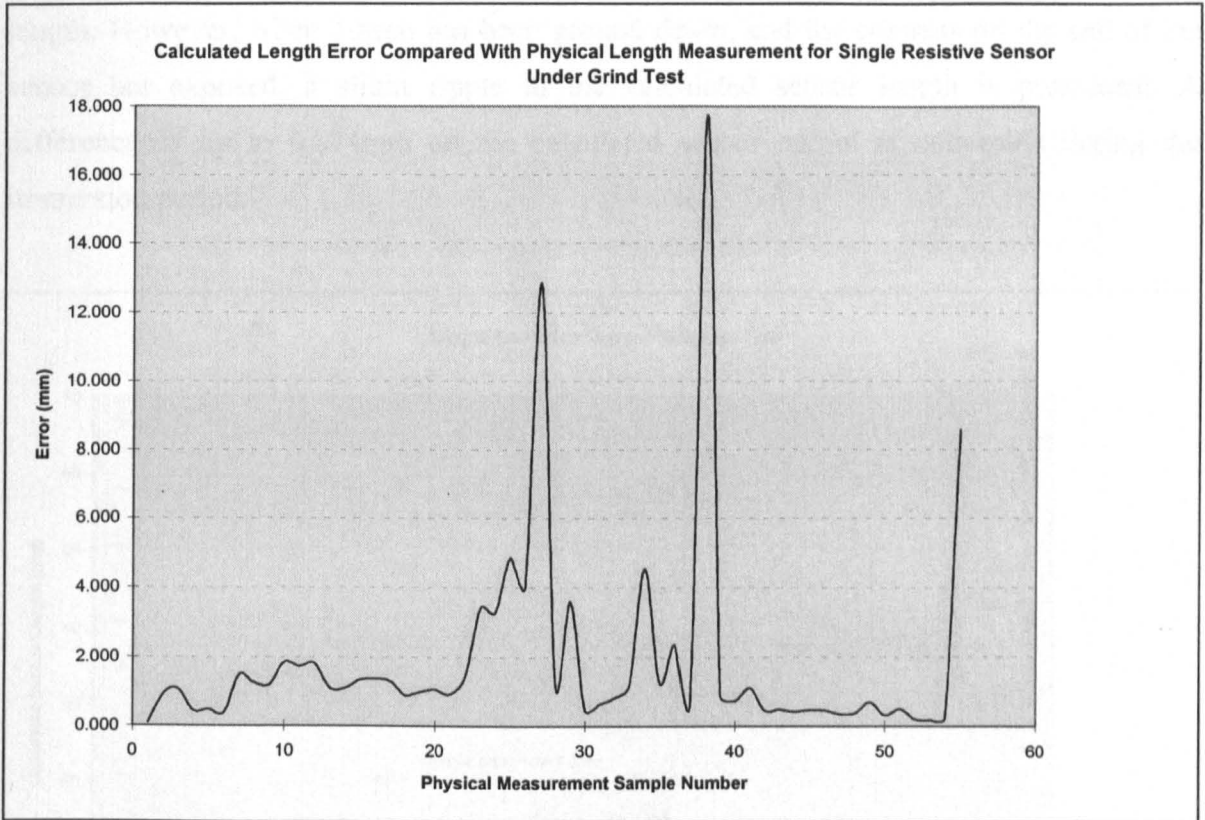
Figure 5.15 - Calculated Length Error Compared With Physical Length Measurement for Single Resistive Sensor Under Grind Test

The error of the calculated and measured lengths is shown in Figure 5.15. The error generally remained under 2mm, with an excessive difference notable between the sample periods of 23 and 38. The highest error was noted at sample 38, an error of 17.721mm. By comparing this graph to Figure 5.14, a distinct period of instability of the sensor can account for the high error rate. The peak in the error is due to a comparison between a physical measurement of 29.3mm and a temporary calculated 'spike' of 11.57mm. An average error of 1.891mm is calculated including the temporary 'spikes'. By examining the first 10mm of wear on the sensor (Figure 5.14), it can be seen that the sensor is stable. This gives an average error of 1.122mm.

5.2.5 Moisture Test

The single resistive sensor was subjected to a laboratory moisture test in which the sensor is exposed, completely housed and housed with 10mm ground down. The results are shown in Figure 5.16. When the exposed sensor is subjected to the water, an increase in the calculated sensor length is noticed. This is not the case for the complete housed sensor length. However, when 10mm has been ground down, and the contacts on the end of the sensor are exposed, a slight ripple in the calculated sensor length is prominent. A difference of up to 0.874mm on the calculated sensor output is noticeable during the immersion period.
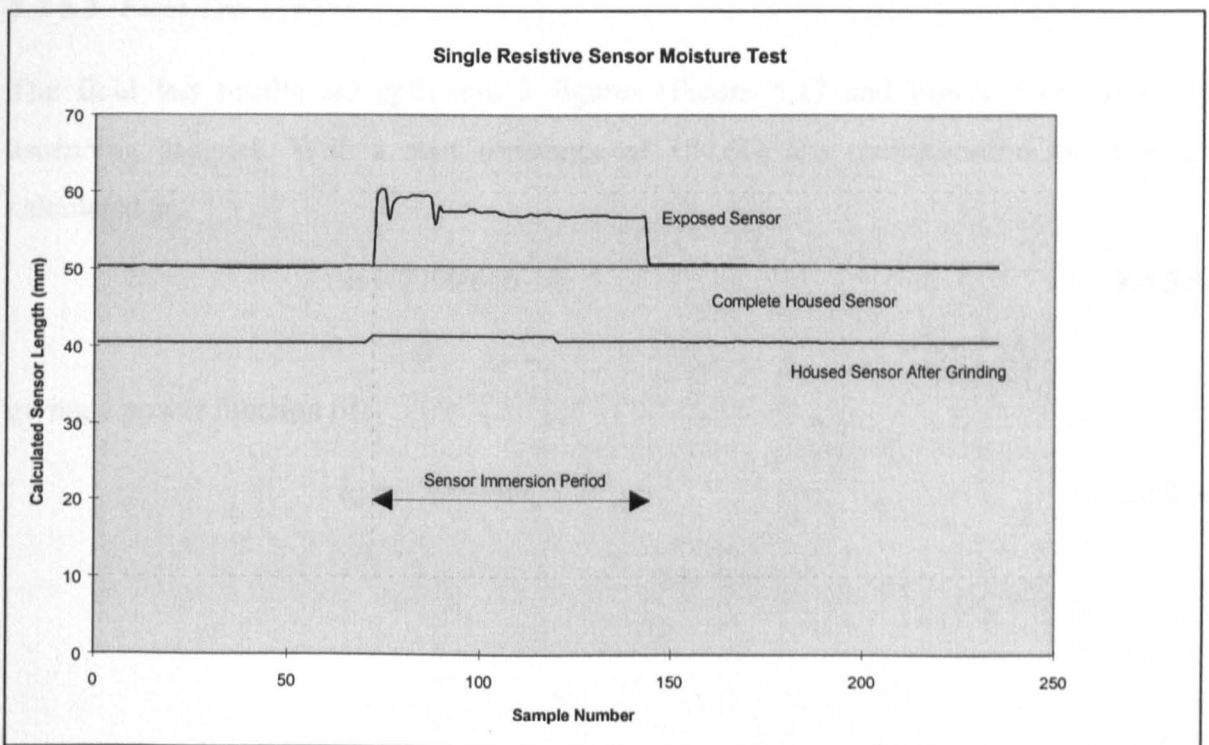
Figure 5.16 - Single Resistive Sensor Moisture Test

This laboratory test shows that the accuracy of the housed sensor would be altered if used in an environment in which moisture is a factor.

5.2.6 Field Test

A similar set up was employed involving the data logger, a potential divider circuit, and a housed sensor. The sensor was placed in a cone crusher at Shardlow quarry, with the sensor positioned towards the nip angle in the outer liner (concave). The attached wires were fed

through the cone crusher to a 0 to 2 volt data logger, with a second channel used to measure the battery voltage. Samples were taken every 5 minutes.

5.2.6.1 Test Parameters

Battery Voltage:	9V (550mAhr)
Initial Length of Sensor:	50.5mm
Initial Resistance of Sensor:	184.6Ω
Potential Divider Resistor:	3.9kΩ
Sample Time:	5 minutes

5.2.6.2 Field Test Results

The field test results are split into 2 figures (Figure 5.17 and Figure 5.18) of 4000 ascending samples. With a start resistance of 184.6Ω, the multiplication factor m is calculated as,

$$m = 7799.446 \qquad \textbf{Eq 5.5}$$

giving a power function of :-

$$l_w = 7799.446 r_w^{-0.9659} \qquad \textbf{Eq 5.6}$$
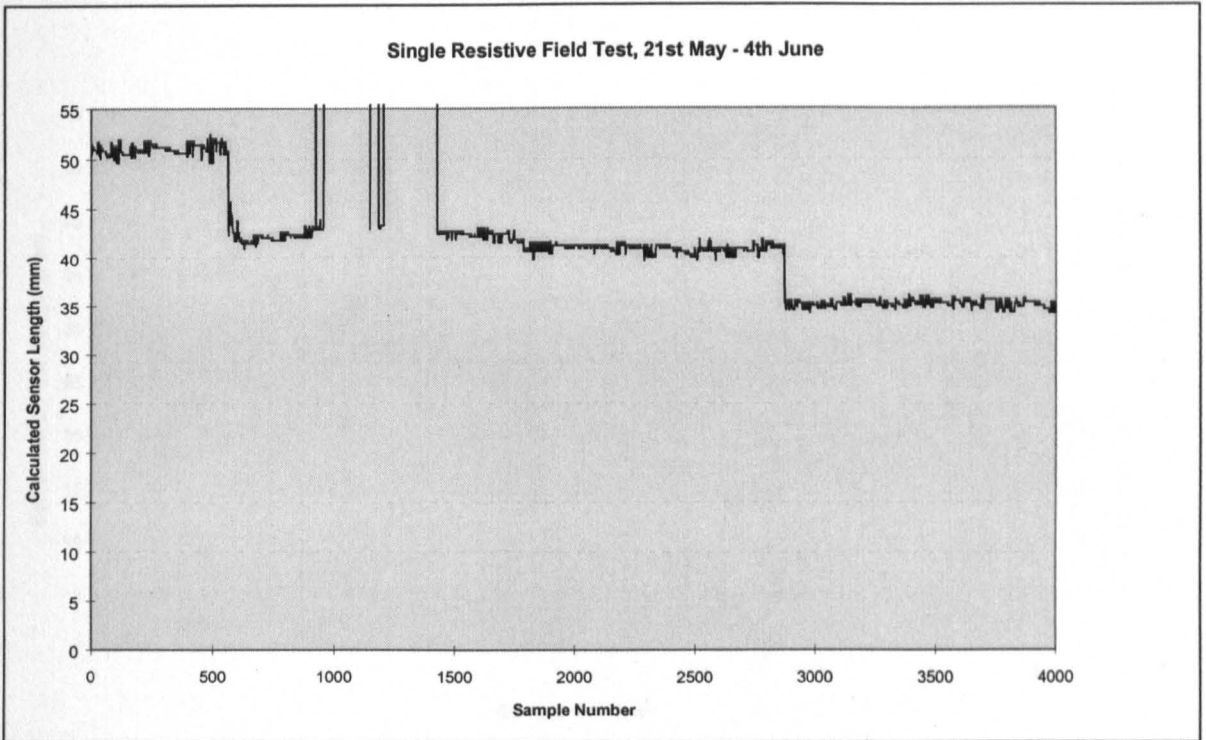
Figure 5.17 - Single Resistive Sensor Field Test, 21st May - 4th June

Figure 5.17 shows the first 4000 samples of the sensor output, from 10:35 on the 21st May to 7:50 on the 4th June. Most noticeable about these results is the distinct descending 'steps' in the output. This is either due to excessive wear on the outer liner in the position of the sensor, or the sensor breaking away at a point further down the sensor than the true wear of the manganese steel. There are explanations which support the theory of both of these occurrences. In the case of the excessive liner wear, the introduction of an insert into the liner could produce a weak area in the manganese steel and thus become a target for the rotating rock. However, it is more likely that due to the brittle nature of the sensor and the epoxy resin not supporting the sensor efficiently, that a single break across the sensor or a crack down the middle caused the stepped output.

WEAR SENSOR RESULTS
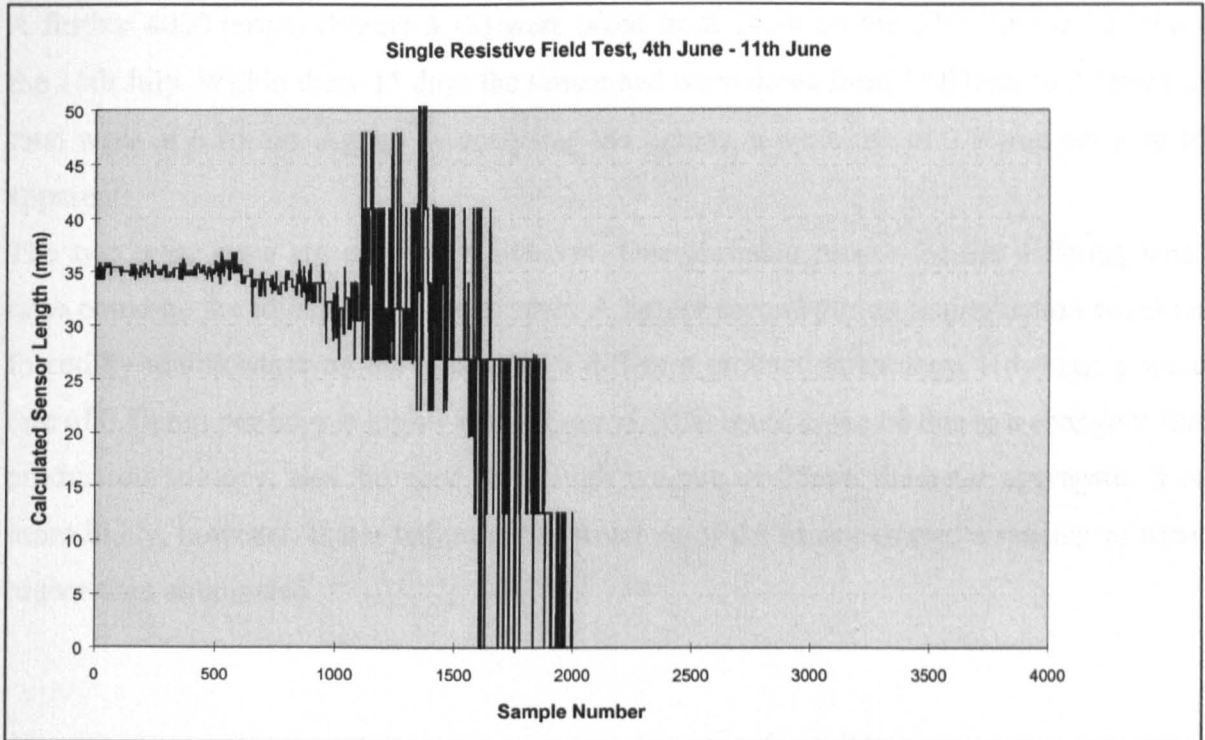
Figure 5.18 - Single Resistive Sensor Field Test, 27 June - 11th July

Also noticeable within both of the graphs is a short period in which the sensor gives a reading of greater than 50mm. A short circuit across the sensor is the most probable cause of this, as the sensor is only 3mm across the contacts. Figure 5.18 shows a further 4000 samples from 14:24 on June 27th to 11:39 on July 11th. During this period a further series of 2-3mm steps are visible, with a couple of momentary short circuits.

5.2.6.3 Result Verification

The first 4000 results (Figure 5.17) were taken from 10:35 on the 21st May to 7:50 on the 4th June. During these 15 days, the sensor had worn down from 50.47mm to 34.70mm (15.77mm worn away). Using the typical daily crushing rate of 1.19 taken from the survey, a total of 17.85 hours crushing was performed in these 15 days. Equating this figure with the data logger results gives a wear rate of 0.88mm per hour.

A further 4000 results (Figure 5.18) were taken from 14:24 on the 27th June to 11:39 on the 11th July. Within these 15 days the sensor had worn down from 14.07mm to 7.31mm, a total wear of 6.76mm. Again, by analysing the figures, a wear rate of 0.38mm per hour is apparent.

The two wear rates are extremely different. One probable reason for the differing wear rates could be the utilisation of the crusher. A lighter second period in production could be forced by maintenance on the crusher or a different production strategy. However, a wear rate of 0.88mm per hour is higher than expected. This could again be due to a change in the production strategy, and the need for a high volume of 25mm diameter aggregate. It is more likely, however, that a failure in the structure of the sensor caused a reading of wear higher than anticipated.

5.3 Multiple Surface Mount Capacitive Sensor

A similar structure to that of the multiple surface mount resistive sensor is employed, in which a series of surface mount capacitors are stacked up on top of each other. The resolution is dependent upon the thickness of the surface mount capacitor used, 0.6mm for the 0805 type capacitors. Signal conditioning is more complex with the capacitive sensor, with oscillators the most accurate method for capacitive measurement.

5.3.1 Compressive Strength Test

A compressive strength test was performed on a capacitive element consisting of 82, 0.1μF capacitors, type 0805. The sensor element was contained in a steel insert similar to that in Figure 5.1 using epoxy resin. The length of the sensor is calculated using a 555 timer and NEC 78330 microcontroller as described in Chapter 4. The data is serially downloaded in real-time to a computer which stores the capacitance value of the sensor in a text file.
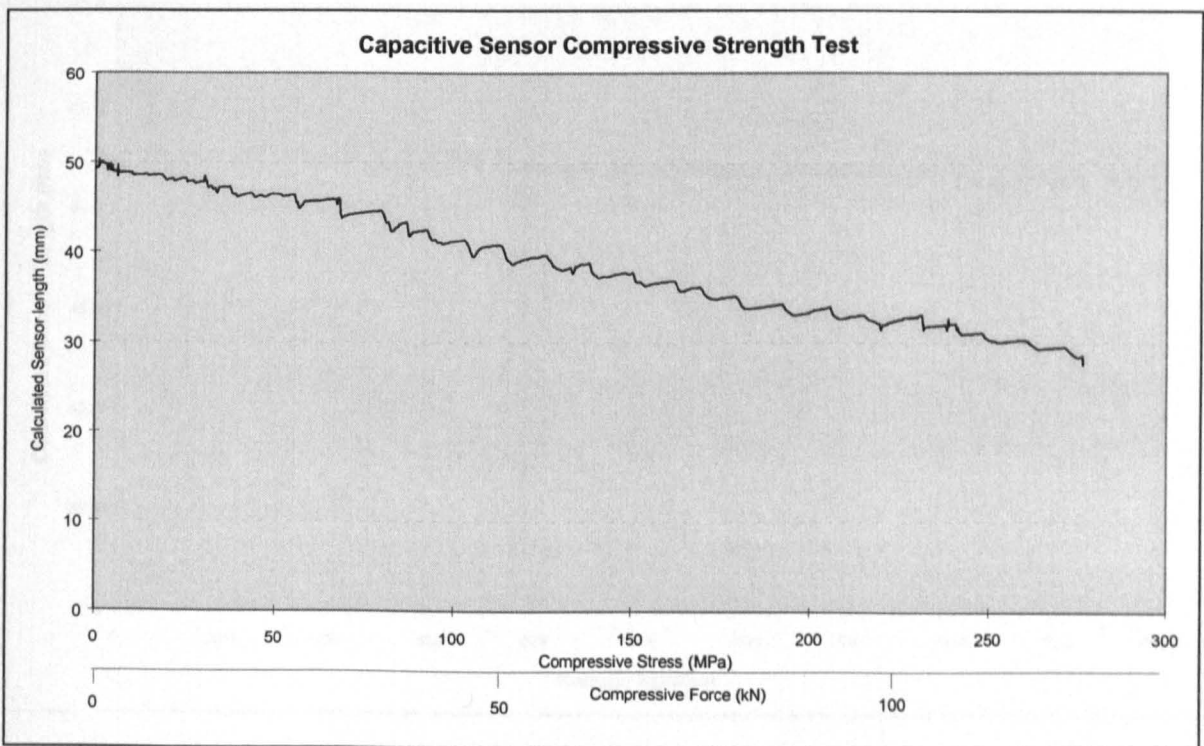
Figure 5.19 - Capacitive Sensor Compressive Strength Test

Figure 5.19 shows the results of this test. From the graph, it is clear that the capacitance value of the sensor changed dramatically due to a compressive force on the sensor, thus

changing the equivalent length of the sensor. This varied from the calculated start length of 49.4mm to 27.65mm at 125kN (Compressive Stress of 278MPa). The change in capacitance is notable after a relatively low compressive force. The poor performance of the sensor could be due to a change in the internal structure of the individual surface mount capacitors.

5.3.2 Stability Test

Due to the nature of capacitive devices, and their intolerance to the surrounding environment, information about the accuracy of the device can be achieved from a stability test (Figure 5.20). The results from the experiment are taken from a capacitive sensor measuring 49.2mm in length comprising of 82, 0.1µF surface mount capacitors type 0805.
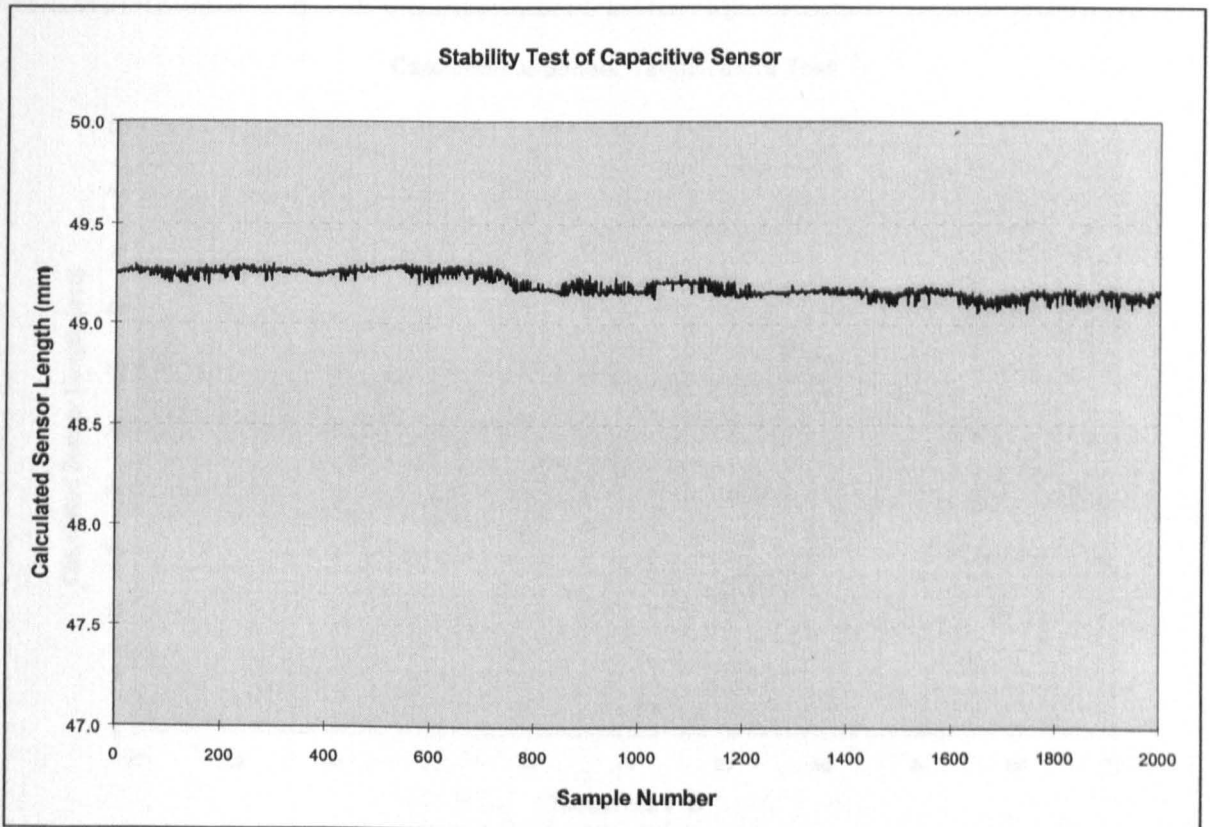
Figure 5.20 - Stability Test of Capacitive Sensor

Two thousand samples at one second intervals are taken which is 33 minutes and 20 seconds. The results use Equation 4.17 to calculate the length of the sensor during operation. From the figure, it can be seen that the sensor is fairly erratic. Maximum and

minimum sensor readings are 49.300mm and 49.065mm respectively. Thus an initial stability accuracy of +0.1/-0.135mm is observed.

5.3.3 Temperature Test

A temperature test on an exposed capacitive stack of 82, 0.1µF surface mount capacitors was performed. The results, shown in Figure 5.21, detail the instability of a capacitive sensor due to temperature. The stack was clamped next to a temperature probe, and a regular heat applied. Capacitance and temperature measurements are regularly taken of the sensor. A temperature range of 22°C to 70°C is applied to the sensor. From the results, the sensor capacitance changes from 7.5µF at 22.3°C to 2.82µF at 70°C. This equates to an equivalent 49.5mm to 18.6mm respectively, thus showing that the temperature of the capacitive elements is an important factor in the sensor design.
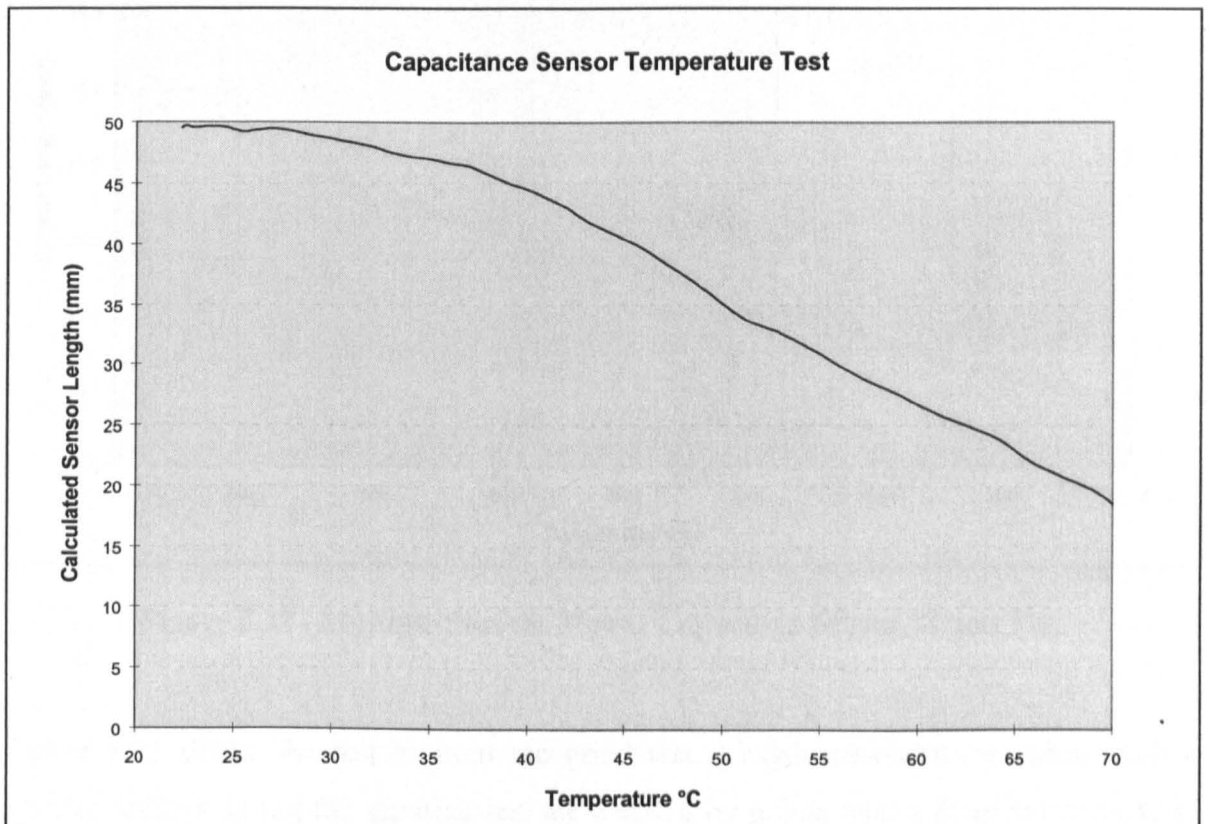
Figure 5.21 - Capacitive Sensor Temperature Test

5.3.4 Grind Test

A grind test was performed on a capacitive stack of 82, 0.1µF surface mount capacitors, size 0805, Z5U dielectric. The sensor is potted using epoxy resin in a plastic housing of diameter 5.7mm. Wire wrap wire is soldered down the entire length of each side of the sensor, and protrudes out of the back of the plastic housing to feed into the signal conditioning circuit.
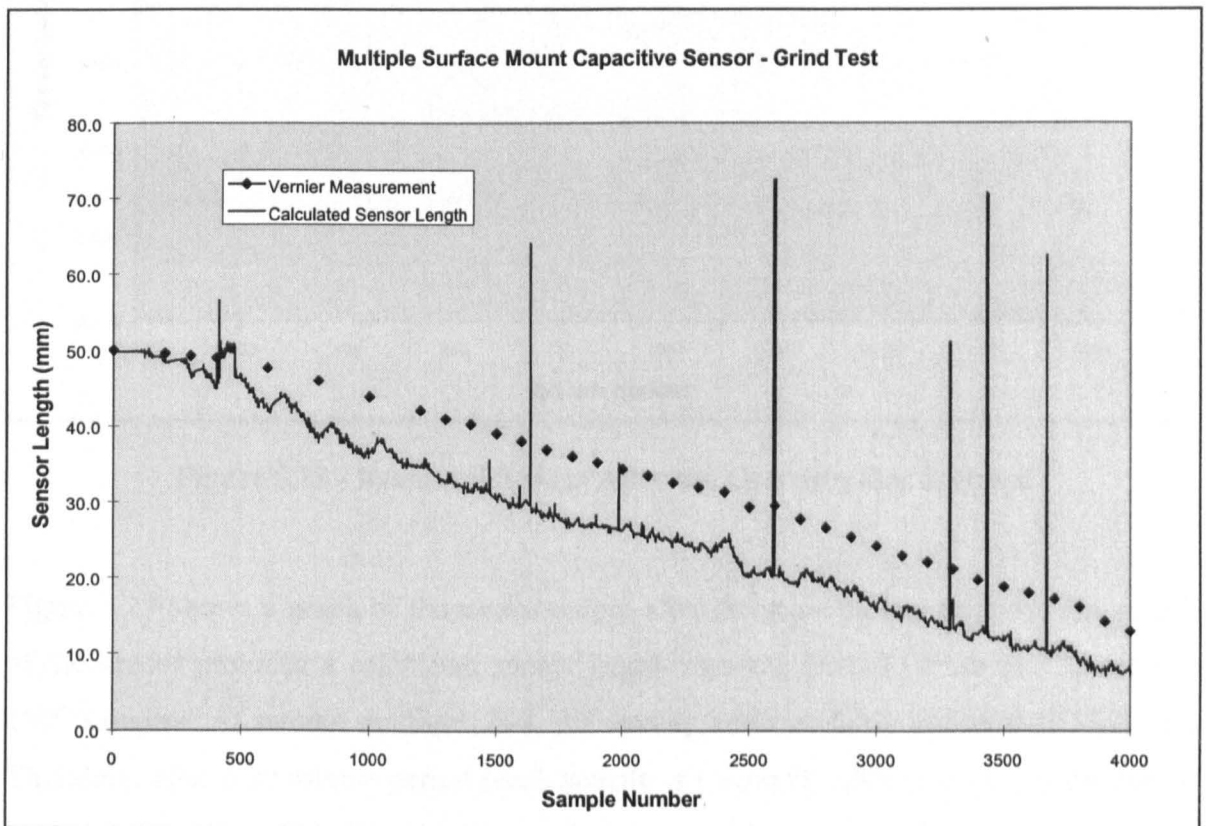
Figure 5.22 - Multiple Surface Mount Capacitive Sensor, Grind Test

Figure 5.22 shows the results from the grind test. Length measurements taken with a vernier calliper during the grinding test are denoted by points with a diamond point style on the graph. Sensor readings are denoted by the unbroken line. What is noticeable from the graph is the offset of the sensor reading and the physical measurement. This becomes effective after an initial 'warming up' period. During the first 250 samples, the capacitors are relatively cold, and thus an accurate reading is output. After this, the temperature of the capacitors rise, the sensor capacitance drops and the equivalent length is reduced.
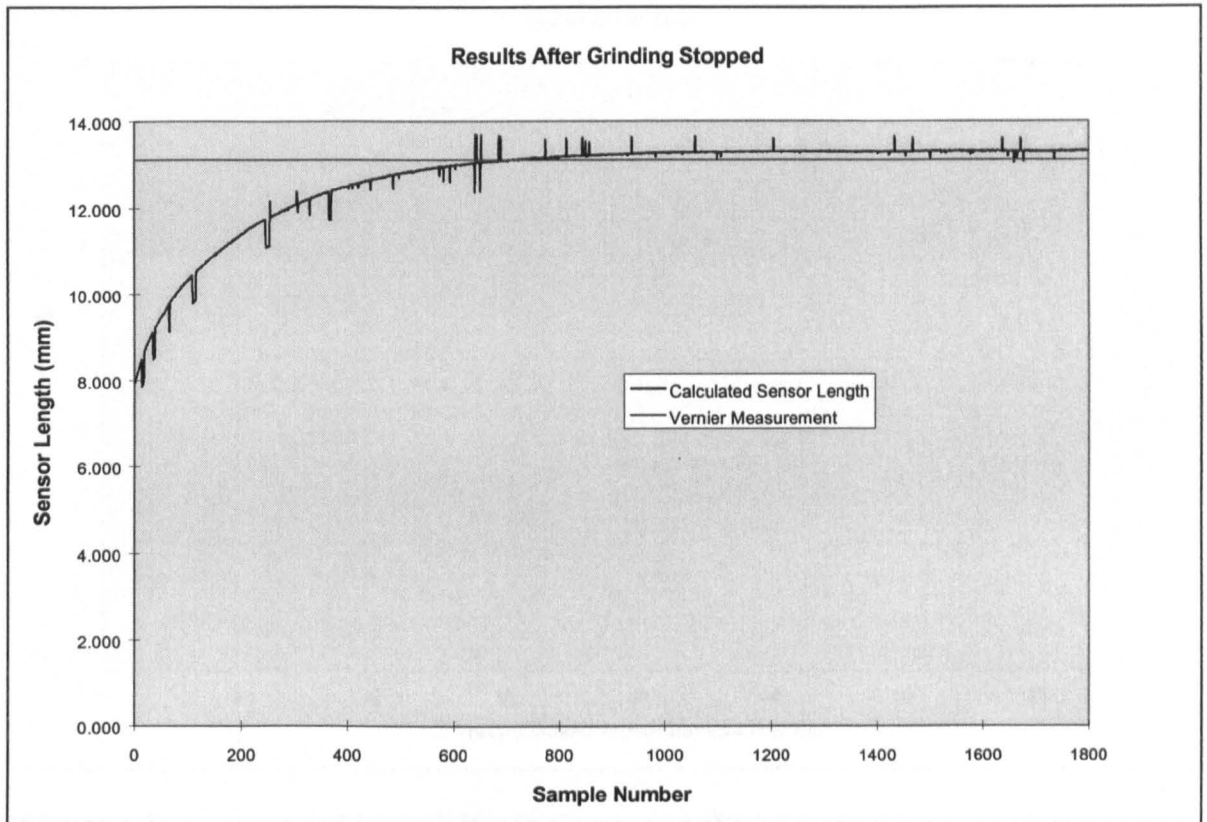
Figure 5.23 - Results of Sensor After the Grinding Has Stopped

Figure 5.23 shows a graph of the sensor output after the grinding has stopped. The output of the sensor provides a calculated sensor length increase from 7.91mm to 13.32mm in 1800 samples. At sample number 1200, the reading remains fairly constant at 13.29mm. Therefore, after a 20 minute period (each sample is 1 second, 1200 samples) of the sensor being removed from grinding conditions, the output settles down to the correct value. The temperature properties of surface mount capacitors with different dielectric are listed in Table 4.2, which shows that devices using Z5U dielectric are susceptible to a +22% to a -56% change over a working temperature of +10°C to +85°C. This is noticeable with this experiment, and a capacitance stack using surface mount capacitors with a lower dielectric would be less susceptible to temperature changes.
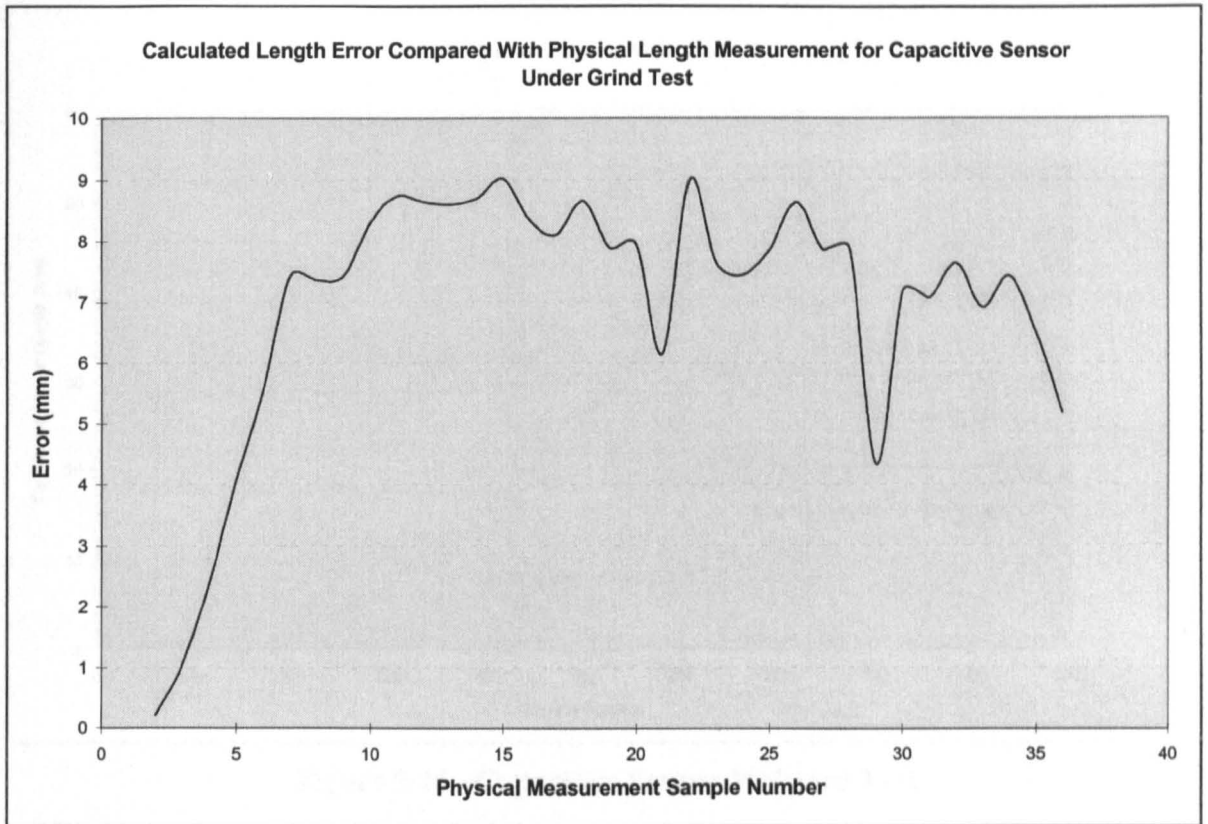
Figure 5.24 - Calculated Length Error Compared With Physical Length Measurement for Capacitive Sensor Under Grind Test

From Figure 5.24, a high error rate is noted. This typically varies between 4mm and 9mm, with and average error of 6.947mm. The rise in error at the start of the experiment is due to the temperature increase in the capacitors.

5.3.5 Moisture Test

A moisture test on the a capacitive strip of surface mount devices produced the results as shown in Figure 5.25. The three data series are for an exposed sensor, a complete housed sensor and a housed sensor in which 10mm has been ground away.
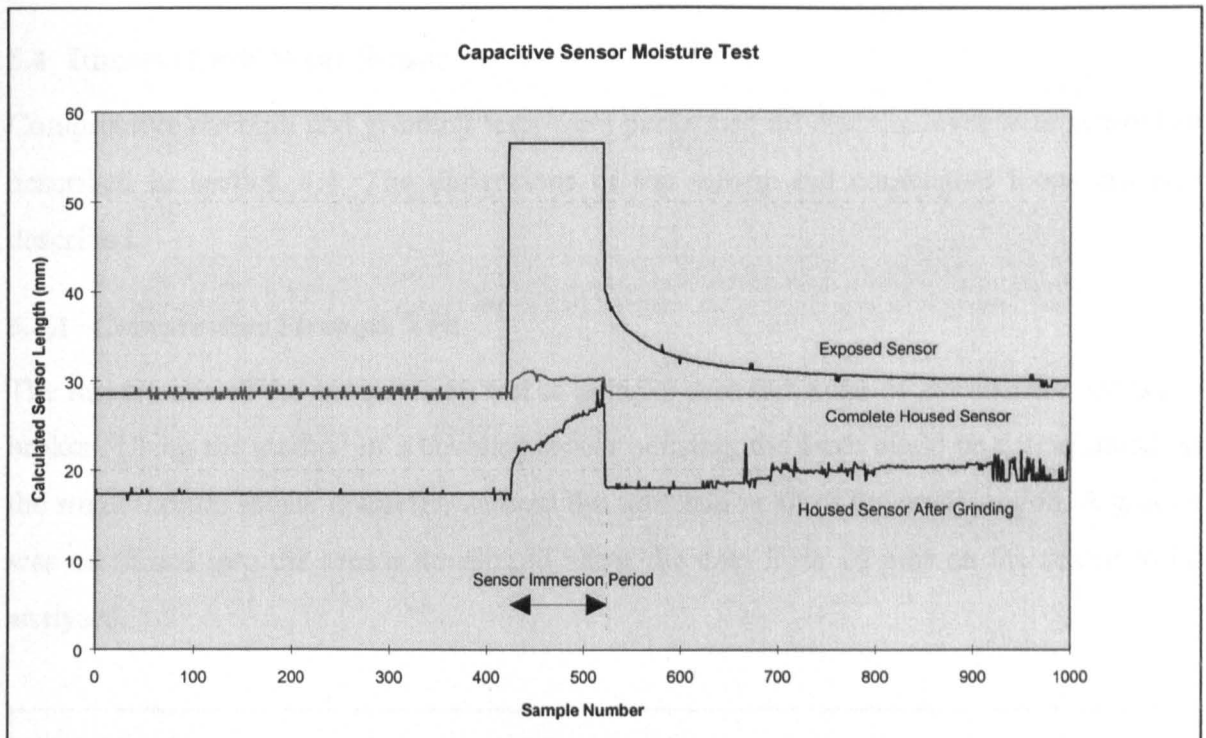
Figure 5.25 - Capacitive Sensor Moisture Test

From the results, it is noticeable that the sensor output is affected in all three cases when immersed in water. Erratic behaviour of the ground sensor after immersion can also be noted. These tests show that if this type of sensor is used in an environment in which moisture is prominent, then inaccuracies in the sensor output will appear.

5.3.6 Field Test

No field tests have been performed on the capacitive sensor.

5.4 Discrete Level Wear Sensor

Compressive strength and grinding tests were performed on discrete level wear sensors as described in section 4.4. The dimensions of the sensor and conductive loops are also described.

5.4.1 Compressive Strength Test

The importance of the compression test is to make sure that none of the discrete levels are broken. Using the method of a bevelled sensor housing, the force could be concentrated on the small 0.6mm sensor diameter, without the steel taking all of the compression. A groove was machined into the sensor housing to allow the data from 16 pins on the sensor to be analysed.
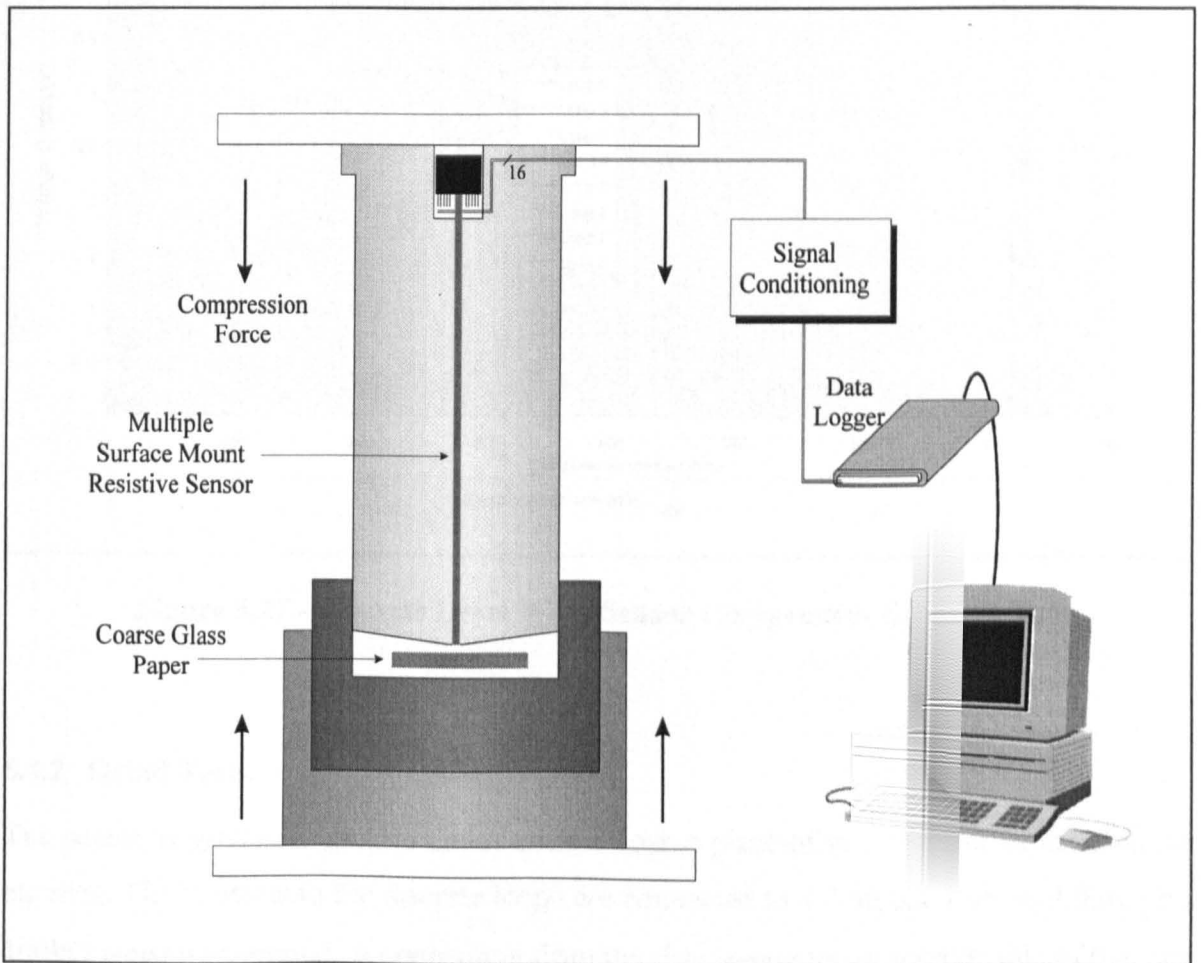
Figure 5.26 - Discrete Level Wear Sensor Strength Test

Coarse glass paper was glued to the mating housing to simulate small rock particles, and to stop any 'short circuit' between the sensor and the steel housing. The conversion circuit used 2.2kΩ & 10kΩ resistors to give an output of around 4V for an unbroken discrete level of the sensor, and an output of 0V if the discrete level is broken.

Figure 5.27 shows the test results of the strength test. At around 325kN (Compressive Stress 718MPa) the discrete levels start to break down.
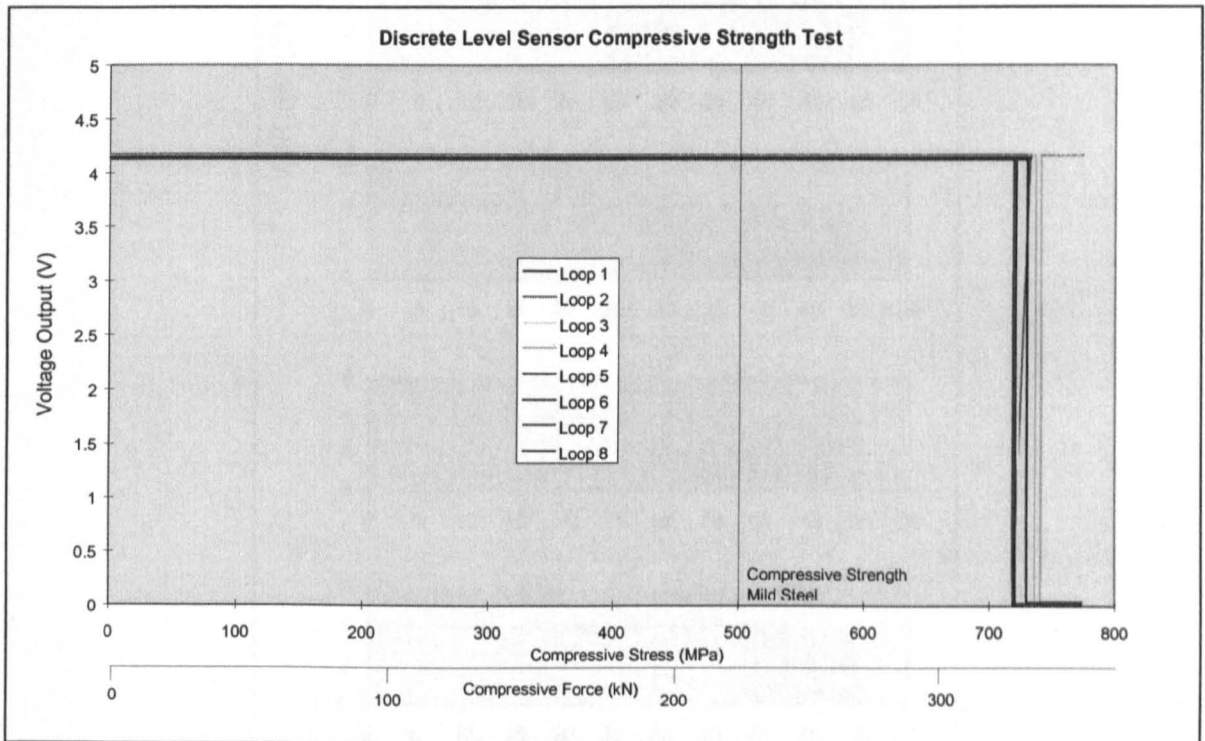
Figure 5.27 - Discrete Level Wear Sensor Compressive Strength Test

5.4.2 Grind Test

The sensor is gradually ground down using a coarse piece of emery paper as the abrasive element. The contacts to the discrete loops are connected to 4V on one side, and through a 100kΩ resistor to ground. A connection from the data logger to the resistor side of the loop provides a 4V output for an unbroken loop, and 0V for a broken loop. Measurement of the sensor length is provided by a vernier calliper.
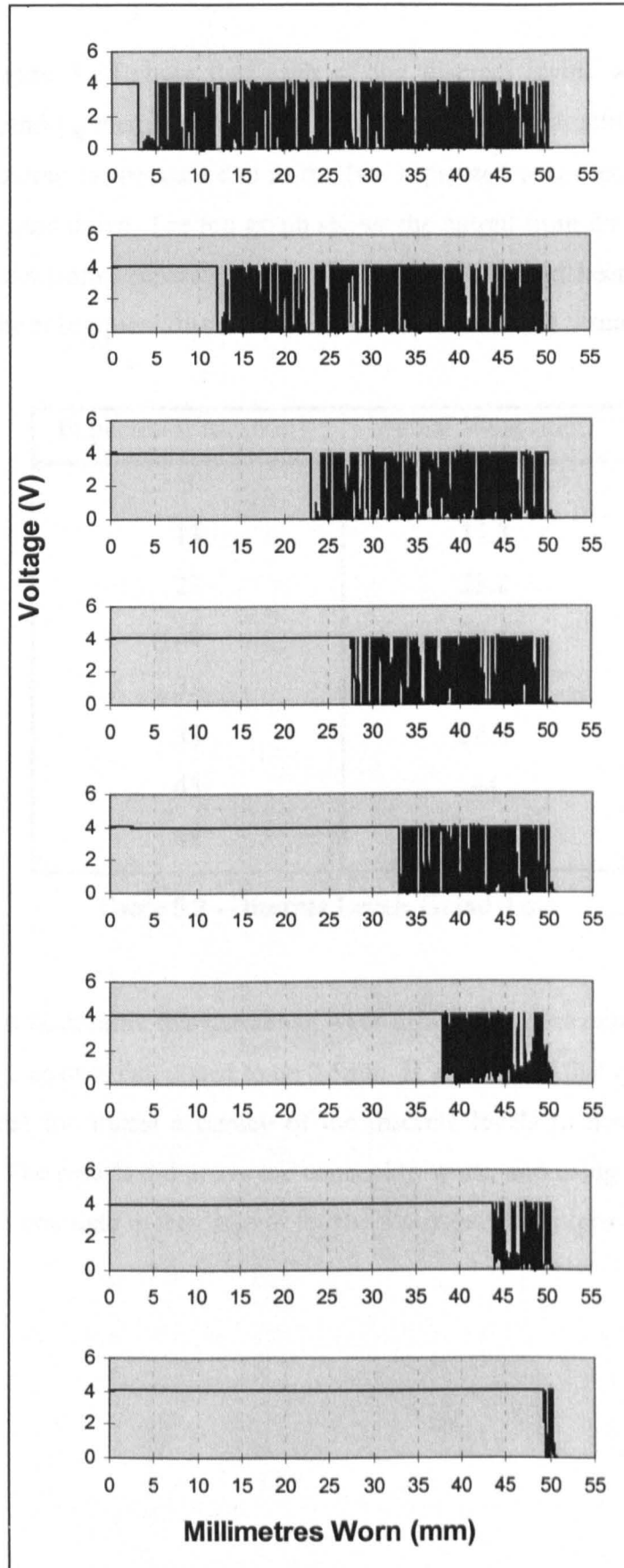
Figure 5.28 - Discrete Level Grind Test

The results in Figure 5.28 show that each of the discrete levels were stable prior to exposure to the grinding element. As soon as the tip of the conductive wire is removed, instability in the output is apparent due to the two separated wires temporarily contacting as the sensor is ground down. The top graph shows the output from the longest loop, and it can be seen that the initial separation was when the sensor had been worn away 3mm. Table 5.2 shows the subsequent discrete levels, with expected and actual wear positions.

Expected Wear (mm)	Actual Wear (mm)
3	3.1
13	12.3
23	23.2
28	28.4
33	33.2
38	38.3
43	44
48	49.1

Table 5.2 - Discrete Levels Grind Test

From the table, it is noticeable that the actual wear differs from the expected wear by up to 1.1mm. An average error is calculated to be 0.5mm. If a sensor of this type is to be utilised, it is imperative that the initial accuracy of the discrete levels in development is in sub millimetre levels. The results did prove the concept to work, and using a sensor with wider bore width to accommodate more discrete levels, a sensor with improved resolution could be developed.
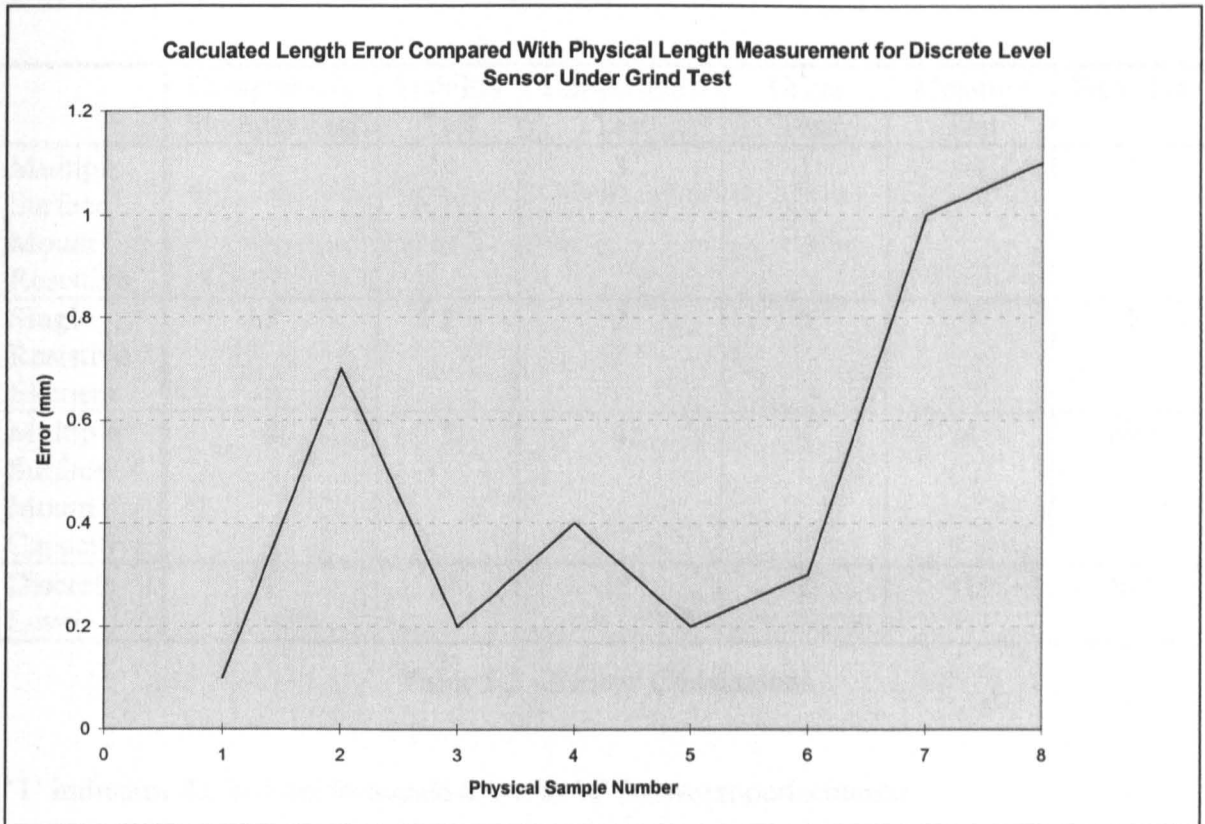
Figure 5.29 - Calculated Length Error Compared With Physical Length Measurement for Discrete Level Sensor Under Grind Test

5.4.3 Field Tests

As the discrete level sensor initially offered poor resolution, a lengthy field test would provide no more information than a laboratory grind test.

5.5 Conclusions

The laboratory and field tests of each of the sensors provides an indication of the accuracy and reliability of the sensor. Table 5.3 shows the relative performance of each sensor with respect to the other sensor for each test.

	Compressive Strength Test	Stability Test	Temperature Test	Grind Test	Moisture Test	Field Test
Multiple Surface Mount Resistive	2	4	3	1	=1	1
Single Resistive Element	3	2	2	3	3	2
Multiple Surface Mount Capacitive	4	3	4	4	4	N/A
Discrete Level	1	1*	1*	2	=1*	N/A

Table 5.3 - Sensor Conclusions

'1' indicates the best performance down to '4' for worst performance.

N/A - Not Applicable.

* The discrete level sensor is noted as performing highly for the stability, temperature and moisture tests without undergoing such a test. The nature of the sensor design means that any external influences such as temperature or moisture will not adversely affect the sensor.

The capacitive sensor performs relatively poorly in the compressive strength test, the temperature test and the grind test. The utilisation of such a sensor in any system would mean using a different dielectric type with lower dielectric constant. In implementing this, an alternative signal conditioning circuit would be necessary to cope with the low capacitances generally found with devices of low dielectric constant.

The single resistive element and multiple surface mount resistive sensors performed similarly for the compressive strength, stability and temperature tests (Table 5.4).

	Compressive Strength Test	Stability Test	Temperature Test
Multiple Surface Mount Resistive	1.55mm variation at compressive strength	+0.183/-0.135mm	+0.109/-0.279mm
Single Resistive Element	1.82mm variation at compressive strength	+0.146/-0.059mm	+0.350mm

Table 5.4 - Resistive Sensor Type Laboratory Test Comparison

The multiple surface mount resistive sensor out-performed the single resistive element in the grind, moisture and field tests. Under grinding, the multiple surface mount resistive sensor provided an output with a calculated sensor length similar to the length measured from the vernier callipers. This proved to be accurate throughout the grinding test. The single resistive element, however, produced an output which is extremely erratic under grinding. The moisture test shows the multiple surface mount resistive sensor operating normally whilst immersed in water. The output of the single resistive element, however, changes for the ground housed sensor during immersion. This is typically a calculated increase of 0.875mm which is noticeable instantaneously. Finally the field tests show the output of the single resistive element as a series of large 'steps', i.e. the sensor output changes dramatically during a single sample and remains at the new level. This is possibly due to the brittle nature of the sensor, and weaknesses forming down the centre of the sensor. This problem is not apparent with the multiple surface mount resistive sensor, and a more accurate field test is shown.

The two sensors with highest performance are the multiple surface mount resistive and the discrete level type. The higher resolution of the multiple surface mount resistive sensor (0.55mm as opposed to 3mm for the discrete level sensor) means that this sensor is the most appropriate for the cone crusher application.

CHAPTER 6
DEVELOPED SYSTEM

6. DEVELOPED SYSTEM

Several modules combine to develop an automated condition monitoring and real time control system for cone crushers. The developed system uses a Knowledge Based System (KBS) for graphical user interface, a distributed control and data acquisition system, novel sacrificial wear sensors for insertion into the cone crusher liners and dedicated wear algorithms and control strategy [Moshgbar, 1994c] to interpolate the readings from the sensors and control the crusher. A schematic of this can be seen in Figure 6.1.
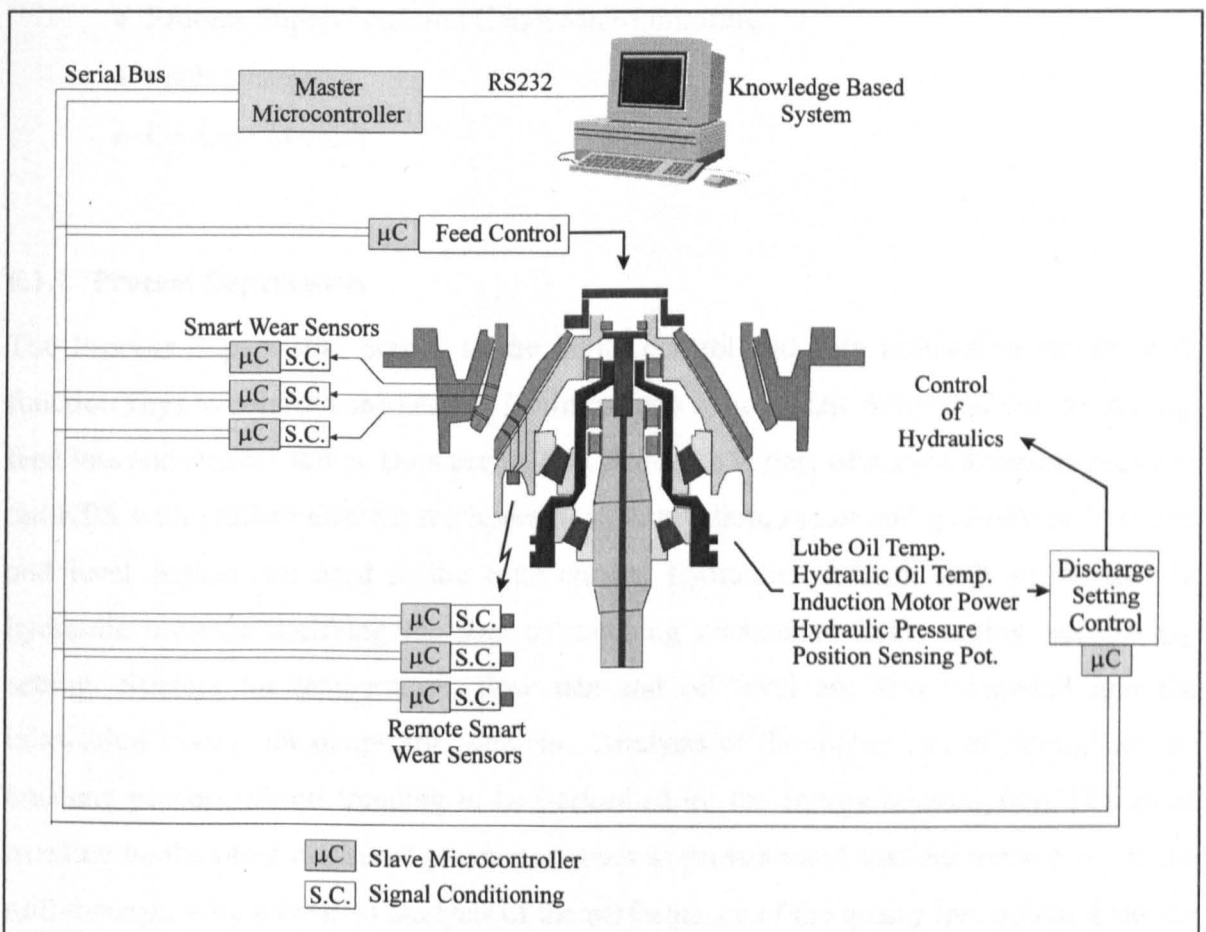
Figure 6.1 - Control Structure

6.1 Knowledge Based System (KBS)

The development of Knowledge Based Systems is generally through a high level language such as LISP, PROLOG or C or through a specific KBS Shell. Whilst the high level

language offers flexibility in the programming environment in terms of increased functionality, the KBS shell provides the system developer with a fully integrated development shell with rapid prototyping abilities. In order to provide a low-cost KBS to the quarrying industry, PC based Knowledge Based Shells were considered as opposed to systems which rely on workstations. The CRYSTAL shell was chosen for its ease of use, compatibility with standard software, and hierarchical rule based programming environment. The utilisation of interfaces enable the shell to provide the user with graphical interfaces, serial communication, mathematical analysis and data import/export.

The cone crusher KBS has three major functions.

- ◆ Process Supervision and Condition Monitoring
- ◆ Fault Diagnosis
- ◆ On-Line Manual

6.1.1 Process Supervision

The Process Supervision Screen is the initial control and data acquisition screen, with function keys to control cone crusher facilities such as automatic or manual crusher setting, feed rate and crusher status. Data acquisition through a variety of standard sensors provides the KBS with crusher data for the hydraulics, lubrication, motor and gap setting. Pressure and level sensors are used in the cone crusher hydraulics system, with an increase in hydraulic pressure notifying the user of touching manganese liners during manual gap setting. Sensors for temperature, flow rate and oil level are also integrated into the lubrication system for diagnostic purposes. Analysis of the motor current throughout the crushing process allows trending to be performed on the energy consumption. Historical trending on the other measured parameters such as pressure and cost per tonne provide the mill manager with a detailed analysis of the performance of the quarry throughout a variety of system conditions. Data from each of the wear sensors within the crusher provide the KBS with accurate information on the state of the liners throughout their life cycle, and trends of undesired wear profiles enable the optimum crushing regime to be formulated.

Throughout the running of the KBS, present alarms and notices appear in the current screen to advise on the state of the crusher whilst in operation.

Figure 6.2 shows the Process Supervision Screen.
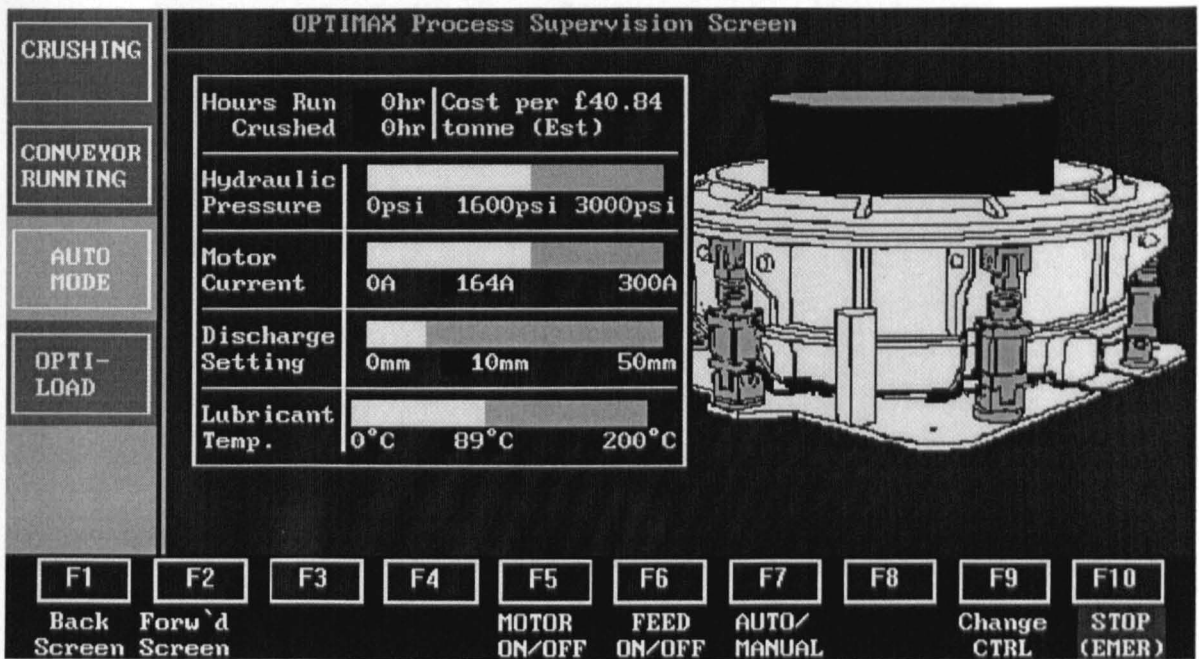
Figure 6.2 - KBS Process Supervision Screen

6.1.2 Fault Diagnosis

In addition to the control and data acquisition, a fault diagnosis knowledge base has been developed which enables step by step maintenance to be carried out by non-skilled service operators. The fault diagnosis rule base contains detailed mechanical and electrical drawings for the particular problem area, with a hypertext facility for detailed descriptions of each of the individual components. A series of screens are shown in Figure 6.3 which analyse the fault of 'Cone Head Will Not Turn Freely'.

PEGSON
P
LIMITED
Trans Tec

O P T I M A X
Autocone Fault Diagnosis Advisor Module

De Montfort University
Pegson Limited

Which of the following describes the fault:

1 Air-Blast Cooler Will Not Work
2 Anti-rotation Stop Broken
3 Cone Head Spins When Crushing
4 Cone Head Will Not Turn Freely
5 Cone Judders on Adjustment
6 Crusher Alarm Sounds
7 Crusher Continually Dumping
8 Crusher Stalls Under Load
9 Crusher Starts But Will Not Run
A Crusher Will Not Move in MANUAL
B Crusher Will Not Return To Setting
C Crusher Will Not Start

D Digital Setting Indicator
E Hydraulic Pump Runs Continually
F Immersion Heater Will Not Work
G Lubrication Oil
H Ram(s) Broken
I Vee Belts Burnt Out
J Vibration
K Return to Main Menu

PEGSON
P
LIMITED
Trans Tec

O P T I M A X
Autocone Fault Diagnosis Advisor Module

De Montfort University
Pegson Limited

Check Skirt

Empty crusher, stop crushing. Inspect the underneath of the crusher to
check whether a stone is wedged between the skirt of the cone head and
the rotary seal ring.

Has any stone been found between the skirt and the rotary seal ring ?

n   Y or N

PEGSON
P
LIMITED
Trans Tec

O P T I M A X
Autocone Fault Diagnosis Advisor Module

De Montfort University
Pegson Limited

Check for Build Up of Material in the Discharge
Chute

Empty and stop the crusher. Inspect discharge
chute.

Is there a build up of material pressing against
the cone head ? Y   (Y or N)

PEGSON
P
LIMITED
Trans Tec

O P T I M A X
Autocone Fault Diagnosis Advisor Module

De Montfort University
Pegson Limited

Material Build Up in Discharge Chute

Remove material and determine reason for build
up. This can be either;

a: Discharge conveyor not working / slow.
b: Chute problem.

Figure 6.3 - Fault Diagnosis Screens

Upon choosing option 4, 'cone head will not turn freely', the next stage of inspection runs the service engineer through a series of yes/no questions. The answer is then provided with either a series of commands, or in severe cases the service engineer is asked to notify the equipment manufacturers.

6.1.3 On-Line Manual

To complement the system is an on-line manual (Figure 6.4) including operating and installation directions. This again uses hypertext to focus on user targeted areas and find the required information as quick as possible.
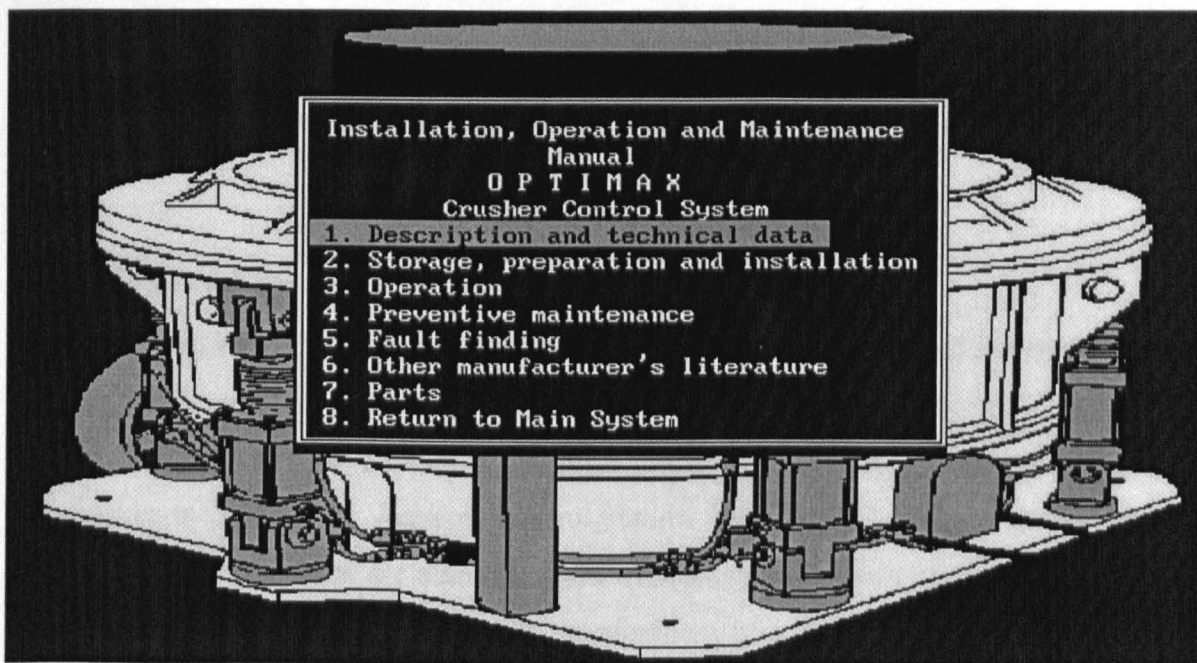
Figure 6.4 - On-Line Manual Opening Screen

The on-line manual is taken from the paper based manual distributed with the cone crushers. The utilisation of the hypertext function reduces the time to finding relevant information and related diagrams. A sample page of the technical data is shown in Figure 6.5.

Figure 6.5 - Technical Data from the On-Line Manual

6.1.4 Modified KBS

Due to the nature of the quarrying industry, the systems that are utilised for control of the Pegson Autocones are often specific to that particular quarry. Pegsons' do produce standard control systems (System 2 - a single electro-hydraulic operating console next to the crusher to System 4+ - remote control station with MIMIC diagram of the crusher operation), however, these are often designed with the specific quarry in mind. This in turn produces further complications into the development of a standard condition monitoring system. The designed system has to be modular with flexibility to connect to existing parts of the control system. A modified version of the previous KBS was installed at Barton-Under-Needwood quarry which used the existing control and data acquisition system.
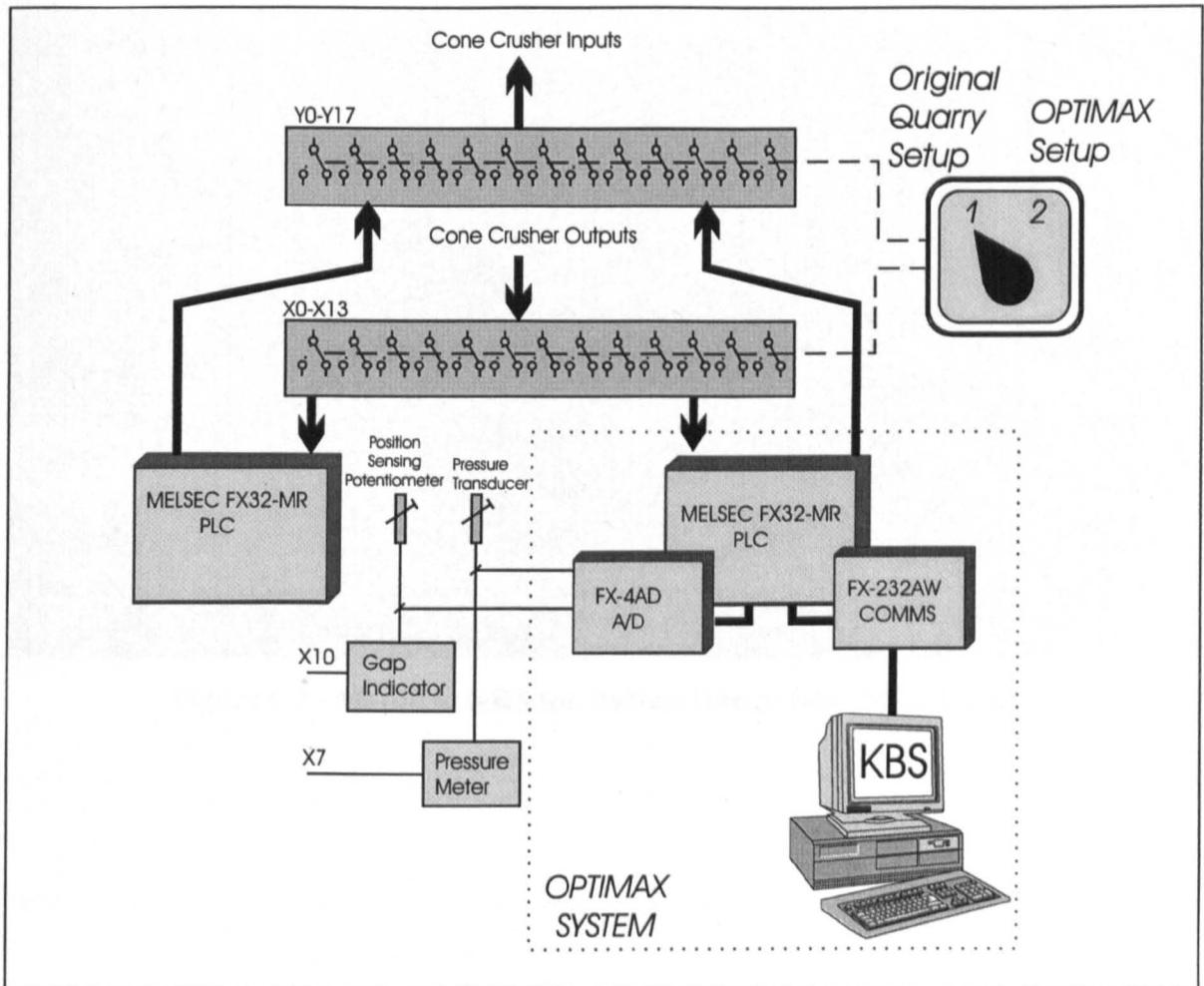
Figure 6.6 - Installation Schematic of KBS at Barton-Under-Needwood Quarry

Figure 6.6 shows the schematic of the KBS installation. In order to keep the crusher fully functioning within the quarry, a duplicate PLC system was used. The ladder logic in the duplicate PLC was modified to provide signals for the KBS. An A/D module allowed information to be retrieved from the pressure meter and gap indicator. The Communication module allows two was communication between the PLC and the KBS.

A screen dump from the modified KBS used at the quarry is shown in Figure 6.7.

Figure 6.7 - Modified KBS for Barton-Under-Needwood Quarry

6.2 Distributed Acquisition and Control System

The KBS links to the master microcontroller of the distributed acquisition and control system via a serial RS232 communications link. NEC 78330 16-bit microcontrollers have been implemented for each of the master and slave devices. An integral serial bus then links to a series of smart sensors and localised controllers to enable data to be retrieved and disseminated at the master microcontrollers request. The structure is modular in design, and extra slave modules can be attached to the two-wire serial bus. Data is taken from standard sensors for hydraulic and lube oil temperature, hydraulic pressure, induction motor power and from the position sensing potentiometer for the gap setting.

6.2.1 Serial Bus Protocol

Information is passed between master and slave devices by way of a two wire connection, serial clock and serial data bus. To enable any transfer of data, each of the slave devices need to know which slave the master is talking to, and what to do. Because of this, an ADDRESS-COMMAND-DATA protocol is implemented, which eliminates the need for any handshaking lines. Each of the slave devices on the bus has an 8-bit address. The master controller communicates with the desired slave by sensing the slaves address after some pre-processing on the SB0 line, i.e. toggle the line LOW, HIGH, LOW. Pre-processing for a COMMAND, is toggling the SB0 line LOW. After this COMMAND has occurred, data can be sent either way down the serial bus depending upon the COMMAND. The structure of the serial bus, and the timing diagrams for the protocol can be seen in Figure 6.8. It is imperative that the device sending the ADDRESS, COMMAND, or DATA, sends the clock signal to synchronise the 8-bit data.
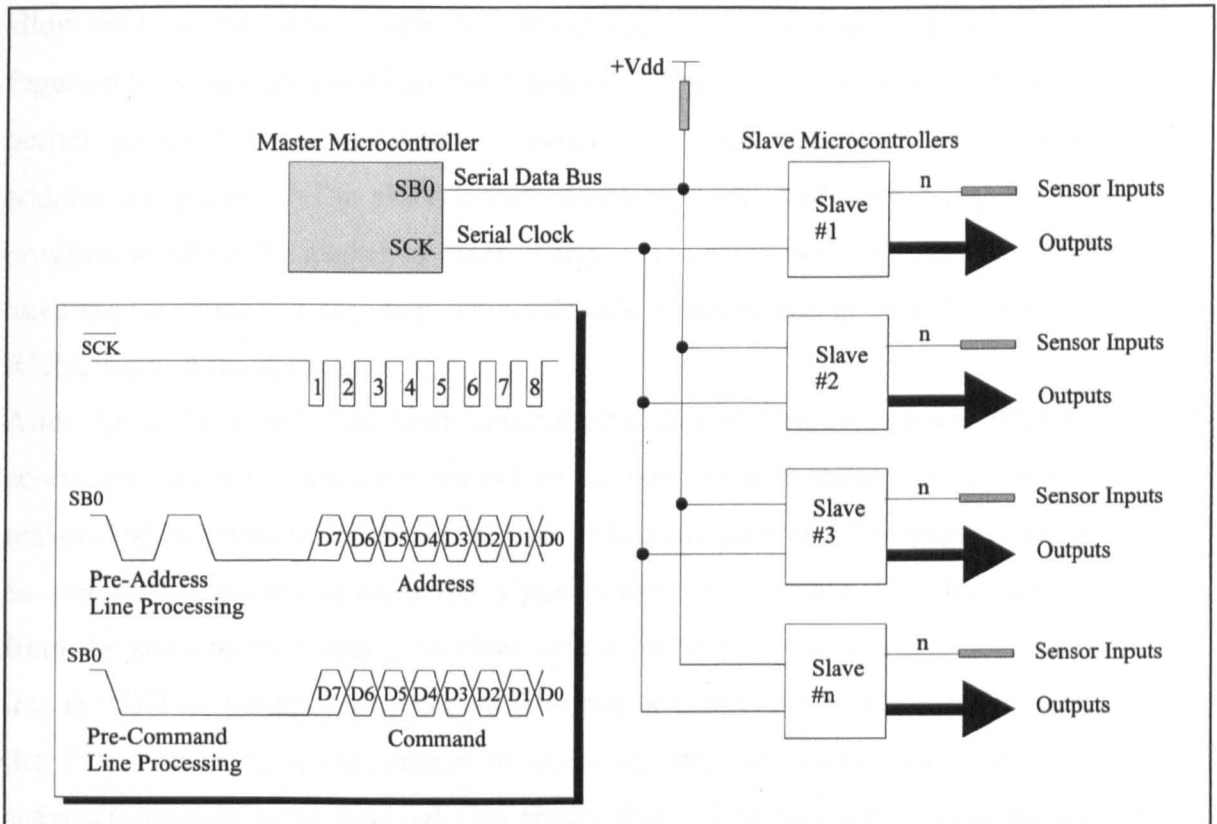
Figure 6.8 - Serial Bus Protocol and Hardware Connections

6.2.2 Program Structure

The master and slave control software is interrupt driven, with communication between nodes controlled by flags set in the interrupt service routine. Upon switch on of the system, the master microcontroller is waiting for an RS232 interrupt from the KBS. The KBS will pass an option for operation mode, i.e. run crusher, set up parameters etc. Upon receipt of this RS232 interrupt, the master device will run through a series of pre-determined functions. The first of these is a slave search. The master microcontroller toggles the SB0 line to notify the slave devices that an address is to be sent on the bus. The master then proceeds to put an address, starting at 0x00, onto the bus. Each of the slave devices wait for an interrupt from the serial bus. The master will increment the address each time, and when a slave device has the same address as the sent address, an acknowledge flag (ACKT) is toggled and the master notified. The master continues to increment the address until it reaches 0xFF (256). It is imperative that a time out function is implemented by the master microcontroller so as to leave a slave device enough time to reply if the address is the same. If this time out period is exceeded, the master has to provide its own acknowledge to

allow the next address to be sent. The timing diagram for the time out period can be seen in Figure 6.9. A time period of 20 clock pulses or 80μs proved to be enough for a time out period, as slave acknowledgements usually occur after around 10 clock pulses of the address transmission. The slave search sequence is repeated continually throughout the program to allow the master to know if any of the slave devices have been removed, or have malfunctioned in any way. The addresses of active slaves on the bus is transmitted RS232 back to the KBS.

After the slave search has been initiated, the master controller sends relevant control commands to each of the slave devices on the bus. Some of these commands will involve retrieval of information regarding the state of transducers within the crusher, and some will be straight commands to carry out a pre-determined sequence. If information is passed from the slave to the master, the slave device controls the serial clock, and puts the data into the SIO for the master to receive. The master then has to produce an acknowledge for the data. The software is written to allow for multiple transitions of the data, if an acknowledgement is not received. One feature that will be necessary, where delicate data is concerned, is that of error checking. This may be necessary for information retrieved from the wear sensors if ultimate control of the crushers is dependent on this value.
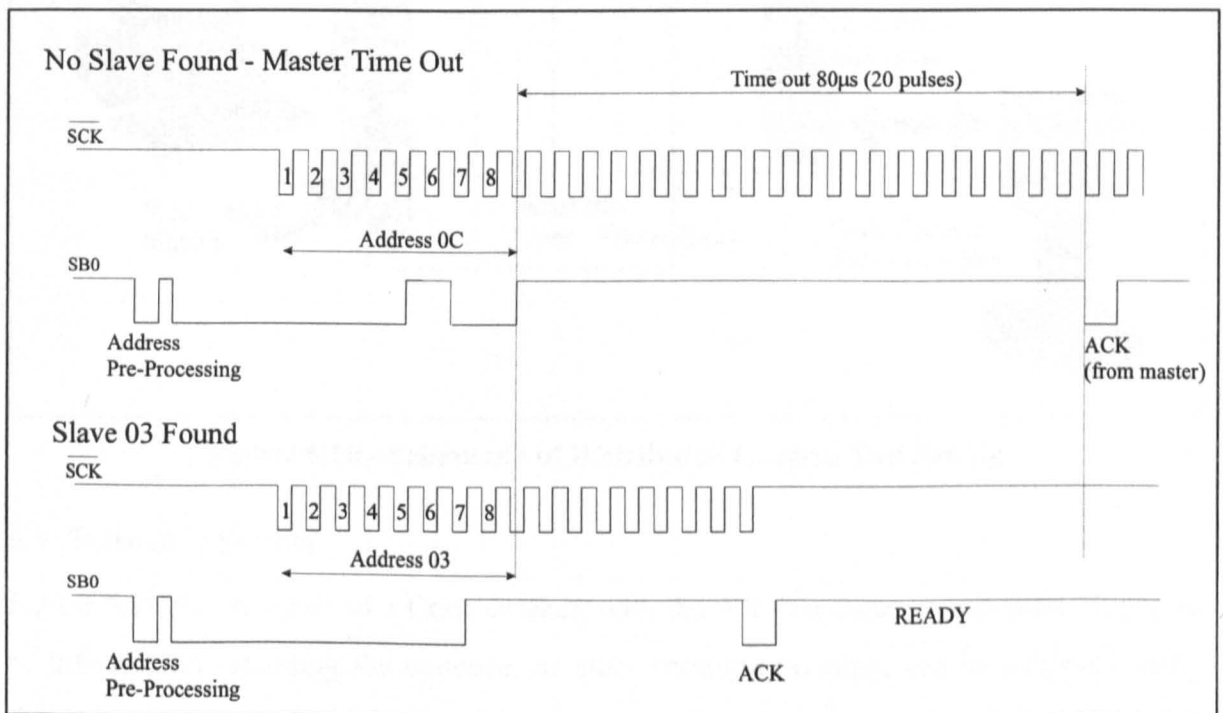
Figure 6.9 - Acknowledge Time Out Timing Diagrams

6.2.3 Test Set-Up

Laboratory tests on the system were performed. A schematic of the set-up is shown in Figure 6.10. The diagram shows the KBS communicating RS232 to the master controller. This, in turn, can access any device on the bus. Slave 1 is used for control of the crusher. An cone crusher signal emulator is used which simulates typical inputs and outputs of a crusher. Slave 2 is used for wear sensor inputs, with dedicated signal conditioning for each type of sensor. Additional slaves can be added by 'clipping' onto the serial bus.
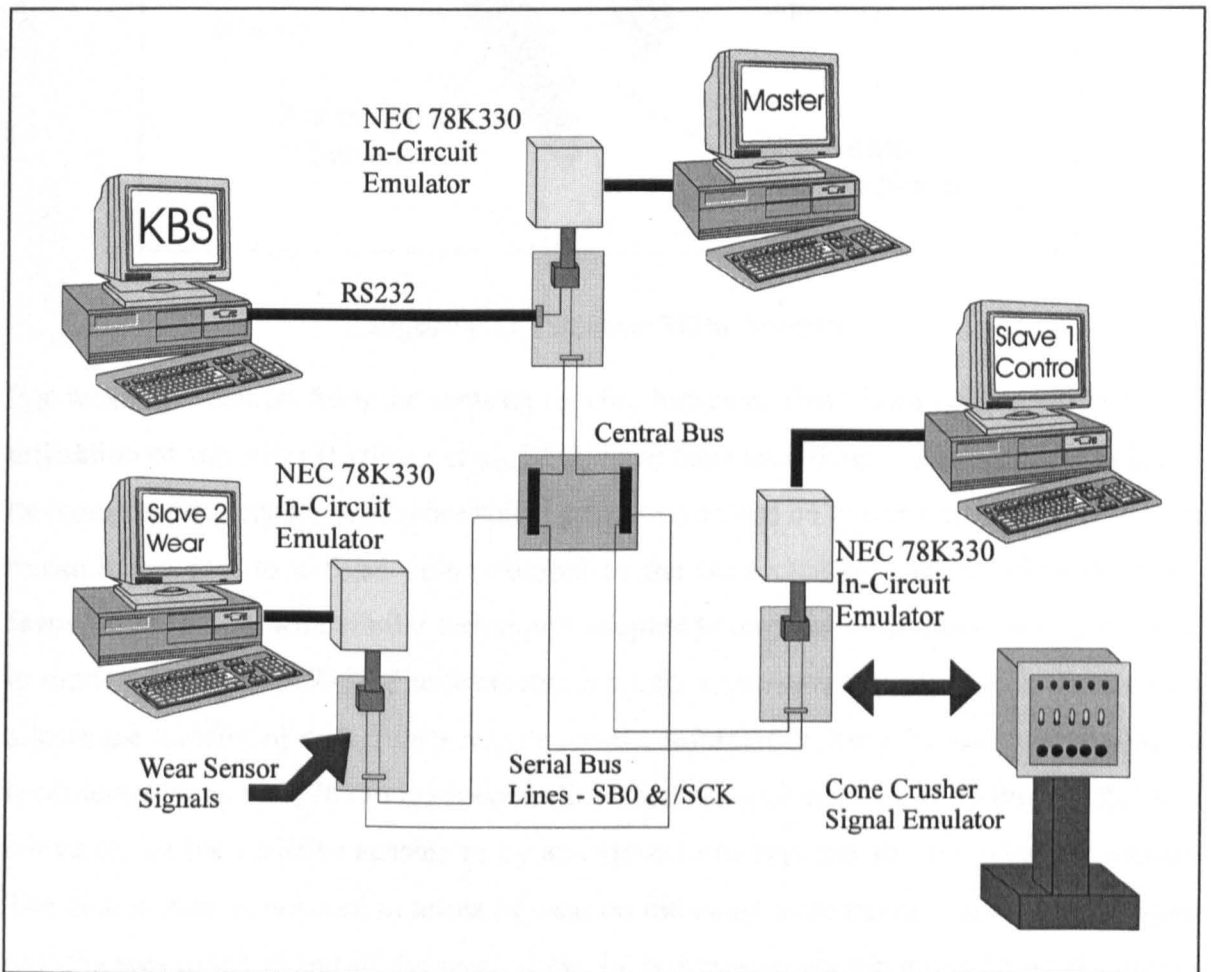
Figure 6.10 - Schematic of Distributed Control Test Set-Up

6.3 Telemetry System

Figure 6.11 shows a part of a Cone crusher, with the full schematic shown inset. Retrieval of information regarding the concave, or outer crushing member, can be achieved using fixed wire links, which are fed through the outer body of the cone crusher.
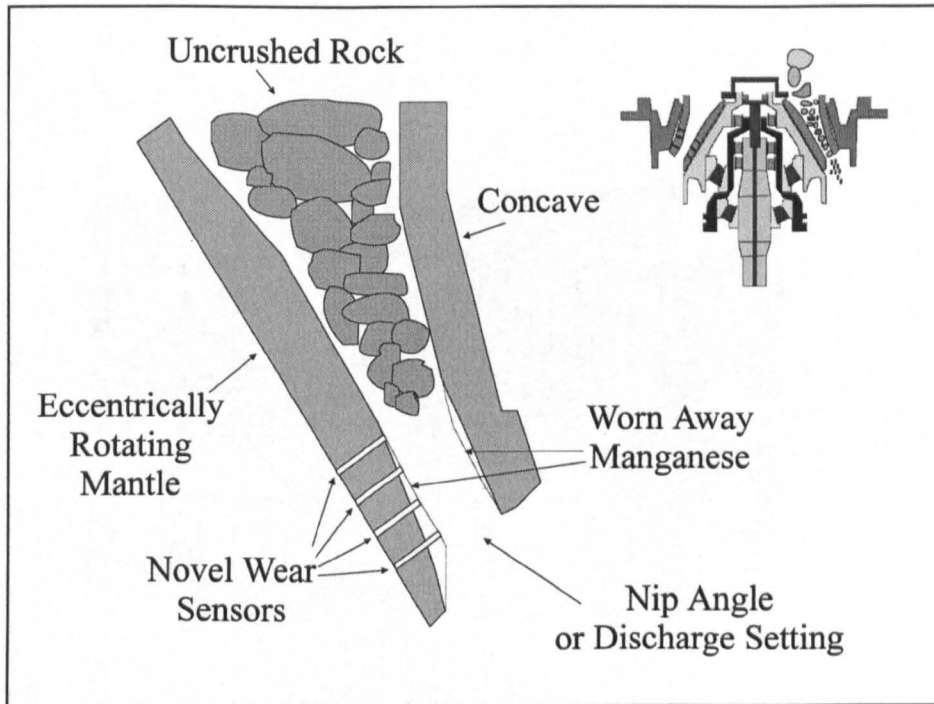
Figure 6.11 - Remote Wear Sensors

For wear information from the rotating mantle, however, fixed links cannot be used. The utilisation of slip rings [Lizheng et al, 1990] have been considered, although a re-design of the conehead is necessary. A more suitable solution would be to use a telemetry module to transmit the data to a fixed point external to the cone crusher. This is often the more favoured approach, with similar techniques adopted to monitor vehicle tyre pressure whilst in motion [Schrader, 1995]. The telemetry is purely a one-way (simplex) radio link, which allows the transfer of data from a remote source. Information from the novel wear sensors is retrieved by an NEC78330 microcontroller either through any one of 16 lines of the A/D converter for the resistive sensors or by an external interrupt pin for the capacitive sensor. The data is then deciphered in terms of wear on the manganese liners, and depending upon any changes to the extent of the wear, it can be transmitted via the transmitting module to the receiver unit. The information can either be sent digitally encoded with a unique address and data, or directly RS232 from the microcontroller.

The transmitter is approved to DTI (RA) MPT1340, which allows general licence exempt telemetry, telecommand and alarms at 417.90 - 418.10MHz transmission, with a maximum radiated power (ERP) of 250μW. It is a small (30mm x 9.5mm) hybrid printed circuit board mounting self contained module capable of transferring data up to 200 metres.
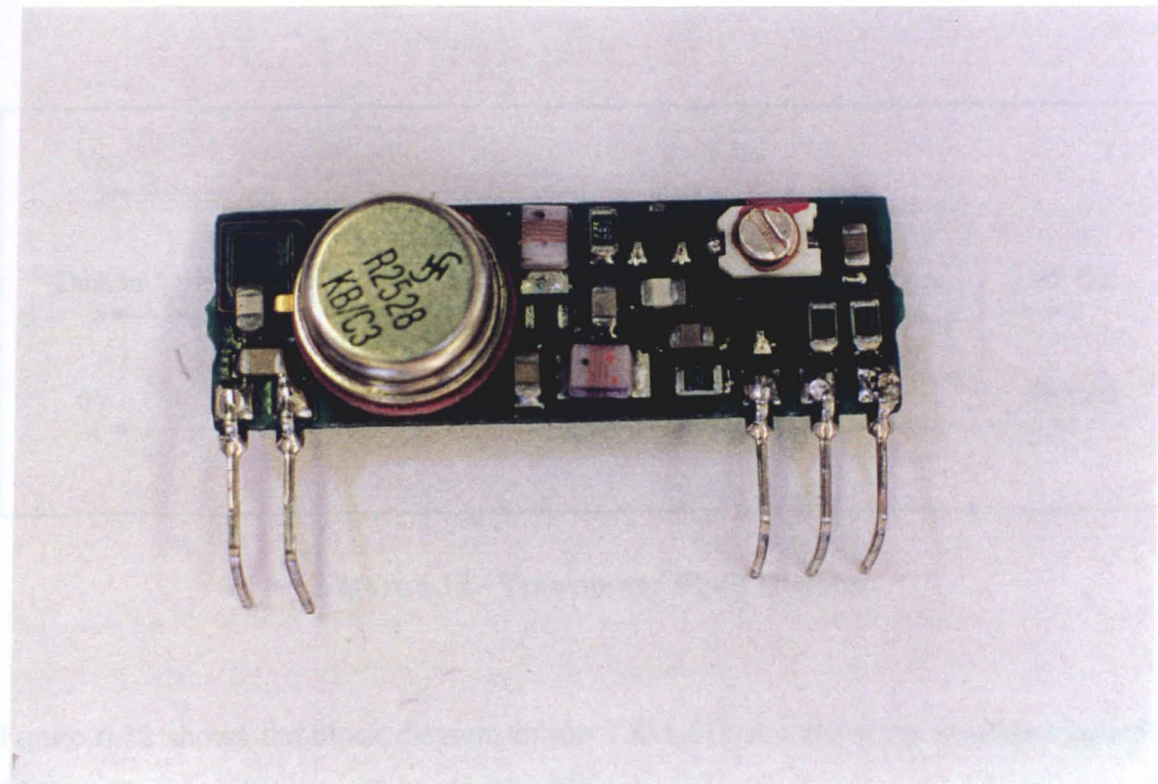
Plate 6.2 - Photograph of Transmitter
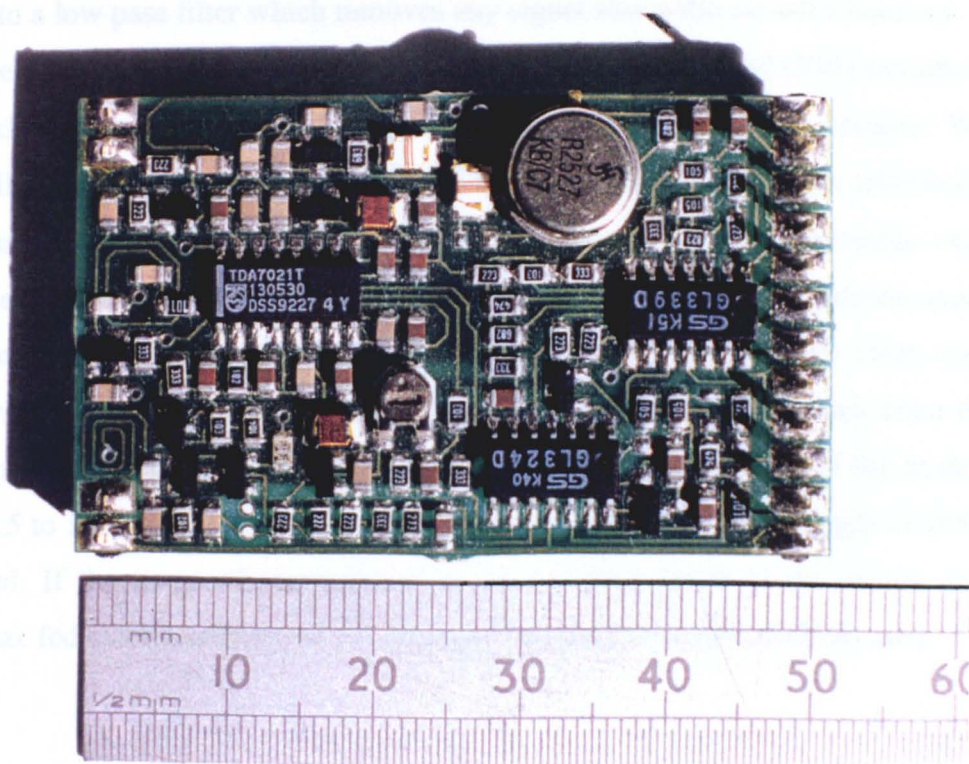
Plate 6.3 - Photograph of Receiver
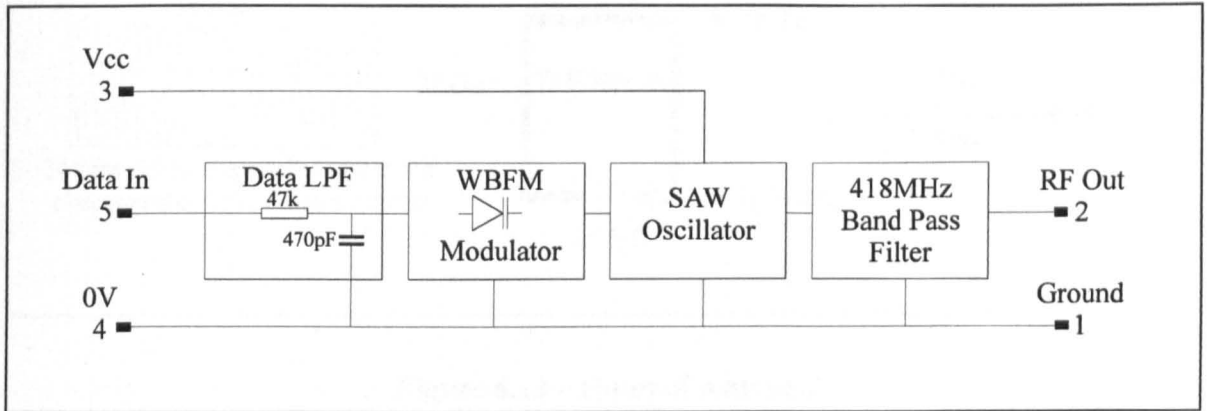
Figure 6.12 - Transmitter Block Diagram

Figure 6.12 shows the block diagram of the TXM-418-A transmitter module [Quantelec, 1992a]. The module has a 10kbps bandwidth and can transmit analogue or digital data. The data is fed into a low pass filter which removes any signal above the cut-off frequency. The signal is wide band frequency modulated, with a Frequency Modulated (FM) deviation of typically 25kHz around the transmit frequency of 418MHz. A Surface Acoustic Wave (SAW) Oscillator and band pass filter clean up the signal and maintain the tolerance for data transmission within the MPT1340 guidelines. Any one of a series of antennas can be used, dependent upon space available and surrounding hazards. A helical antenna specific for this transmitter would need 34 turns of 0.5mm enamelled copper wire close wound around a 2.5mm diameter former. A loop antenna is printed circuit board track 1mm thick which has an inside area enclosing between 40 and 100mm^2. The ends of the track are joined by a 1.5 to 5pF capacitor. A single whip antenna with an optimum length of 165mm could be used. If the range of transmission is the deciding factor in the design of the antenna, a coax fed external dipole, or 1/4 wave ground plane antenna could be used.
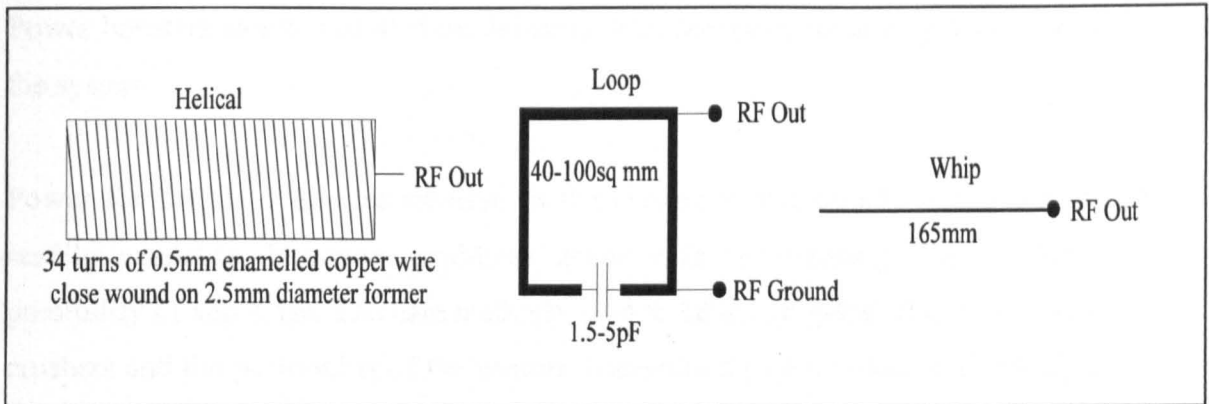
Figure 6.13 - Types of Antenna

Facilities on the receiver unit allow the integrity of the data to be questioned. The Received Signal Strength Indicator (RSSI) is useful for monitoring the performance of the radio link [Quantelec, 1992b]. By feeding this into the analogue to digital converter on the microcontroller, the integrity of the transmitted data can be analysed by cross referencing the checksum with the signal strength. The DETECT pin recognises an incoming signal which is strong enough to provide clean decodable data. The JAM pin detects when a strong signal has been present on the receiver input for a time greater than that set by an external capacitor on pin 12. Finally, the TAMPER pin is activated if the resistance of the antenna exceeds 5kΩ.

6.4 Power Considerations

One of the most important consideration for the system is the supply of power to each of the slave devices. The positioning of each of these devices within the quarry will determine the power used. Further constraints are imposed by maximum lengths of wire for serial bus and RS232 communications. Maximum lengths for RS232 communications are in the region of 30 metres. This is the limit imposed for the master to KBS link. With regard to the serial bus links, distance is constrained by the voltage drop between the two communicating devices. Over a long distance, the clock and transmitted data will become unreadable by the other device. The distance between the master and furthest slave should either be kept to a minimum, or data can be transmitted between adjacent slave devices.

Power boosters can be installed on the serial bus, however, these may affect the timing of the system.

Power for the slave devices external to the crusher is not usually a problem, as this is readily available. However, problems occur with the rotating sensors. Without the possibility of slip rings, alternate methods need to be investigated. Due to the size of cone crushers and the positioning of the sensors, transmitted power [Akin et al, 1990] would be unsuitable. Further problems occur with the irregular rotating of the central liner, and the large amounts of manganese steel employed in the cone crusher design. With these design constraints in mind, the supply of power to the remote smart wear sensors must be from long life lithium batteries. Problems, however, with batteries is their size and the relatively short lifespan. It is therefore imperative that the smart wear sensor units use as little current as possible. From the questionnaire for Shardlow quarry (Chapter 5), a typical lifespan of a liner is 12 weeks with an average crushing time of 100 hours. In keeping with this, the lifespan of the smart wear sensor should be comparable.

Typical power consumption of the microcontroller is in the region of 40mA with the transmitter module 10mA. Lithium Manganese Dioxide Batteries are small enough to be acceptable for this application, and have a capacity of 1.3Ah. Thus the length of time before battery saturation of a transmitting wear sensor with internal RAM and ROM used continuously is around 26 hours. As this is not sufficient for installation within a cone crusher, power saving techniques need to be adopted.

Continuous transmission of the state of the wear sensor is unnecessary due to the typical wear rates of the manganese liners. The smart sensor can be set at a sample rate of every second for example. To save battery power whilst the sensor is inactive, i.e. not sampling or transmitting, the processor can be put into standby mode. Two modes are available, HALT and STOP. In HALT mode, the power consumption of the processor is halved and the processor clock stops. In STOP mode, the power consumption is dramatically decreased to 10µA, and the internal clock supply and the program is stopped. A non-maskable interrupt (NMI) returns the processor from STOP mode. This is in the form of an external pulse on port 2.0. By using a low power 555 timer (120µA) to provide regularly

timed pulses, this can be effectively realised. This timer can also be used to switch on the sampling and transmitting circuits. An effective data transmission and reception time is estimated to be 3ms worst case. This is due to the power up settling time of the receiver. Assuming a further 2ms for sampling and conversion of the data, a total of 5ms out of every second is used to sample and transmit the wear sensor data. Figure 6.14 details the power used during and after the 5ms sampling time.
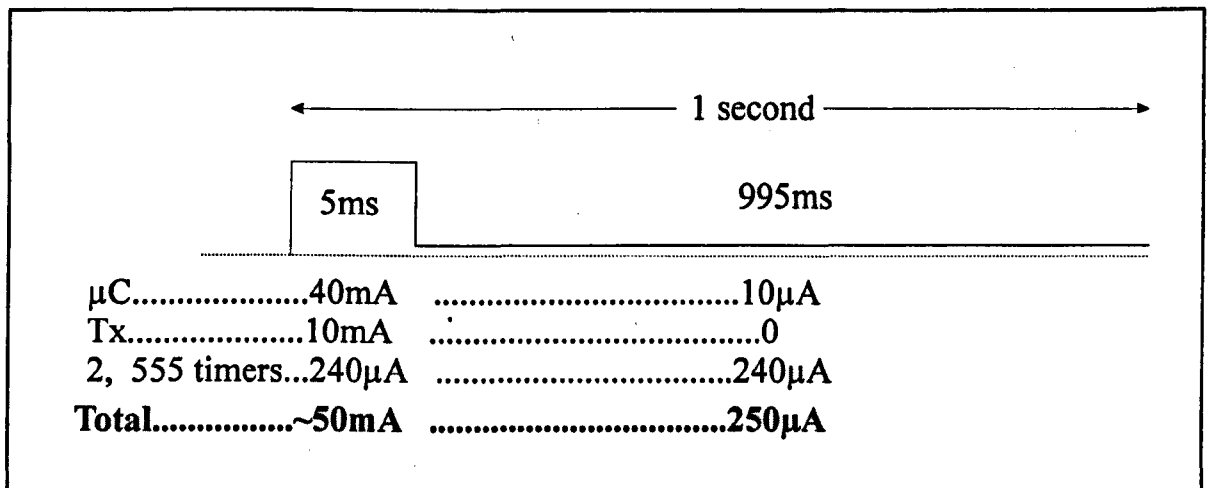
Figure 6.14 - Remote Smart Wear Sensor Power Saving

Total current used = (50mA x 0.005s) + (250μA x 0.995s) = 0.5mA

Therefore, the 1.3Ah battery will last 2600 hours (108 days or 15 weeks). This would be ample for the lifespan of the manganese liners. To increase this figure, further power saving techniques could be adopted such as an increased time between samples or a policy of transmit only when the data has sufficiently changed.

6.5 Remote Smart Wear Sensor Prototype

A laboratory smart sensor integrates the developed wear sensors to a microcontroller with on board A/D conversion, timers and serial bus. This is linked to a telemetry system which transmits wear sensor data from the rotating liner to an intelligent slave receiver with LCD indicator. The on chip serial bus enables the slave to be linked to the master microcontroller. The data can then be transmitted to the KBS to provide on-line accurate measurements of the wear on the manganese liners.
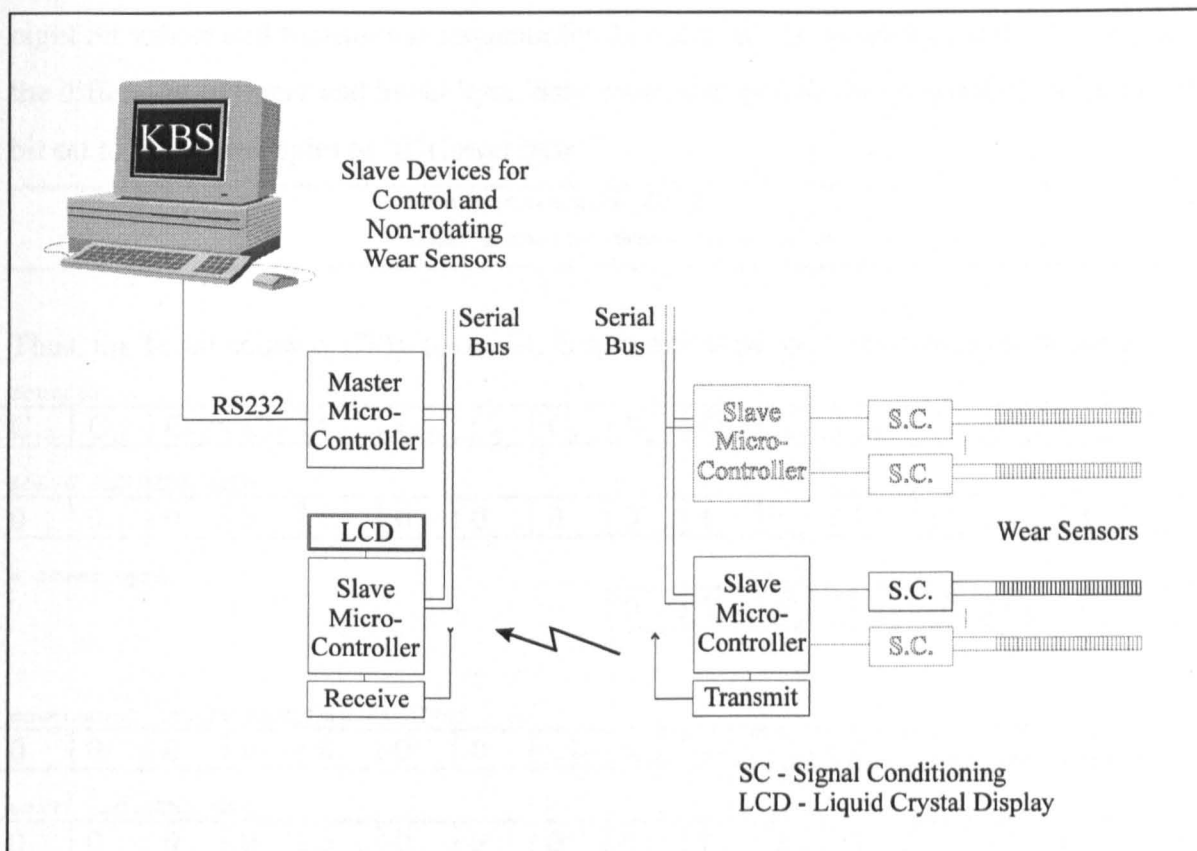
Figure 6.15 - Remote Smart Wear Sensor Transmitter and Receiver Schematic

6.5.1 Remote Smart Wear Sensor Transmitter

This prototype uses a 16-bit NEC78330 microcontroller with external RAM and EPROM. A 16MHz crystal and address decoding are used on the top of two "piggybacked" boards. The lower half is for the signal conditioning of the sensor input and the transmitter electronics. A 16.5mm whip antenna is used to transmit the data. Power is supplied by a single 9V battery. Light Emitting Diodes (LED's) are also integrated into the design for demonstration purposes. A red LED indicates power, and green flashes as the wear sensor data is being transmitted. A capacitive sensor is measured using a 555 timer circuit (4.3.2.1). An oscillating square wave is output from the circuit, with output frequency relative to the capacitance of the sensor. This is fed directly into an interrupt pin (INTP0) on the microcontroller. As the leading edge of the pulse appears on the interrupt pin, an internal timer starts counting until the next available leading edge. As the leading edge appears, the interrupt pin causes an interrupt in the software, in which the corresponding register (CT10) to the timer counter is read. As the data is transmitted bytewise over the telemetry link using the RS232 protocol, the 16-bit counter value needs to be split into 2

eight bit values and transmitted sequentially. In order for the receiving master to recognise the difference in upper and lower byte, only seven bits of data are transmitted, with the top bit set to '1' (upper byte) or '0' (lower byte).

```
lower_byte=counter&0x7f;
upper_byte=((counter>>7)&0x7f)|0x80;
```

Thus, the 16 bit counter, CT10 (counter), is split into upper and lower bytes as follows:-

counter

| C_{15} | C_{14} | C_{13} | C_{12} | C_{11} | C_{10} | C_9 | C_8 | C_7 | C_6 | C_5 | C_4 | C_3 | C_2 | C_1 | C_0 |

&0x7f (LOGICAL AND)

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

= lower_byte

| 0 | C_6 | C_5 | C_4 | C_3 | C_2 | C_1 | C_0 |

counter>>7 (SHIFT RIGHT SEVEN BITS)

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | C_{15} | C_{14} | C_{13} | C_{12} | C_{11} | C_{10} | C_9 | C_8 | C_7 |

&0x7f (LOGICAL AND)

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

|0x80 (LOGICAL OR)

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

=upper_byte

| 1 | C_{13} | C_{12} | C_{11} | C_{10} | C_9 | C_8 | C_7 |
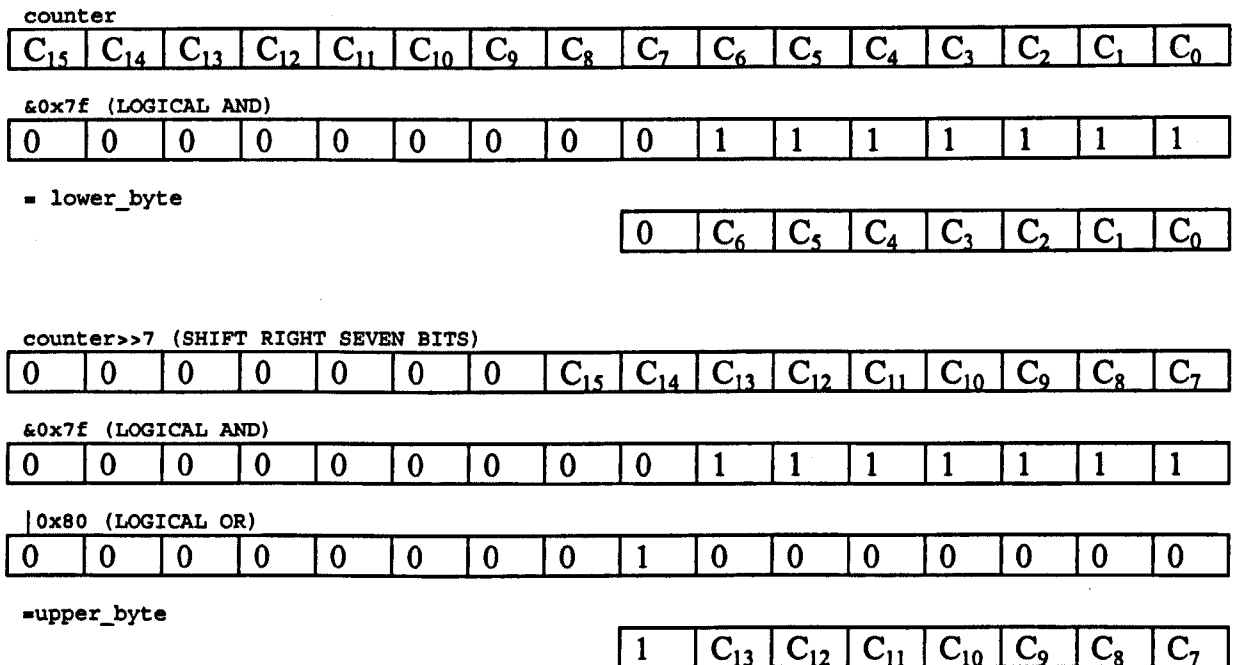
This produces a 14-bit counter value. The counter uses an internal timer which oscillates at a frequency of the internal clock frequency (f_{CLK}) divided by eight. The internal clock frequency runs at half the crystal frequency, thus a 16MHz crystal produces a timer frequency of 1MHz. Using the frequency value for the 555 timer output, and the timer frequency, a value for the internal counter CT10 can be estimated (Table 6.1).

Sensor Length (mm)	No. of capacitors	Capacitance (μF)	Frequency	Microcontroller Counter Value
0.6	1	0.1	7.42kHz	134
49.8	83	8.3	89Hz	11235

Table 6.1 - Related Microcontroller Counter Values

A maximum counter value of 11235 is possible with a 50mm sensor length. Thus, only 14 bits of the 16-bit counter are necessary (2^{14}=16384).

Plate 6.4 shows a photograph of the transmitter prototype unit.
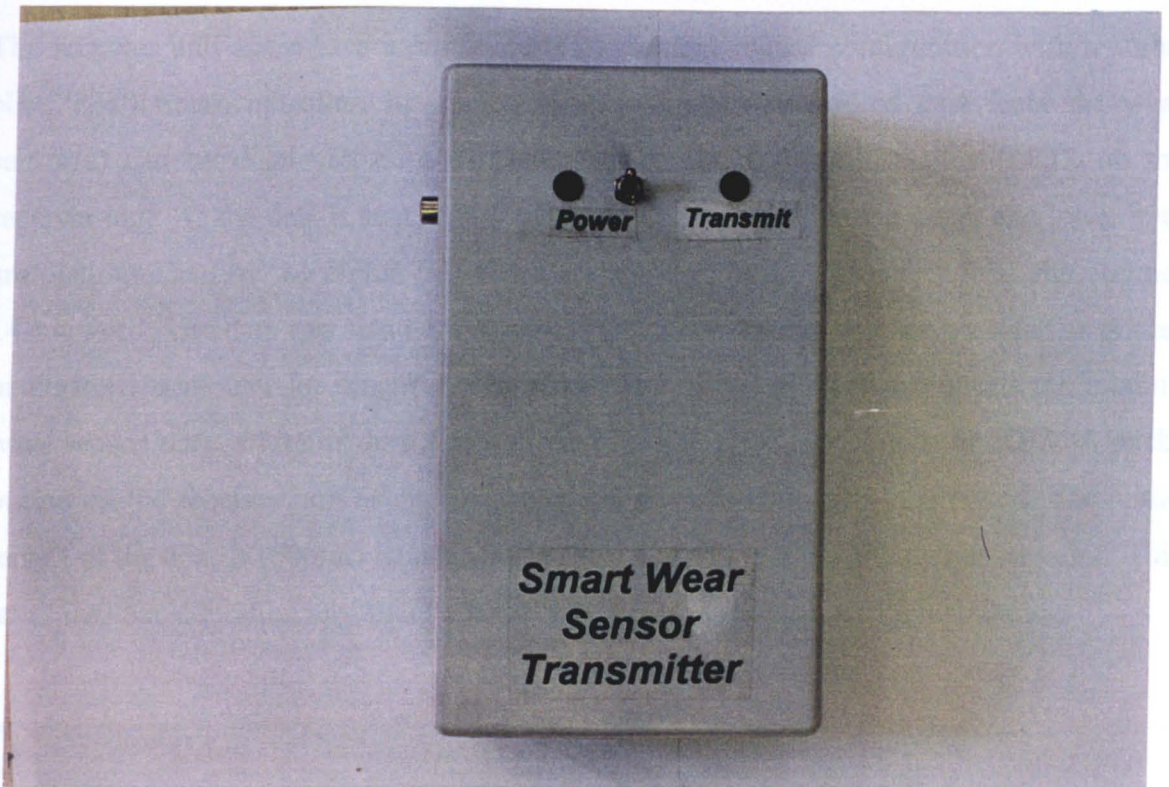
Plate 6.4 - Prototype Smart Wear Sensor Transmitter

6.5.2 Smart Wear Sensor Receiver

The receiver unit again uses a prototype 'piggybacked' board configuration with a 16-bit NEC78330 microcontroller. In stand alone mode, the transmitted data from the wear sensor(s) can be displayed on a 16x2 character Liquid Crystal Display (LCD) on the receiver unit. As the data is sent as two bytes, it is imperative that the upper and lower byte are distinguished by the eighth bit, and that they are pieced together to form the original 14-bit word. The data can also be transmitted RS232 to a peripheral device such as printer or personal computer for diagnostic purposes. The master controller requests the rotating wear sensor data, which in turn feeds the data to the wear model and the KBS. A push-button on the receiver unit enables capacitive sensors to be initially calibrated. The initial length of the sensor (50mm) is assigned to the value of the transmitted counter value. This takes into consideration any small change in sensor capacitance during installation.

Plate 6.5 - Prototype Wear Sensor Receiver

6.6 Designed Smart Wear Sensor Unit

For the final design to be practical, reliability, ease of installation and cost are important factors. The first design consideration is the metal housing for the sensor. This has to be large enough to give support for the sensor when housed in the manganese. However, further consideration must be given to the method of inserting the sensors into the liners. If the holes are too large, the strength of the liners could be compromised, and the difficulty of obtaining the holes increases. Further stability can be increased by introducing a screw thread to the housing and the holes in manganese liner.

The electronics for the smart sensor are to be housed in a sealed capsule on the opposing side of the wearing manganese liner. This will connect to the sensor itself, and can be used again when the liner is worn out. Figure 6.16 shows a schematic of the smart wear sensor prior to installation within the rotating liner.
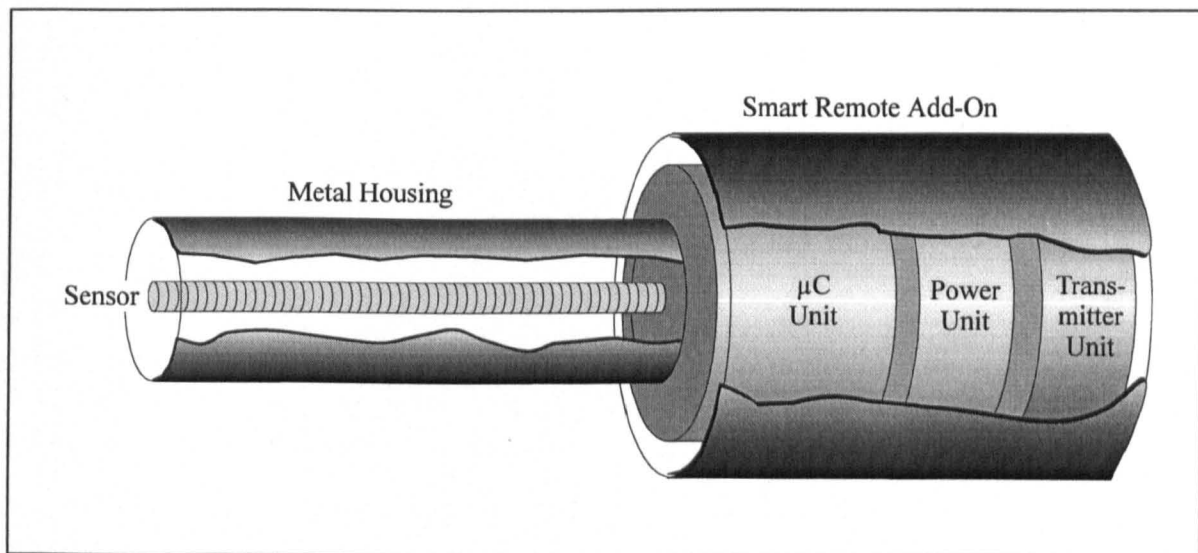
Figure 6.16 - Remote Smart Wear Sensor Design Schematic

The schematic shows the add-on capsule for the remote sensors on the rotating liner. Shown is the microcontroller with integral electronics, the power supply and the transmitter unit. This unit will be of similar structure for the stationary sensors, without the transmitter module. In the case where multiple wear sensors are to be used on the rotating liner, there are two options for the transmitter.

1. A single central transmitter can be used to transmit the information from all of the wear sensors. The sensors will still be 'smart' each with a dedicated microcontroller. These will then be linked via the serial bus, and the transmitter linked to the 'master' remote microcontroller. Transmission of the data will be based on a time share, with each sensor transmitting its data to a single receiver in turn. Software codes can be used to identify the data with the relevant sensor. This option is cheaper, however, wiring is necessary to each smart wear sensor unit.

2. Each smart wear sensor has its own transmitter, and telemetry networking is employed [Long, 1994]. The identity of the wear sensors are hardware or software encoded, with a separate address for each node. Only a single receiver is needed, with knowledge of each of the sensor addresses. This is more costly, however, no wiring between nodes is necessary. As the duty cycle for a single transmitter need only be 1 in 100, a significant number of transmitters can use the same channel.

CHAPTER 7

DISCUSSION

7. DISCUSSION

One of the aims of this project was to develop a series of novel sensors capable of indicating the state of wear within the manganese liners. This information must be readily available to a control system, linked via a serial bus. Using information gained from the sensors as well as a developed wear model for the particular quarry, the amount and type of wear apparent on the liners can be determined, and the discharge setting automatically controlled. The information is distributed to a Knowledge Based System which informs the user as to the state of the crusher during operation. An effective man-machine interface also allows the user to control the crusher and produce an optimum crushing regime.

The sensors and their suitability for this industry are discussed, along with the methods of integration into a quarry. This includes the need for a distributed control and acquisition system and telemetry modules for the rotating wear sensors. The implementation of the KBS is also discussed with the related benefits to the quarrying industry.

It is believed that each of the modules discussed within this thesis would be of a distinct advantage to the quarrying industry, and provide a more accurately controlled and informative system.

7.1 Suitability of the Sensors

In order to produce a sensor which can be used within the quarrying environment, the following constraints need to be adhered to:-

♦ A small diameter for ease of insertion into the liner, including the insert.

♦ Sub-millimetre resolution

♦ High accuracy, with little or no drift due to external effects such as temperature.

♦ Ease of manufacture, with accurate repeatibility

♦ Reliability

With these constraints in mind, the main findings from the laboratory and field tests of the three types of sensor are discussed.

7.1.1 Multiple Surface Mount Resistive Sensor

The laboratory strength tests on the multiple surface mount resistive sensor show that the sensor is reliable up to the compressive strength of mild steel. A 1.55mm calculated variation is noted whilst the sensor is tested under the compressive force.

The sensor exhibits a stability of +0.183mm/-0.135mm over the 2400 one second samples, and a tolerance of +0.109/-0.279mm over the temperature range of 21°C to 70°C. The latter tolerance is within the 200ppm/°C manufacturers claim.

The grinding test proved to highlight the difference in output with relation to the type of grinding method. Whilst under the slow grinding method of the coarse emery paper, a high volume of short circuits appeared across the sensor. When the sensor was applied to a grinding wheel, the short circuits were eliminated. However, throughout the grinding test, the accuracy of the sensor remains high, with a maximum error between the measured and calculated wear of 0.978mm.

Finally, the field tests on the sensor show that the nature of the environment can cause malfunctions within the sensor. These were manifested in the form of open circuits. These could have been due to the vibrations of the crusher breaking the solder joint between the wire and the sensor. The results show a similar wear rate to that predicted from the questionnaire, with the calculated wear rate at 0.51mm/hour and an estimated wear rate of 0.52mm/hour.

7.1.2 Single Resistive Sensor

The single resistive sensor proved stable during the strength test, with a change in the sensor output of 1.82mm.

Stability of the sensor was not problematic in that over the test period of 2400 one second samples, a variation of +0.146/-0.059mm was apparent. During the temperature test, fluctuations of +0.35mm around the 50.5mm sensor length were noticed.

The grind test using a motorised grinding wheel reduced the sensor from 50.5mm to 20.1mm. During this time, the trend of the sensor output followed the physical measured results with an average deviation of 1.891mm. However, instability of the sensor output during the central part of the test was apparent with sudden deviations of up to 17mm. Taking the stable part of the test, an average deviation of 1.122mm was achieved. Problems of instability could be due to the brittle nature of the single resistive sensor, with hairline cracks appearing down the sensor.

The field tests of the single resistive sensor shows definite steps of wear in the liner of up to 10mm. It is unlikely that such large instantaneous wear rates could occur, and that the results of the sensor are questionable. The brittle nature of the sensor is the most likely cause, with a break in the sensor at a point lower than the actual grinding position. Whilst the rock is grinding away at a fragment separated from the rest of the sensor, the reading from the complete part of the sensor will remain constant.

7.1.3 Capacitive Sensor

The implementation of capacitive sensors means a increase in the signal conditioning, yet a sub-millimetre resolution is still possible.

The strength test performed proved the sensor to be unreliable when subjected to a compressive force. This is the case after a relatively low compressive force. It is possible that this could be an important factor when choosing a sensor for a particular task. With this in mind, this type of sensor would prove to be inaccurate when used within the liners of a cone crusher.

A stability of ±0.118mm was recorded over 2000 one second samples. However, when subjected to a temperature test, the instability of a capacitive sensor was shown. Equivalent lengths of 49.5mm to 18.6mm were measured when the sensor was subjected to a temperature range of 22°C to 70°C respectively. This implies that if capacitive sensor is to be used in a wear situation, temperature ranges which the sensor is to be used need to be known, and a feedback system involving temperature probes needs to be implemented. Furthermore, the distribution of heat across the sensor plays an important factor. Surface mount capacitors at the grinding face are likely to be subjected to higher temperatures than those at the base of the sensor, thus different capacitance changes will occur throughout the sensor. This problem of capacitance changing with heat at indeterminate levels throughout the sensor makes the true length of the sensor difficult to calculate.

The grind test proved that wear measurement using a capacitive sensor is problematic. At the start of the grind test, the calculated length followed the measured length. As the test continued, the temperature of the sensor increased and a calculated length was below the measured length by between 5mm and 6mm. At the end of the test, the calculated equivalent sensor length returned from the heat affected measurement to a stable true length equivalent to the measured length. This occurred after a 20 minute cooling down period.

7.1.4 Discrete Level

The discrete level sensor offered a good strength test, with compressive forces up to 325kN before any of the conductive loops failed. Under the grind test, the conductive loops remained intact until breached by the grinding mechanism. At this point, the conductive loop immediately breaks and the true length can be calculated. After this, however, instantaneous contacting between the two loose wires appears, and instability in the results for that particular loop can be seen. This is not a problem, as the other loops are not effected and the initial breaking point is the only result that matters. The accuracy under grind test is 1.1mm worst case, with a resolution of 5mm for this particular sensor. The resolution is determined by the width and length of the sensor, the materials used and the manufacturing method employed.

7.1.5 Summary

Table 7.1 shows a table of the capabilities of each of the sensors tested. In brackets are the high end constraints using off the shelf products.

	Multiple Resistive	Single Resistive	Capacitive	Discrete Level
Resolution	0.55mm (0.35mm[*1])	-	0.6mm (0.5mm[*2])	5mm (0.05mm[*3])
Stability Accuracy	+0.183mm -0.135mm	+0.146mm -0.059mm	+0.1mm -0.135mm	0mm
Strength Test	1.55mm variation at compressive strength	1.82mm variation at compressive strength	Change in output at relatively low compressive force	No failure until a compressive force of 325kN
Temperature Stability	+0.109mm -0.279mm[*4]	+0.350mm	-30.9mm[*5]	0mm
Grind Test Error[*6]	0.978mm Max 0.477mm Ave	1.862mm Max 1.122mm Ave[*7]	9.001mm Max 6.947mm Ave	1.1mm Max 0.5mm Ave
Field Test	Good accuracy but periods of sensor failure	The brittle nature of the sensor proved to be a key factor	N/A	N/A

Table 7.1 - Summary of Sensor Capabilities

[*1] Using a Welwyn WCR0402 Surface mount resistor (4.7Ω to 2.2MΩ)

[*2] Using an AVX Kyocera 0402 Surface mount capacitor (0.5pF to 10nF)

[*3] Based upon work by [Kovacevic, 1991]

[*4] The main results were not generally affected by temperature, and these results show stability accuracy.

[*5] At 70°C.

[*6] Excluding unstable results

[*7] First 10mm of Wear (stable period)

By comparing the 4 types of sensors, obvious advantages and disadvantages are apparent. The resolution offered by the single resistive sensor seems to have a distinct advantage over the other sensors. However, under test, the brittle nature shows it to be inaccurate. This shows up noticeably during the grind and field tests. The problems of the capacitive sensor are initially with the problem of stability under conditions of higher temperature, and compressive forces acting upon the sensor. Greater temperature stability is offered by dielectrics with a lower dielectric constant (COG dielectrics have a dielectric constant K of

between 60 and 80, and typical capacitance change with respect to temperature is in the order of ±30ppm/°C). A low dielectric constant, however, means low capacitances available, thus more involved signal conditioning. The highest grind test accuracy is provided by the multiple resistive sensor, which offers a resolution of 0.5mm (0.35mm possible) and is accurate to within 1mm. The multiple resistive sensor also provides good results during the field test. Finally, the discrete level sensor proved to excel in the strength test. This is probably due to the more compact design of the sensor. Also, the sensor is not susceptible to temperature changes or stability problems. An increase in resolution is possible, however to achieve this throughout a 50mm sensor would mean a great deal of contacts at the base of the sensor.

Laboratory tests can only mimic field tests to a certain extent. Vibration, moisture and temperature are just a few parameters in the conditions to which the sensor is subjected. The sensor implemented must be designed with the field conditions as critical constraints.

7.2 System Development

A modular approach to testing an installation was adopted. These include:-

◆ KBS Development (Stand Alone Model, **Laboratory**)
◆ KBS Development (Distributed Control System, **Laboratory**)
◆ KBS Development (Current PLC System, **Quarry**)
◆ Wear Sensors (Stability, Strength, Temperature, Grinding, Moisture Tests, **Laboratory**)
◆ Wear Sensors (Field Tests, **Quarry**)
◆ Distributed Control & Data Acquisition (4 Stations, **Laboratory**)
◆ Prototype Remote Smart Wear Sensor (**Laboratory**)

7.2.1 KBS Development

The modular development of the Knowledge Based System enabled each system to be tested in laboratory and field conditions and give the user a graphical interface for information and control of the system. An initial prototype was developed using computer

generated data. Rule bases include the Crusher Management (control and data acquisition of the cone crusher), Fault Diagnosis, On-Line Manual and Trends. This was successfully demonstrated at the Hillhead Quarry Exhibition (Hillhead, June 1993), with positive feedback from users from within the industry. A brochure printed for this exhibition is shown in Appendix F.

The modified KBS, installed at Barton-Under-Needwood interfaced the KBS with a duplicate PLC to control the crusher. The rule bases for the on-line manual and fault diagnosis were also installed. This was successfully demonstrated to the mill manager and operators and to a consortium from the DTi and SERC.

A series of rule bases were developed for the laboratory distributed control and acquisition system, which uses embedded microcontrollers for wear sensor data acquisition and cone crusher control. The KBS interfaces to the master microcontroller through an RS232 link, which controls the crusher through a dedicated serial bus system. This KBS sends out control commands and receives data back from the wear sensors and 'dummy' laboratory based sensors.

There are distinct benefits of using a Knowledge Based System for control and condition monitoring of a cone crusher machine.

1. Historical Data - trends can be immediately graphically analysed, and optimum crushing settings can be determined. This will provide quality produce and reduce the need for re-circulation.

2. Graphical Man-Machine Interface - control and analysis of crusher status can be carried out through a single screen.

3. On-Line Manual - A permanent computer copy of the manual is provided.

4. Fault Diagnosis - Faults can be detected quicker and machine downtimes reduced. This also adds awareness of system faults and maintenance procedures to new users.

5. Sensors - All sensor values can be viewed on a single screen at the same time. Additional sensors can be easily integrated into the system without adding an extra dial to the control room. Respective alarms can also provide the user with a greater awareness as to the operation of the crusher.

6. Control - This can be set to automatic or manual. By allowing the KBS to take automatic control, the KBS can force the hydraulic rams on the crusher to produce a constant nip angle, and thus the aggregate diameter, during the lifespan of the manganese liners.

7.2.2 Distributed Control and Data Acquisition

The utilisation of embedded systems for industrial control is increasing with traditional PLC systems not necessarily becoming a first choice option. The flexibility offered by using distributed control and embedded smart sensors for the quarrying industry is higher than current PLC systems.

A laboratory set up included a master to multiple slave set-up using state-of-the-art In Circuit Emulators. Each slave could be configured for control or data acquisition, with its own dedicated address and connection to the serial bus. The program strategy was tested including the slave search polling to check for attached devices. Also implemented was two way passing of information between nodes. Control information was passed from the master to a slave, and specific data was passed back to the master. Data was also passed back to the KBS to inform the user on the state of the system.

7.2.3 Remote Smart Wear Sensor Prototypes

The development of a smart wear sensor prototype incorporating telemetry transmitters and receivers proved the concept of the remote smart wear sensor. Integrated into the smart wear sensor transmitter is a microcontroller with embedded serial bus, signal conditioning for the wear sensor used, battery backed memory for emergency storage of data and telemetry transmission. The receiver has a similar structure, with LCD for stand alone use and RS232 communication to a dedicated KBS.

The prototype proved reliable, with maximum working distances of around 30 metres.

CHAPTER 8

CONCLUSIONS

8. CONCLUSIONS

This work has identified a need for a smart wear measurement and control system in the quarrying industry. The main conclusions are as follows:-

1. Four types of sensor were designed, built and tested. The results of these experiments show the strengths and weaknesses of each design.

   Multiple Surface Mount Resistive Sensor: This sensor performed well in both laboratory and field tests. During the strength test, no catastrophic failures were present at forces up to the compressive strength of the housing material. The stability test and temperature test show the sensor to be stable to within 0.183mm and 0.279mm respectively. The laboratory grinding test show the sensor to be more stable during regular grinding motions such as that of a grinding wheel. The accuracy is to within 1mm. Finally, the field tests show the sensor to give a fair indication of the wear inherent in the liners. There are problems of short circuits and irregular spikes, however, which need to be addressed.

   Single Resistive Element Sensor: The design and production of this sensor potentially offered a sensor with similar properties to the multiple resistive sensor, with less complex manufacturing methods. The sensor, however, proved to be brittle in nature. Further problems of an initial 10% tolerance and non-uniform resistance along the sensor are areas which need to be analysed to develop an accurate sensor. The sensor proved to be fairly reliable under general laboratory tests, however, the grind test and field test show the weaknesses of the sensor. A greater amount of erroneous spikes can be noticed during the grind test, and dramatic changes in length can be noticed during the field test.

   Multiple Surface Mount Capacitive Sensor: Again, the sensor performed well during the stability tests. However, due to the dielectric property of the surface mount capacitive devices, temperature has a profound effect on the sensor readings. The temperature test show the capacitive sensor to vary in length from 49.5mm to 18.6mm over the 22°C to 70°C range. The strength test also provided an equivalent decrease in sensor length as the compressive force increased. This change was noticeable after 2 or

3kN of compressive force. A linear degradation of the equivalent sensor length is visible as the compressive force is increased. The grind test shows that the sensor reading mimics the actual wear, with an offset due to the temperature change. Occasional spikes are noticeable during the test.

Discrete Level Sensor: One of the most desirable features of the discrete level sensor is the diameter of the housing. The disadvantage, is the resolution currently offered. The tests show that the sensor performed well during the strength test, with failures appearing at 325kN. The grind test also shows the sensor to be fairly accurate, with a definitive point at which the conducting wire is broken.

Of all of the sensors, the multiple surface mount resistive is the most reliable. The single resistive element has the advantage of ease of manufacture. However, a great deal of further development is necessary to provide an accurate sensor. In order to produce an accurate capacitive sensor, a much lower dielectric would be necessary with signal conditioning which can handle low capacitances. Finally, by investigating improved manufacturing techniques, the discrete level sensor has potential to be a highly accurate and reliable sensor.

2. The infrastructure of quarry control systems would be greatly improved by the introduction of a distributed control and acquisition system. The flexibility would enable greater expansion possibilities as the quarry develops.

3. A series of Knowledge Based Systems have been developed and tested. A laboratory KBS using simulated data was first demonstrated. This includes control and acquisition, fault diagnosis and on-line help rule bases. A second KBS was developed to integrate to the laboratory based distributed control and acquisition system. Finally, a field test involving a modified rule base linked the KBS to the existing control strategy. Communication to the crusher PLC routed the control commands and acquisition to the KBS rule base. This proved effective in control and data acquisition and provided the user with more information with regard to the state of the crusher. The quarry in which the KBS was installed was intended as a beta test site, and due to time constraints and problems involved with installations within a working quarry, a quantitative study on the benefits of the KBS could not be made.

4. A distributed control structure has been developed using the NEC 78330 microcontrollers. This is designed with data acquisition and control nodes linked via a two wire serial bus. A developed prototype using a dedicated KBS, master and four slaves has been laboratory tested. The bus system proved to be effective and reliable, offering a low-cost local bus solution.

5. The need for radio telemetry on the rotating manganese liner has been identified. This is in conjunction with the installation of wear sensors on the eccentrically rotating mantle. Standards for using telemetry systems have been investigated, with the recommendations as laid down by the Radiocommunications Agency.

6. A prototype remote smart wear sensor has been designed, built and developed. The prototype integrates signal conditioning, microcontroller, RAM, ROM, battery power and transmit/receive. This demonstrates effective transmission up to 30 metres. Further enhancements to miniaturise this device have been investigated, with power considerations at the forefront of the design.

7. The use of the smart wear sensors is not necessarily restricted to the quarrying industry. This thesis has investigated further areas in which the wear in critical situation could be analysed by the developed wear sensors. These include the automotive and drilling industries.

CHAPTER 9
FURTHER RESEARCH

9. FURTHER RESEARCH

9.1 Sensors

Further sensor development is necessary before installation into a control strategy. Further research for the four sensors are as follows

1. Multiple Resistive: Further field tests, removal of short circuits, increased resolution using smaller surface mount devices.

2. Single Resistive: Improvement of the brittle structure of the sensor, improved tolerance.

3. Multiple Capacitive: Improve temperature effects on the sensor by using lower dielectric capacitors, increased resolution using smaller surface mount devices, field tests. A greater variety of surface mount capacitors is now readily available [AVX, 1996] which boast a high capacitance value with reduced component size and increased temperature and voltage stability. The problem of the reaction of the sensor due to a compressive strength needs to be analysed.

4. Discrete Level Sensor: Improve resolution by introducing more conductive loops, field tests.

Further research into the manufacturing method employed to produce the sensors is also important to provide an accurate resolution. This is extremely important for the resistive and capacitive stacks, as alignment is of paramount importance. This is also true for the discrete level sensor, with positional accuracy important for the tip of the conductive loop.

Further field tests are currently under development in Brisbane, Australia. A need for this type of sensor has been identified, and a sample of wear sensors have been sent out for field trials.

9.2 Smart Wear Sensor Development

Miniaturisation is the key factor for the smart wear sensor. By developing a microcontroller with integral signal conditioning as well as RAM, ROM and serial bus will reduce the overall size of the smart wear sensor. Power considerations are always

necessary, and the power saving techniques addressed in Chapter 6 need to be adopted. The life span of the manganese liner used will determine the necessary power. Enclosures for the smart wear sensor need to be examined, with vibrational stability and safety regulations important factors in the design.

9.3 Developed Control and Data Acquisition System

Further system development includes the installation of the infrastructure into a fully working quarry. Sensor and control nodes can then be integrated into the system.

9.4 Knowledge Based System

Several modules of the Knowledge Based System have been tested. The laboratory based KBS proved successful with prospective users at the Hillhead exhibition. The use of the specifically developed system for the Barton-under-Needwood quarry proved successful in control of the crusher. One of the problems, however, is producing a generic KBS which will run at a quarry. A multitude of different systems are implemented at each quarry with extras often provided for the standard Pegson systems. With this in mind, further development work for the KBS would be to produce a system configuration rule base which analyses the needs of the quarry.

Finally, further work should endeavour to provide full KBS integration with the distributed control and data acquisition system at a working quarry.

REFERENCES

Akin T., Ziaie B., Najafi K. (1990) RF Telemetry Powering & Control of Hermetically Sealed Integrated Sensors and Actuators. Technical Digest. IEEE Solid State & Actuator Workshop pp145-8

Anon (1990) High Tech Operations - The Manless Mill, Northern Miner Magazine Vol 5 No 5, May 1990, pp22-23

Aoyama H., Kishinami T., Saito K. (1987) A Study on a Throw Away tool Equipped With a Sensor to Detect Flank Wear. Bulletin of Japan Society for Precision Engineering Vol 21 Part 3 pp203-8

Armstrong J. (1986) Condition Monitoring - An Introduction to its Application in Building Services. BSRIA Technical Note TN 1/86

AVX (1996) AVX Offers Unique Capacitor Solution. Components In Electronics. p48

Barker R.W., Hinich M.J. (1994) Statistical Monitoring of Rotating Machinery by Cumulant Spectral Analysis. SPIE Proceedings Advanced Signal Processing: Algorithms Architectures and Implementations. Vol.2296 Pt 4 pp43-51

Batanov D., Nagarur N., Nitikhunkasem P. (1993) EXPERT-MM - A Knowledge Based System for Maintenance Management. Artificial Intelligence in Engineering Vol 8 No 4 pp.283-291

Bearman R.A., Barley R.W. and Hitchcock A. (1990) The Development of a Comminution Index for Rock and the Use of an Expert System to Assist the Engineerin Predicting Crushing Requirements. Minerals Engineering Vol. 3, No. 1/2, pp117-127

Bearman R.A. (1991). The Application of Rock Mechanics Parameters to the Prediction of Crusher Performance PhD Thesis Camborne School of Mines

Bearman R. A., Parkin R. M. (1992a) Intelligent Control of Cone Crushers. Proceedings IMechE: Mechatronics, The Integration of Engineering Design. University of Dundee. IMechE MEP ISBN 0 85298 840 0, pp151-7

Bearman R. A., Parkin R. (1992b) The Use of Knowledge Based Systems in the Control of Rock Crushers. Proceedings, BCS SGES Expert Systems 92. Churchill College Cambridge.

Black P. (1994). Troubleshooting with LANs and Smart sensors Intech May vol41 pt5 pp29-31

Brinksmeier E. (1990) Prediction of Tool Fracture in Drilling <u>Annals of the CIRP</u> Vol 39/1/1990 pp97-100

Brown J. (1990) Magnetic Sensors measure shaft torque without contact <u>Power Transmission design</u> vol32 pt9 pp39-40

Bryzek J. (1993). Smarter, less-costly sensors are on the way. <u>Instrumentation and Control Systems</u> Oct Vol66 pt10 pp85-7

Burkhart A. (1991). Recent Developments in Flip Chip Technology. <u>Surface Mount Technology</u>, Vol 5, part 7 ISSN 0 893 3588, pp40-42

Case T. (1996) Flaw Identification from Time and Frequency Features in Ultrasonic Waveforms. <u>IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control</u> Vol 43 No 4. pp592-600

Charbonneau L, Plouffe P (1993) SENSODEC-10 Condition Monitoring-System. <u>Pulp & Paper-Canada,</u> Vol.94, No.12, pp.161-164 IS: 0316-4004

Chowanietz, E (1995) Automobile Electronics. ISBN 1 56091 739 3

Chutatape O, (1989) A Design of a Large Scale Telemetry Network for the Monitoring of Remote Refrigerated Containers. <u>TENCON 89. Fourth IEEE Region 10 International Conference 'Information Technologies for the '90's'</u> p604-607

Cook T.A., Fernald K.W., Miller T.K., Paulos J.J. (1990). A Custom Microprocessor for Implantable Telemetry Systems. <u>Proceedings of the Third Annual Symposium on Computer Based Medical Systems</u> pp412-417.

Dargie P.G., Hughes S.T. (1994) A Thick Film Capacitive Differential Pressure Transducer. <u>Meas. Sci Technology</u> Vol 5.. pp1216-1220

Das S, Roy R, Chattopadhyay A.B (1996) Evaluation Of Wear Of Turning Carbide Inserts Using Neural Networks. <u>International Journal Of Machine Tools & Manufacture,</u> Vol.36, No.7, pp.789-797 IS: 0890-6955

Dauw D. (1986) On the Derivation and Application of a Real-Time Tool Wear Sensor in EDM. <u>Annals of the CIRP - Electro Physical</u> Vol 35 Pt 1 pp111-6

de Jong G., Meijer G.C.M, van der Lingen K., Spronck J.W., Aalsma A., Bertels A.J.M (1994) A Smart Capacitive Absolute Angular Position Sensor. <u>Sensors and Actuators A,</u> 41-42 pp212-6

DiGuinta R., Goglio L, Gola M.M. (1996) How Calcereous Layers Affect Ultrasonic Thickness Gaging. <u>Materials Science Forum</u> Vol 210 No Pt1 and 2, pp349-55

Dornfeld D.A., (1990). Neural Network Sensor Fusion for Tool Condition Monitoring. Annals of the CIRP. Vol 39 pt 1 pp101-5.

Drake P.R, Jennings A.D, Grosvenor R.I (1996) A Data-Acquisition System For Machine-Tool Condition Monitoring. Quality And Reliability Engineering International, Vol.12, No.1, pp.19-26 IS: 0748-8017

DTI (Department of Trade and Industry publication) (1990) Fault Diagnosis and Recovery: Complex Machinery - British Coal SHEARER Application, HMSO

Echelon Corporation (1990) LONworks Product Line Brief. Motorola Semiconductor Technical Data.

Edwards T. Innovative Developments in Smart Sensors. Sensor Review. Vol.13 No.2 pp8-10

Fernald K.W., Stackhouse B.A., Paulos J.J., Miller T.K. (1989). A System Architecture for Intelligent Implantable Biotelemetry Instruments. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society. November Vol 11 pp1411-12.

Fowler K.A., Elfbaum G.M., Smith K.A., Nelligan T.J. (1996) Theory and Application of Precision Ultrasonic Thickness Gauging. Insight, Vol 38 No 8 pp582-87

Ghibu C. and Dupuis D. (1987) Development of an Expert System to Assist the Operator of a Modern Electrolytic Zinc Refinery: Preliminary Results. Proc. of the CANMET Symposium on Expert Systems in the Minerals Industry. Dept. of Energy, Mines and Resources, CANMET, SP87-4E, March 1987

Goldman M.A., Groves S.E., Sibigtroth J.M. (1990) The Role of Microprocessors in Future Automotive Electronics. Vehicle Electronics in the 90's: Proceedings of the International Congress on Transportation electronics October. pp121-30

Gomez F., Olivares G., Prieto A., Gazquez J.A. (1991) SPIRAL: A Microprocessor Based Telemetry and Telecontrol System. Microcomputer Applications, vol 10 iss 1 ISSN 0820 0750 pp22-8

Grimmelius H.T, Woud J.K, Been G (1995). Online Failure Diagnosis For Compression Refrigeration Plants. International Journal Of Refrigeration-Revue Internationale Du Froid, Vol.18, No.1, pp.31-41 IS: 0140-7007

Hales L.B. (1989) Supervisory Control of Flotation Circuits, Advances in Coal and Mineral Processing Using Flotation, Ed: Chander S. and Klimpel R.R., SME, pp298-302

Heerens W.C. (1986) Application of Capacitance techniques in sensor design. Journal of Physics E vol 19, pp-897-906

Heintz F., Zabler E. (1989), Application Possibilities and Future Chances of "Smart" Sensors in the Motor Vehicle. International Congress & Exposition, Sensors and Actuators Feb27-Mar3, pp39-50

Hills P.W (1996) Production Benefits From A Vibration-Based Condition Monitoring Program. Insight, Vol.38, No.8, pp.563-565 IS: 1354-2575

Hinch S.W. (1988), Handbook of Surface Mount Technology, Longman Scientific & Technical ISBN 0-582-00517-5

Hogenbirk E.J., Verhoeven H-J., Huijsing J.H. (1995) An Integrated Smart Sensor for Flow and Temperature with I^2C Bus Interface: FTS2. IEEE International Symposium on Circuits and Systems pp2225-28

Hughes S.T. (1995) Sensors for Road Vehicle based upon Capacitance Variation. Autotech 95, Institute of Mechanical Engineers

Huijsing J.H., Riedijk F.R., van der Horn G. (1994) Developments in Integrated Smart sensors. Sensors and Actuators A (Physical) May volA43 pt1-3 pp276-88

Irish J.D., Needell G.J., Morey K., Wood J., Baldwin K.C. (1990) A Current Meter with Intelligent Data System, Environmental Sensors and Real Time Telemetry. Proceedings of the IEEE Fourth Working Conference on Current Measurement pp73-9

Iyengar S.S., Prasad L., Min H. (1995) Advances in Distributed Sensor Technology. ISBN 0-13-360033-5

Jones J. (1992) Can Fieldbus Survive. Control & Instrumentation August pp25-6.

Kakade S, Vijayaraghavan L, Krishnamurthy R (1994) In-Process Tool Wear And Chip-Form Monitoring In Face Milling Operation Using Acoustic-Emission, Journal Of Materials Processing Technology, Vol.44, No.3-4, pp.207-214 IS: 0924-0136

Klein B., Fischer J. (1995) Brake Lining Wear Control in a Motor Vehicle Brake System. UK Pat. No. 2266568B. 9/8/95

Knights M. (1993). Radio Telemetry. Control & Instrumentation June pp18-9

Ko T.J., Cho D.W., Lee J.M. (1992) Fuzzy Pattern Recognition for Tool Wear Monitoring in Diamond Turning. Annals of the CIRP Vol 41/1 pp125-8

Kovacevic R., Evizi M. (1990) Nozzle Wear Detection in Abrasive Waterjet Cutting Systems. Materials Evaluation, Vol 48, pp348-53

Kovacevic R. (1991) A New Sensing System to Monitor Abrasive Waterjet Nozzle Wear in: Computer-Aided Production Engineering, ed. V.C Venkatesh and J.A. McGeouch. Elsevier Publications.

Leonard M. (1993), Creating the Truly Smart Sensor. <u>Electronic Design</u>. Vol 41. Part 8 p18 (66) TI:

Li S, Elbestawi M.A (1996) Tool condition monitoring in machining by fuzzy neural networks. <u>Journal Of Dynamic Systems Measurement And Control-Transactions Of The ASME</u>, Vol.118, No.4, pp.665-672 IS: 0022-0434

Lion K.S. (1969), Transducers, Problems and Prospects. <u>IEEE Trans. Industr. Electron. Contr. Intrum.</u> Vol IECI-16, p2

Lizheng X., Jiayu C., Dingzhong H. (1990), Rotating Measurement and Data Processing System for a Superconducting Turbine Generator. <u>Cryogenics</u> Vol 30 iss supplement. ISSN 0011 2275, pp739-742

Long N.R.W., (1994), Radio Telemetry. <u>Sensor Review.</u> Vol 14 No 2, pp10-15

Malcovati P., Azerado Leme C., O'Leary P., Maloberti F., Baltes H. (1994) Smart sensor interface with A/D conversion and Programmable calibration. <u>IEEE Journal of solid state circuits</u> Vol29 Iss 8 pp963-6

Martin K. (1981) A voltage controlled switched capacitor relaxation oscillator. <u>IEEE Journal of Solid State Circuits SC-16,</u> No 4 Aug

Mechefske C.K, Stephens M.J, Turner G.A, Macdonald J.A, Palylyk R.A, Pollock B, Franklin D.E (1994) A Comprehensive Condition Monitoring-System For Hydroelectric Generating-Units. <u>Insight</u> , Vol.36, No.4, pp.238-244 IS: 0007-1137

Meech J.A. (1990) Expert Systems for Teaching and Training in the Minerals Industry, <u>Minerals Engineering</u>, Vol 3, No 1/2, Proc. of the International Comminution Sym. Pergamon Press

Middelhoek S. (1994) Quo Vadis Silicon Sensors? <u>Sensors and Actuators A,</u> 41-42 pp1-8

MIRC (1991), World Smart Sensor Markets, Mountain View

Moshgbar M., Parkin R., Bearman R. (1995a) The Impact of System Tribology on the Automatic Control of Cone Crushers. <u>Transactions of the Institute of Measurement and Control.</u> Vol 17 Pt 1, pp2-9

Moshgbar M., Parkin R., Bearman R. (1995b) Optimal Control of Cone Crushers Utilising An Adaptive Strategy for Wear Compensation. <u>Mineral Engineering</u> Vol 8 Pt 4-5 pp367-76

Moshgbar M., Parkin R.M., Bearman R.A. (1995c) Adaptive and Intelligent Control of Cone Crushers. _APCOM XXV, Application of Computers and Operations Research in the Minerals Industries_, Brisbane, Australia, 10-14 July

Mutagwaba W.K., Terezopoulos N.G. (1994) Knowledge Based System for Mine Method Selection. _Transactions of the Institute of Mining and Metallurgy, Section A - Mining Industry_ Vol 103, ppA27 - A32

Olbrich T., Bradley D., Richardson A. (1994) Self-test for smart sensors in automotive applications. _IEE Colloquium on Automotive Sensors_ pp 3/1-4

Ormond T. (1993) Smart Sensors tackle tough environments. _EDN (European edition)_ October vol38 pt 2 pp35-42

Peck J.P, Burrows J (1994) Online Condition Monitoring Of Rotating Equipment Using Neural Networks. _ISA Transactions_, Vol.33, No.2, pp.159-164 IS: 0019-0578

Phillips Components. (1990) I^2C - Bus Compatible IC's.

Popovics J.S., Rose J. L (1994) A Survey of Devlopments in Ultrasonic NDE of Concrete. _IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control_ Vol 41 No 1 pp140-43

Porter R. (1997) Ultrasonic Thickness Surveys on Tankers and Bulk Carriers. _Insight_, Vol39 No 1 pp21-23

Powner E.T., Yalcinkaya F., (1995) Intelligent Sensors: Structure and System. _Sensor Review._ Vol 15 No. 3. pp31-35

Puers B., Lapadatu D. (1994) Extremely Miniaturized Capacitive Movement Sensors Using New Suspension Systems. _Sensors and Actuators A,_ 41-42 pp129-35

Quantelec (1992a). TXM-418 Data Sheet. Issue 4. May pp1-8

Quantelec (1992b). RXM-418 Data Sheet. Issue 3. February pp1-9

Quin Systems. (1991) ITV Uses Intelligent Bitbus Modules For A Clearer Picture. _Control & Instrumentation_ December pp21

Reeve A. (1993) Fieldbus - Are Users Involved. _Control & Instrumentation_, August pp25-6

Roark R.J. (1965) Formulas for Stress and Strain

Schrader (1995) Remote Tire Pressure Monitoring. _Sales Brochure_

Serra P, Sanchez M, Lafuente J, Cortes U, Poch M (1997) Iscwap - A Knowledge-Based System For Supervising Activated- Sludge Processes. _Computers & Chemical Engineering_ , Vol.21, No.2, pp.211-221 IS: 0098-1354

Shiffbauer W.H. (1991) A Flexible Control, Communications and Data Collection Network for Mining Machines. Proceedings of the IEEE International Symposium on Intelligent Control. ISBN 0 7803 0106 4, pp353-8

Smith G.B. (1993) Smart Sensors for the Automotive Industry. IEE Colloquiem on ASICS for Measurement Systems. p6/1-3

Smith K. (1997) Thickness Gauge Enquiry. Panametrics. E-Mailed Message 21/3/97

Spring R. and Edwards R. (1989) Real Time Expert System Control of the Brenda Mines Grinding Circuit. AIME Pre-print No 89-58, Presentes at the SME Annual Meeting, Las Vegas, Nevada, Feb 27th - March 2nd 1989

Startfield A.M, Butala K.L, England M.M. and Smith K.A., (1983) Mastering Engineering Concepts by Building an Expert System. Eng. Education, November 1983, pp 104-7.

Strock O.J. (1983). Telemetry Computer Systems - An Introduction. ISBN 0-13-902487-5. (3) TI:

Swindells R.J, Kent N.W, Cullen R.N, Johnson B.A (1994) Condition Monitoring At Bundaberg-Refinery. International Sugar Journal, Vol.96, No.1142, pp.63-68 IS: 0020-8841

TESTEM Measuring Systems. (1992) Complete Solutions for Every Problem. Sales Brochure.

Toth F.N., Meijer G.C.M. (1992), A Low Cost, Accurate and Fast Capacitive Displacement Sensor with Microcontroller Interfacing. Conference Record IEEE Instrumentation and Measurement Technology. 12-14 May pp414-7

Uusijärvi R.W., Törngren M.E. (1994) Introducing Distributed Control in Mobile Machines based on Hydraulic Actuators. MECHATRONICS, Vol.4, No.2, pp.139-157

Vanderstichelen J., (1987) Expert Systems Prototype for the Process Diagnosis in the Clinker Grinding Circuit of a Cement Factory. Proc. of the CANMET Symposium on Expert Systems in the Minerals Industry, Dept. of Energy, Mines and Resources, Ottowa SP87-4E

Vasilow T.R., Busovne B.J., Johnson G.J. (1986). Resistive Ceramic Bushings for Brake Lining Wear Sensors. U.S. Pat. No. 4606434. Aug 19

Wilson P.D., Hopkins S.P., Spraggs R.S., Robinson K. (1995), Applications of a Universal Sensor Interface Chip in Smart Sensor Systems. Sensors Series, pp357-64

Wimmer W. (1990) Sensor for Measuring Drum Brake Shoe Wear. German Pat. No. 3638152. 8/2/90

Yellowley I., Hosepyan Y. (1991) Tool Wear Detector. United States Patent Number 5000036 March 19 1991.

Zabler E., Heintz F., Dietz R., Gerlach (1992) Mechatronic sensors in integrated vehicle architecture. Sensors and Actuators A31 54-59

Zhang Q., Litchfield J.B., Reid J.F. (1994), Smart Sensor Technologies for Food and Bioprocess Automation. Food Processing Automation III, Vol 94 Pt 1, pp289-96

APPENDIX A
DISTRIBUTED PROCESSING SOFTWARE

APPENDIX A - DISTRIBUTED PROCESSING SOFTWARE

```
/*********************************************************************/
/*                                                                   */
/*            Master      v7.01                                      */
/*                                                                   */
/*********************************************************************/
/*                                                                   */
/* Filename: MASTERE.C                                               */
/* Author   : S.Yaxley                                               */
/* Date     : 29/9/95                                                */
/* Project  : Cone Crusher Control - OPTIMAX+                        */
/*                                                                   */
/*                                                                   */
/* De Montfort University                                            */
/*                                                                   */
/*********************************************************************/

/*****************************/
/* INCLUDE FILES         */
/*****************************/

#include <delay.h>                   /* wait & delay procedures <own>      */
#include <stdio.h>                   /* standard I/O                       */
#include <io78330.h>                 /* address locations for flags        */
#include <display.h>                 /* lcd control 16x2 NEClab board <own>*/
#include <serial.h>                  /* defined flags <own>                */
#include <stdlib.h>                  /* standard library                   */


/*****************************/
/* ADDRESSES             */
/*****************************/

#define WEAR_ADDRESS     0x60        /* wear slave     */
#define CONTROL_ADDRESS 0x70         /* control slave  */


/*****************************/
/* COMMANDS              */
/*****************************/

#define SLAVE_CHECK_COMMAND 0xa0
#define HYD_PRESSURE_COMMAND 0xa1
#define MOTOR_CURRENT_COMMAND 0xa2
#define LUBE_TEMP_COMMAND 0xa3
#define HYD_TEMP_COMMAND 0xa4

#define INLET_OIL_COMMAND 0xb0
#define OUTLET_OIL_COMMAND 0xb1
#define FLOAT_SWITCH_COMMAND 0xb2
#define TRAMP_METAL_COMMAND 0xb3
#define CONVEYOR_COMMAND 0xb4

#define PRESSURE_ALARM_COMMAND 0xc1
#define MOTOR_ALARM_COMMAND 0xc2
#define LUBE_ALARM_COMMAND 0xc3
#define HYD_ALARM_COMMAND  0xc4


/*****************************/
/* PROTOTYPE PROCEDURES      */
/*****************************/

void reset232(void);
void SB0_inis(void);                 /* SB0 initialization processing       */
void RS232_inis(void);               /* RS232 initialisation processing     */
void timer_set(void);                /* set time out period for Acknowledge */
void ACKD_wait(void);                /* wait for Acknowledge from slave     */
void address_pulses(void);           /* pre process SB0 line for address    */
void command_pulses(void);           /* pre process SB0 line for command    */
void slave_search(void);             /* boot up & search for any slaves     */
void reset(void);                    /* reset serail bus for next ADDR/COMM  */
```

```
void internal_clock(void);           /* set internal clock, master to slave    */
void external_clock(void);           /* set external clock, slave to master     */
void transmit(void);                 /* set transmit flag ON, receive flag OFF */
void receive(void);                  /* set transmit flag OFF, receive flag ON */
void error(void);                    /* error in RS232 reception                */
void slave_info(void);               /* find amount of slaves on bus            */
void slave_addr(void);               /* find slave addresses                    */
void LUT(int);                       /* look up table for relation of addresses*/
void send_cone_variable(int,int);    /* send any cone variable to KBS           */
void send_cone_parameter(int,int);   /* send parameters to slave               */
void wait_for_echo(int);

/******************************/
/* GLOBAL VARIABLES           */
/******************************/

char count,                          /* amount of slaves on serial bus          */
     cone_var,
     slaves[254],                    /* array of slave addresses                */
     slave_check_val,
     temp,
     temp_sio;

int  controller_option,              /* check for valid control address rcvd    */
     wear_option,                    /* check for valid wear address rcvd       */
     lube_alarm,
     hyd_alarm,
     motor_alarm,
     test_var,
     pressure_alarm,
     kbs_option;

/******************************/
/* FLAGS                      */
/******************************/

char RS232_int_occurred;             /* RS232 interrupt occurred                */
char ERRORFLG;                       /* Error flag              */
char DATA_WAIT_FLAG;                 /* wait for data from slave */
char ADDRESS_GONE;                   /* address sent            */
char COMMAND_GONE;                   /* command sent            */
char SLAVE_WAIT_FLAG;
char VARIABLE_WAIT_FLAG;
char AUTO_FLAG;

/******************************/
/* INTERRUPT SERVICE ROUTINES */
/******************************/

/*******************************************/
interrupt [INTCM20_vect] void time_out(void)    /* time out interrupt    */
/*******************************************/
{
ERRORFLG=TRUE;                       /* error if time out has occurred */
ACKT=1;                              /* flag own master acknowledge    */
BSYE=0;                              /* reset busy flag                */

TMC1.3=0;                            /* stop timer                     */
}

/*******************************************/
interrupt [INTCSI_vect] void tx_end(void)       /* serial bus interrupt   */
/*******************************************/
{
if ((ADDRESS_GONE==TRUE)&&(COMMAND_GONE==TRUE)&&(AUTO_FLAG==TRUE))
       {
       cone_var=SIO;
       VARIABLE_WAIT_FLAG=TRUE;
       SLAVE_WAIT_FLAG=TRUE;
       }
}


/*******************************************/
interrupt[INTSR_vect] void receiving(void)          /* RS232 interrupt    */
```

```
/*******************************************/
{
disable_interrupt();

kbs_option = RXB;                       /* set incoming port value to kbs_option */
RS232_int_occurred=TRUE;                /* interrupt has occured                 */

enable_interrupt();
}


/*********************************************/
interrupt[INTSER_vect] void error1(void)          /* RS232 error           */
/*********************************************/
{
disable_interrupt();

enable_interrupt();
}


/*********************************************/
interrupt[INTST_vect] void transmission(void)
/*********************************************/
{
disable_interrupt();

enable_interrupt();
}

/*--------------------------------------------------------------*/
/******************** END OF INTERRUPTS ******************/
/*--------------------------------------------------------------*/



/*******************************************************************/
void main(void)
/*******************************************************************/
{
reset232();

SB0_inis();                             /* SB0 Initialisation                   */
RS232_inis();                           /* RS232 initialisation                 */
timer_set();                            /* set time out period                  */
init_lcd();                             /* initialise LCD                       */

SLAVE_WAIT_FLAG=FALSE;
VARIABLE_WAIT_FLAG=FALSE;
AUTO_FLAG=FALSE;

test_var=0;

enable_interrupt();

wait(4000);            /* on boot up, makes sure all slaves are settled    */

for(;;)                        /* continuous loop                 */
    {
    if ((RS232_int_occurred==TRUE)&&(kbs_option=='1'))/* if kbs sends an option */
            {
            AUTO_FLAG=TRUE;
            slave_search();                /* search for slave devices        */
            slave_info();                  /* transmit to KBS number of slaves */
            slave_addr();                  /* transmit slave addresses */
            test_var++;
            if (controller_option==TRUE)
                    {
                    send_cone_variable(CONTROL_ADDRESS,HYD_PRESSURE_COMMAND);
                    send_cone_variable(CONTROL_ADDRESS,MOTOR_CURRENT_COMMAND);
                    send_cone_variable(CONTROL_ADDRESS,LUBE_TEMP_COMMAND);
                    send_cone_variable(CONTROL_ADDRESS,HYD_TEMP_COMMAND);

                    send_cone_variable(CONTROL_ADDRESS,INLET_OIL_COMMAND);
                    send_cone_variable(CONTROL_ADDRESS,OUTLET_OIL_COMMAND);
```

```c
                            send_cone_variable(CONTROL_ADDRESS,FLOAT_SWITCH_COMMAND);
                            send_cone_variable(CONTROL_ADDRESS,TRAMP_METAL_COMMAND);
                            send_cone_variable(CONTROL_ADDRESS,CONVEYOR_COMMAND);
                            }
                    reset232();
                    }
            if ((RS232_int_occurred==TRUE)&&(kbs_option=='2'))/* if kbs sends an option  */
                    {
                    reset232();

                    send_cone_parameter(CONTROL_ADDRESS,PRESSURE_ALARM_COMMAND);
                    send_cone_parameter(CONTROL_ADDRESS,MOTOR_ALARM_COMMAND);
                    send_cone_parameter(CONTROL_ADDRESS,LUBE_ALARM_COMMAND);
                    send_cone_parameter(CONTROL_ADDRESS,HYD_ALARM_COMMAND);

                    reset232();
                    }
            }
}


/*********************/
void reset232(void)
/*********************/
{
count=0;                            /* reset amount of slaves counter  */
RS232_int_occurred=FALSE;           /* reset RS232 interrupt flag      */
controller_option=FALSE;            /* reset controller slave check    */
wear_option=FALSE;                  /* reset wear slave check flag     */
ERRORFLG=FALSE;                     /* reset time out error flag       */
ADDRESS_GONE=FALSE;                 /* reset ADDRESS GONE FLAG         */
COMMAND_GONE=FALSE;                 /* reset COMMAND GONE FLAG         */
AUTO_FLAG=FALSE;
}


/*********************/
void SB0_inis(void)      /* serial bus initialization processing    */
/*********************/
{
MK1L = MK1L & 0xfc;          /* enable 1111 1100 INTCSI enabled      */
CSIM = 0x0a;                 /* clock = fCLK / 32!!!!, SBI-mode      */
CSIM = CSIM | 0x80;          /* transmit / receive                  */
SBIC = 0x00;                 /* !bus release signal = output RELT = 1 */
PMC3 = 0x1c;                 /* SB0 SB1 AND !SCK                     */
}


/*********************/
void RS232_inis(void)    /* RS232 initialization processing         */
/*********************/
{
int brgg;                        /* baud rate generator             */

MK0H = MK0H & 0x37;              /* enable 0011 0111 INTSR,INTSER,timer */
PMC3=PMC3|0x03;
ASIM = 0xf8;                     /* even_parity, 8bits, 1stopbit, int baudrate */
brgg = 103;
BRG = brgg;                      /* BAUD RATE = 2400, page 6-109        */
BRGM = BRGM | 0x88;
}


/*********************/
void slave_search(void) /* search for slaves on the bus            */
/*********************/
{
int address=0;                      /* reset loop variable         */

count=0;                            /* reset slave count variable   */

SLAVE_WAIT_FLAG=FALSE;
for(address=0;address<=0xff;address++)
        {

        ERRORFLG=0;                 /* reset ERRORFLG               */

        reset();                    /* reset bus                     */
        internal_clock();
```

```
        transmit();

        address_pulses();              /* pre process SB0 line, address*/
        SIO = address;                 /* send loop address              */

        ACKD_wait();                   /* wait for acknowledge from a slave */

        if  (ERRORFLG!=1)              /* if slave found                 */
                ADDRESS_GONE=TRUE;

        ERRORFLG=FALSE;

        if (ADDRESS_GONE==TRUE)
                {
                reset();
                internal_clock();
                transmit();
                        do
                                {
                                command_pulses();       /* pre processing command pulses*/
                                SIO = SLAVE_CHECK_COMMAND; /* send slave check cmd   */
                                ERRORFLG=FALSE;         /* reset ERRORFLG         */

                                ACKD_wait();            /* wait for acknowledge   */
                                }
                        while (ERRORFLG==TRUE);     /* repeat until cmd sent s.fully */
        COMMAND_GONE=TRUE;                          /* command gone flag ON        */

                receive();                          /* rx on, tx off           */
                external_clock();                   /* slave-master, external clock */

                do
                {}
                while (P3.2==0);                    /* wait while SB0 pin is ready */

                temp=SIO;            /* needed to initialise the transfer of data */

                do
                {}
                while (SLAVE_WAIT_FLAG==FALSE);

                SLAVE_WAIT_FLAG=FALSE;

                ACKT=1;

                slaves[count]=address;      /* save slave value in array    */
                count++;                            /* increment slave counter      */
                LUT(address);                       /* look up to see type of slave */

                ADDRESS_GONE=FALSE;
                COMMAND_GONE=FALSE;
                }
        }
ERRORFLG=0;                                     /* reset ERRORFLG                 */
}


/*******************/
void LUT(int slave_address)    /* Look Up Table for Slave duties         */
/*******************/
{
switch (slave_address)
        {
        case WEAR_ADDRESS:wear_option=TRUE;break;
        case CONTROL_ADDRESS:controller_option=TRUE;break;
        }
}


/*******************/
void slave_info(void)     /* send RS232 the amount of slave devices */
/*******************/
{
TXS = count;
wait_for_echo(count);
}
```

```
/********************/
void slave_addr(void)
/********************/    /* send RS232 the slave addesses          */
{
char loop;

for(loop=0;loop<count;loop++)
        {
        TXS=slaves[loop];
        wait_for_echo(slaves[loop]);
        }
}

/********************/
void timer_set(void)    /* set time out period for Acknowledge    */
/********************/
{
TUM0.2=1;                                      /* overflow                      */
TMC1.2=1;                                      /* interval timer mode           */
CM20=200;                                      /* 40 gives time out on analyser*/
}                                              /* 15360 = 30.72ms               */

/********************/
void ACKD_wait(void)    /* wait for Acknowledge from slave         */
/********************/
{
do
        {
        TMC1.3=1;                              /* start timer                   */
        }
while(ACKD==0);                                /* No ACKD found                 */

TMC1.3=0;                                      /* timer is cleared & stopped    */
}

/********************/
void address_pulses(void) /* pre process SB0 line for address      */
/********************/
{
CMDT = 1;                                      /* CMDT SET                      */
RELT = 1;                                      /* RELT SET                      */
CMDT = 1;                                      /* CMDT SET                      */
}

/********************/
void command_pulses(void) /* pre process SB0 line for command      */
/********************/
{
RELT = 0;                                      /* RELT RESET                    */
CMDT = 1;                                      /* CMDT SET                      */
}

/********************/
void reset(void)           /* reset SB0 line                        */
/********************/
{
int clearval;                                  /* variable to clear tx & rx pins */

CTxE=0;                                        /* reset tx & rx                 */
CRxE=0;
P3.2=clearval;
P3.4=clearval;
/*SIO=0xff;     */                             /* write to SIO clears buffer */
}

/********************/
void internal_clock(void)   /* master-slave, internal clock */
/********************/
{
CSIM.0=0;                                      /* internal clock                */
CSIM.1=1;
}

/********************/
void external_clock(void)   /* slave-master, external clock */
```

```
/********************/
{
CSIM.0=0;                                /* external clock              */
CSIM.1=0;
}

/********************/
void receive(void)                /* rx on, tx off                       */
/********************/
{
CTxE=0;
CRxE=1;
}

/********************/
void transmit(void)               /* tx on, rx off                       */
/********************/
{
CTxE=1;
CRxE=0;
}

/********************/
void error(void)
/********************/
{
/*      exit(1);*/
}


/********************/
void send_cone_variable(int slave_used,int command_used) /* send lube to KBS    */
/********************/
{
VARIABLE_WAIT_FLAG=FALSE;

ERRORFLG=0;
reset();                          /* reset SB0 line                       */
internal_clock();                  /* master-slave, internal clock */
transmit();                       /* tx on, rx off                    */

do
        {
        address_pulses();              /* pre processing address pulses*/
        SIO=slave_used;                    /* send serial control address    */
        ERRORFLG=FALSE;                /* reset ERRORFLG                  */
        ACKD_wait();                   /* wait for acknowledge           */
        }
while(ERRORFLG==TRUE);     /* repeat until address sent successfully */

ADDRESS_GONE=TRUE;                     /* address gone flag ON            */

reset();                               /* reset SB0 line                  */
internal_clock();
transmit();                            /* tx on, rx off                   */

do
        {
        command_pulses();               /* pre processing command pulses*/
        SIO = command_used;        /* send lube temp cmd */
        ERRORFLG=FALSE;                 /* reset ERRORFLG                 */
        ACKD_wait();                    /* wait for acknowledge           */
        }
while (ERRORFLG==TRUE);    /* repeat until command sent successfully */

COMMAND_GONE=TRUE;                     /* command gone flag ON            */

receive();                             /* rx on, tx off                   */
external_clock();                      /* slave-master, external clock */

do
{}
while (P3.2==0);                       /* wait while SB0 pin is ready */

temp=SIO;
```

```
do
{}
while (VARIABLE_WAIT_FLAG==FALSE);

VARIABLE_WAIT_FLAG=FALSE;
ACKT=1;

TXS=cone_var;              /* transmit RS232 wear value   */
wait_for_echo(cone_var);

ADDRESS_GONE=FALSE;      /* reset ADDRESS/COMMAND GONE  */
COMMAND_GONE=FALSE;

ERRORFLG=FALSE;
}

/*********************/
void send_cone_parameter(int slave_used,int command_used)
/* send alarm parameters to slave */
/*********************/
{

int alarm_value;

RS232_int_occurred=FALSE;               /* reset RS232 interrupt flag     */

do
{}
while (RS232_int_occurred==FALSE);

alarm_value=kbs_option;

reset();                        /* reset SB0 line                         */
internal_clock();                       /* master-slave, internal clock */
transmit();                     /* tx on, rx off                  */

do
        {
        address_pulses();               /* pre processing address pulses*/
        SIO=slave_used;                     /* send serial control address      */
        ERRORFLG=FALSE;                 /* reset ERRORFLG                   */
        ACKD_wait();                    /* wait for acknowledge             */
        }
while(ERRORFLG==TRUE);    /* repeat until address sent successfully */

ADDRESS_GONE=TRUE;                      /* address gone flag ON            */

reset();                        /* reset SB0 line                           */
transmit();                     /* tx on, rx off                  */

do
        {
        command_pulses();               /* pre processing command pulses*/
        SIO = command_used;          /* send lube temp cmd */
        ERRORFLG=FALSE;                 /* reset ERRORFLG                   */
        ACKD_wait();                    /* wait for acknowledge             */
        }
while (ERRORFLG==TRUE);   /* repeat until command sent successfully */

COMMAND_GONE=TRUE;                      /* command gone flag ON            */

reset();                        /* reset SB0 line                           */
internal_clock();                       /* master-slave, internal clock */
transmit();                     /* tx on, rx off                  */

ERRORFLG=FALSE;

do
{
SIO = alarm_value;      /* send alarm_value */
ERRORFLG=FALSE;
ACKD_wait();
}
while (ERRORFLG==TRUE);
```

```
  reset();

  ADDRESS_GONE=FALSE;      /* reset ADDRESS/COMMAND GONE  */
  COMMAND_GONE=FALSE;
  }

/*******************************/
void wait_for_echo(int echo_val)
/*******************************/
{
RS232_int_occurred=FALSE;
do
{}
while(RS232_int_occurred==FALSE);
}
```

```
/**********************************************************************/
/*                                                                    */
/*                    Slave      v7.01                              */
/*                                                                    */
/**********************************************************************/
/*                                                                    */
/*   Filename: S_SER232.C                                             */
/*   Author  : S.Yaxley                                               */
/*   Date    : 29/9/95                                                */
/*   Project : Cone Crusher Control OPTIMAX+                          */
/*   Slave Addresses 0x60 wear controller                            */
/*                   0x70 main control                               */
/*                                                                    */
/*   De Montfort University                                           */
/*                                                                    */
/**********************************************************************/


/*******************************/
/* INCLUDE FILES               */
/*******************************/

#include <stdio.h>          /* standard I/O                           */
#include <io78330.h>        /* address locations for flags         */
#include <serial.h>         /* defined flags <own>                 */
#include <display.h>        /* lcd control 16x2 NEClab <own>    */


/*******************************/
/* SLAVE ADDRESS               */
/*******************************/

#define SLAVE_ADDR 0x70             /* control slave address              */


/*******************************/
/* COMMANDS                    */
/*******************************/

#define SLAVE_CHECK_COMMAND 0xa0
#define HYD_PRESSURE_COMMAND 0xa1
#define MOTOR_CURRENT_COMMAND 0xa2
#define LUBE_TEMP_COMMAND 0xa3
#define HYD_TEMP_COMMAND 0xa4

#define INLET_OIL_COMMAND 0xb0
#define OUTLET_OIL_COMMAND 0xb1
#define FLOAT_SWITCH_COMMAND 0xb2
#define TRAMP_METAL_COMMAND 0xb3
#define CONVEYOR_COMMAND 0xb4

#define PRESSURE_ALARM_COMMAND 0xc1
#define MOTOR_ALARM_COMMAND 0xc2
#define LUBE_ALARM_COMMAND 0xc3
#define HYD_ALARM_COMMAND 0xc4


/*******************************/
/* PROTOTYPE PROCEDURES        */
/*******************************/

void interrupt_proc(void);
void SB0_inis(void);                /* serial bus initialise              */
void ACKD_wait(void);               /* wait for acknowledge from master */
void timer_set(void);               /* time out period for acknowledge  */
void reset(void);                   /* reset SB0 line                     */
void transmit(void);                /* set transmit flag ON, rx flag OFF*/
void receive(void);              /* set transmit flag OFF, rx flag ON*/
void external_clock(void);          /* external clock, master-slave       */
void internal_clock(void);          /* internal clock, slave-master       */
void wait_for_address(void);        /* wait for address sent              */
void wait_for_command(void);        /* wait for command sent              */
void slave_check(void);             /* function for slave search command*/
void read_lube_alarm(void);
void read_pot(int);                 /* slave potentiometer                */
void read_switch(int);
void read_alarms(void);
void led_init(void);
```

```
void switch_init(void);
/*******************************/
/* GLOBAL VARIABLES            */
/*******************************/

char command;                         /* type of command variable         */
char    lube_alarm,
                hyd_alarm,
                motor_alarm,
                pressure_alarm;

char pot_val,switch_val,temp;
/*******************************/
/* FLAGS                       */
/*******************************/

char ADDRESS_FLAG,                    /* address received flag            */
        COMMAND_FLAG,                 /* command received flag            */
        ERRORFLG,                     /* error flag                       */
        ALARM_FLAG,                   /* master-slave alarm transfer      */
        ALARM_RCVD_FLAG;

/*******************************/
/* INTERRUPT SERVICE ROUTINES   */
/*******************************/

/*****************************************/
interrupt [INTCSI_vect] void rx_end(void)    /* serial bus interrupt     */
/*****************************************/
{
char sio_temp;

sio_temp=SIO;

if ((ADDRESS_FLAG==TRUE)&&(COMMAND_FLAG==TRUE)&&(ALARM_FLAG==TRUE))
        {
        switch (command)
                {
                case PRESSURE_ALARM_COMMAND:pressure_alarm=sio_temp;break;
                case MOTOR_ALARM_COMMAND:motor_alarm=sio_temp;break;
                case LUBE_ALARM_COMMAND:lube_alarm=sio_temp;break;
                case HYD_ALARM_COMMAND:hyd_alarm=sio_temp;break;
                }
        ALARM_RCVD_FLAG=TRUE;
        }

if ((ADDRESS_FLAG==TRUE)&&(COMMAND_FLAG==FALSE))
        {
        switch (sio_temp)
                {
                case SLAVE_CHECK_COMMAND:
                        interrupt_proc();
                        command=SLAVE_CHECK_COMMAND;break;
                case HYD_PRESSURE_COMMAND:
                        interrupt_proc();
                        command=HYD_PRESSURE_COMMAND;break;
                case MOTOR_CURRENT_COMMAND:
                        interrupt_proc();
                        command=MOTOR_CURRENT_COMMAND;break;
                case LUBE_TEMP_COMMAND:
                        interrupt_proc();
                        command=LUBE_TEMP_COMMAND;break;
                case HYD_TEMP_COMMAND:
                        interrupt_proc();
                        command=HYD_TEMP_COMMAND;break;
                case INLET_OIL_COMMAND:
                        interrupt_proc();
                        command=INLET_OIL_COMMAND;break;
                case OUTLET_OIL_COMMAND:
                        interrupt_proc();
                        command=OUTLET_OIL_COMMAND;break;
                case FLOAT_SWITCH_COMMAND:
                        interrupt_proc();
                        command=FLOAT_SWITCH_COMMAND;break;
                case TRAMP_METAL_COMMAND:
```

```
                                interrupt_proc();
                                command=TRAMP_METAL_COMMAND;break;
                        case CONVEYOR_COMMAND:
                                interrupt_proc();
                                command=CONVEYOR_COMMAND;break;
                        case LUBE_ALARM_COMMAND:
                                interrupt_proc();
                                command=LUBE_ALARM_COMMAND;ALARM_FLAG=TRUE;break;
                        case HYD_ALARM_COMMAND:
                                interrupt_proc();
                                command=HYD_ALARM_COMMAND;ALARM_FLAG=TRUE;break;
                        case MOTOR_ALARM_COMMAND:
                                interrupt_proc();
                                command=MOTOR_ALARM_COMMAND;ALARM_FLAG=TRUE;break;
                        case PRESSURE_ALARM_COMMAND:
                                interrupt_proc();
                                command=PRESSURE_ALARM_COMMAND;ALARM_FLAG=TRUE;break;
                }
        }

if ((sio_temp==SLAVE_ADDR)&&(COMMAND_FLAG==FALSE)&&(ADDRESS_FLAG==FALSE))
        {
        ACKT=1;                 /* set acknowledge flag                          */
        BSYE=0;              /* BYSE clear - needed else command will not be sent */

        receive();              /* set receive flag                              */

        ADDRESS_FLAG=TRUE;          /* address sent                              */
        }

}

/*****************************************/
interrupt [INTCM20_vect] void time_out(void) /* time out interrupt */
/*****************************************/
{
ERRORFLG=TRUE;                  /* error if time out has occurred    */
ACKT=1;                         /* flag slave acknowledge               */
BSYE=0;                         /* reset busy flag                          */

TMC1.3=0;                       /* stop timer                               */
}

/*------------------------------------------------------------------------*/
/***************** END OF INTERRUPTS *****************************/
/*------------------------------------------------------------------------*/

/***********************************************************************/
void main(void)
/***********************************************************************/
{
ADDRESS_FLAG=FALSE;             /* reset ADDRESS FLAG                     */
COMMAND_FLAG=FALSE;             /* reset COMMAND FLAG                     */
ERRORFLG=FALSE;                 /* reset ERROR FLAG                         */
ALARM_FLAG=FALSE;
ALARM_RCVD_FLAG=FALSE;

SB0_inis();                     /* initialise serial bus                 */
timer_set();                    /* set time out period                   */

led_init();
switch_init();
enable_interrupt();

for(;;)
        {
        receive();
        external_clock();

        wait_for_address();         /* wait for address to be sent          */
        wait_for_command();         /* wait for command to be sent          */

        switch (command)            /* check command type from interrupt*/
                {
                case SLAVE_CHECK_COMMAND:slave_check();break;
```

```
                case HYD_PRESSURE_COMMAND:read_pot(1);break;
                case MOTOR_CURRENT_COMMAND:read_pot(2);break;
                case LUBE_TEMP_COMMAND:read_pot(3);break;
                case HYD_TEMP_COMMAND:read_pot(4);break;

                case INLET_OIL_COMMAND:read_switch(0);break;
                case OUTLET_OIL_COMMAND:read_switch(1);break;
                case FLOAT_SWITCH_COMMAND:read_switch(2);break;
                case TRAMP_METAL_COMMAND:read_switch(3);break;
                case CONVEYOR_COMMAND:read_switch(4);break;

    /*          case LUBE_ALARM_COMMAND:read_lube_alarm();break;*/

                case PRESSURE_ALARM_COMMAND:read_alarms();break;
                case MOTOR_ALARM_COMMAND:read_alarms();break;
                case LUBE_ALARM_COMMAND:read_alarms();break;
                case HYD_ALARM_COMMAND:read_alarms();break;
                default :break;
                }
        ADDRESS_FLAG=FALSE;            /* reset ADDRESS FLAG                          */
        COMMAND_FLAG=FALSE;           /* reset COMMAND FLAG                          */
        }
}

/*****************************/
void interrupt_proc(void)
/*****************************/
{
ACKT=1;                       /* set acknowledge flag                          */
BSYE=0;                       /* BYSE clear - needed else command will not be sent */

COMMAND_FLAG=TRUE;            /* command sent                                  */
}

/*****************************/
void wait_for_address(void)                 /* wait for address              */
/*****************************/
{
while(ADDRESS_FLAG==FALSE)
        {
        }
}

/*****************************/
void wait_for_command(void)                 /* wait for command              */
/*****************************/
{
while(COMMAND_FLAG==FALSE)
        {
        }
}

/*****************************/
void slave_check(void)          /* function for slave search command */
/*****************************/
{
reset();
transmit();
internal_clock();

do
{
        SIO=SLAVE_ADDR;               /* transmit Slave address             */
        ERRORFLG=FALSE;               /* reset error flag                   */
        ACKD_wait();                  /* wait for acknowledge from master */
}
while(ERRORFLG==TRUE);                /* repeat until sent correctly        */

external_clock();                        /* set external to receive address   */

CTxE=0;                               /* reset transmit & receive          */
CRxE=0;

receive();
}
```

```c
/*****************************/
void read_lube_alarm(void)
/*****************************/
{
receive();                          /* set receive flag                         */
external_clock();

ACKE=1;
BSYE=1;

do
{}
while (ALARM_RCVD_FLAG==FALSE);

ALARM_FLAG=FALSE;
ALARM_RCVD_FLAG=FALSE;
ACKE=0;
}


/*****************************/
void SB0_inis(void)              /* serial bus initialization processing  */
/*****************************/
{
CSIM = 0x48;                     /* RxD SBI mode & external clock      */
SBIC = 0x81;                     /* busy enabled & output RELT            */
PMC3 = 0x1c;                     /* SB0 SB1 & SCK                            */

ISM1L.1=0;                       /* NOT MACRO INTCSI  vector             */
MK1L.1 = 0;                      /* INTCSI     .                          */
MK0H.3=0;                        /* timer           */
}


/*****************************/
void timer_set(void)                 /* time out for acknowledge               */
/*****************************/
{
TUM0.2=1;
TMC1.2=1;

CM20=200;                        /* 15360=30.72ms - 100=0.2ms          */
}


/*****************************/
void ACKD_wait(void)             /* wait for acknowledge                    */
/*****************************/
{
do
      {
      TMC1.3=1;                  /* start timer                              */
      }
while(ACKD==0);

TMC1.3=0;                        /* stop & clear timer                       */
}


/*****************************/
void reset(void)                 /* reset serial bus                        */
/*****************************/
{
int clearval;                        /* variable to clear tx & rx pins    */

CTxE=0;                          /* reset rx & tx                            */
CRxE=0;
P3.2=clearval;
P3.4=clearval;
}


/*****************************/
void transmit(void)                      /* tx on rx off               */
/*****************************/
{
CTxE=1;
CRxE=0;
}
```

```
/****************************/
void receive(void)                          /* rx on tx off              */
/****************************/
{
CTxE=0;
CRxE=1;
}


/****************************/
void external_clock(void)                   /* master-slave                  */
/****************************/
{
CSIM.0=0;                                   /* external clock      */
CSIM.1=0;
}


/******************/
void internal_clock(void)                   /* slave-master                  */
/******************/
{
CSIM.0=0;                                   /* internal clock      */
CSIM.1=1;
}


/******************/
void read_pot(int ADval)                    /* slave potentiometer           */
/******************/
{
reset();
transmit();                                 /* tx on, rx off      */
internal_clock();                           /* internal clock, slave-master*/

ADM=0xd0 | ADval;

switch (ADval)
        {
        case 1:pot_val=ADCR1H;
                if (pot_val>=pressure_alarm)P1.0=1;
                else P1.0=0;
                break; /* hyd pressure */
        case 2:pot_val=ADCR2H;
                if (pot_val>=motor_alarm)P1.1=1;
                else P1.1=0;
                break; /* motor current */
        case 3:pot_val=ADCR3H;
                if (pot_val>=lube_alarm)P1.2=1;
                else P1.2=0;
                break; /* lube temp */
        case 4:pot_val=ADCR4H;
                if (pot_val>=hyd_alarm)P1.3=1;
                else P1.3=0;
                break; /* hyd temp */
        }

do
{
        SIO=pot_val;            /* transmit SB0 pot value     */
        ERRORFLG=FALSE;         /* reset error flag                      */
        ACKD_wait();            /* wait for acknowledge from master */
}
while(ERRORFLG==TRUE);          /* repeat until sent correctly           */

external_clock();               /* set external to receive address    */

CTxE=0;                         /* reset transmit & receive              */
CRxE=0;

receive();
}


/******************/
void read_switch(int switchport)                    /* slave potentiometer       */
/******************/
{
```

```
    reset();
    transmit();                    /* tx on, rx off                              */
    internal_clock();              /* internal clock, slave-master        */

    switch (switchport)
            {
            case 0:switch_val=P0.0;
                    if (switch_val==1)P1.5=1;
                    else P1.5=0;
                    break; /* inlet oil */
            case 1:switch_val=P0.1;
                    if (switch_val==1)P1.7=1;
                    else P1.7=0;
                    break; /* outlet oil */
            case 2:switch_val=P0.2;
                    if (switch_val==1)P1.6=1;
                    else P1.6=0;
                    break; /* float switch */
            case 3:switch_val=P0.3;
                    if (switch_val==1)P1.5=1;
                    else P1.5=0;
                    break; /* tramp metal */
            case 4:switch_val=P0.4;
                    if (switch_val==1)P1.5=1;
                    else P1.5=0;
                    break; /* conveyor */
            }

    do
    {
            SIO=switch_val;        /* transmit SB0 pot value              */
            ERRORFLG=FALSE;        /* reset error flag                          */
            ACKD_wait();           /* wait for acknowledge from master */
    }
    while(ERRORFLG==TRUE);         /* repeat until sent correctly          */

    external_clock();              /* set external to receive address     */

    CTxE=0;                        /* reset transmit & receive             */
    CRxE=0;

    receive();
    }

/*****************************/
void read_alarms(void)
/*****************************/
{
receive();              /* set receive flag                                     */
external_clock();

ACKE=1;
BSYE=1;

do
{}
while (ALARM_RCVD_FLAG==FALSE);

ALARM_FLAG=FALSE;
ALARM_RCVD_FLAG=FALSE;
ACKE=0;
}

/********************/
void led_init(void)
/********************/
{
PM1=0;
PMC1=0;
P1=0;
}

/********************/
void switch_init(void)
/********************/
```

```
{
/*PM0=1;*/
PMC0=0;
}
```

APPENDIX B

SMART WEAR SENSOR SOFTWARE

APPENDIX B - SMART WEAR SENSOR SOFTWARE

Transmitter Software

```
/****************************************************/
/* Capacitor Wear Sensor Software + transmit        */
/* Stuart Yaxley  25/9/95                           */
/****************************************************/

#include <io78330.h>
/*#include <intrins.h>*/
#include <stdio.h>

void RS232_inis(void);                    /* RS232 initialisation processing     */
void delay_time(int val);

unsigned int counter,upper_byte,lower_byte;
static int ledval,time_count,send_flag;
float num;


/*********************************************/
interrupt [INTP0_vect] void data_ready(void)
/*********************************************/
{
disable_interrupt();
counter=CT10;
lower_byte=counter&0x7f;
upper_byte=((counter>>7)&0x7f)|0x80;
TMC0.7=0;
TMC0.7=1;
enable_interrupt();
}


/*********************************************/
interrupt [INTCM20_vect] void time_ready(void)
/*********************************************/
{
disable_interrupt();
time_count++;
if(time_count==5)
        {
        P0.0=ledval;
        ledval=!ledval;
        time_count=0;
        send_flag=1;
        }
enable_interrupt();
}


/*********************************************/
interrupt[INTSER_vect] void error1(void)                    /* RS232 error     */
/*********************************************/
{
disable_interrupt();

enable_interrupt();
}


/*********************************************/
interrupt[INTST_vect] void transmission(void)
/*********************************************/
{
disable_interrupt();

enable_interrupt();
}
```

```
/*******************************************/
void main(void)
/*******************************************/
{
PM0=0x00;
PMC0=0x00;

RS232_inis();

TUM0=0x02;      /* Overflow on Timer 1 OVF1=1 */
TUM1=0x04;      /* CT10 capture trigger source INTP0 */
INTM0=0x00;     /* INTP0 valid on both edge */
/*IFOL=0x02;*/  /* PIF0 interrupt signal generated for INTP0 */
MK0L.1=0;       /* interrupt processing enabled */
ISM0L.1=0;      /* vectored interrupt */
CSE0L.1=0;      /* Vectored interrupt processing for INTP0 */
TMC0.4=0;       /* Timer 1 internal clock fclk/8 */
TMC0.5=0;       /* TM1 clears disable, not when INTP0 input - opposite to valid edge */
TMC0.6=0;       /* TM1 free running mode */

TUM0.2=0;
TMC1.2=1;
CM20=50000;
MK0H.3=0;
PM0=0x00;
PMC0=0x00;
ledval=0;
time_count=0;
send_flag=0;

TMC1.3=1;

TMC0.7=1;       /* Start TM1 counting */

enable_interrupt();

for(;;)
        {
        if(send_flag==1)
                {
                if(ledval!=0)   /* txs upper byte */
                        TXS=upper_byte;
                if(ledval==0)   /* txs lower byte */
                        TXS=lower_byte;
                send_flag=0;
                }
        }
}


/*********************/
void RS232_inis(void)    /* RS232 initialization processing        */
/*********************/
{
int brgg;                        /* baud rate generator                   */

MK0H = MK0H & 0x3f;              /* enable 0011 1111 INTSR,INTSER    */
PMC3=PMC3|0x03;
ASIM = 0xb8;           /* even_parity, 8bits, 1stopbit, int baudrate */
brgg = 103;
BRG = brgg;               /* BAUD RATE = 2400, page 6-109            */
BRGM = BRGM | 0x88;
}


/*****************************************************/
void delay_time(int val)
/*****************************************************/
{
int i;
```

```
for(i=0;i<=val;i++){}
}
```

Reciever Software

```
/**********************************************/
/* Receiver Software for transmitted RS232     */
/* Stuart Yaxley  18/9/95                       */
/**********************************************/

#include <io78330.h>
/*#include <intrins.h>*/
#include <stdio.h>
#include <display.h>
#include <string.h>

/*#define TRUE 1
#define FALSE !TRUE*/
#define SENSOR_LENGTH 50

unsigned char RS232_val;
int RS232_int_occurred;
int PB_FLAG,
        CALIBRATE_FLAG;
int trans_end=0;

void RS232_inis(void);            /* RS232 initialisation processing    */
void wait_trans_end(void);

/****************************************/
interrupt[INTSR_vect] void receiving(void)       /* RS232 interrupt       */
/****************************************/
{
disable_interrupt();

RS232_val = RXB;                        /* set incoming port value to RS232_val */
RS232_int_occurred=TRUE;             /* interrupt has occured              */

enable_interrupt();
}


/****************************************/
interrupt[INTSER_vect] void error1(void)         /* RS232 error           */
/****************************************/
{
disable_interrupt();

enable_interrupt();
}

/****************************************/
interrupt[INTST_vect] void transmission(void)
/****************************************/
{
disable_interrupt();
trans_end=1;
enable_interrupt();
}


/****************************************/
interrupt [INTP1_vect] void push_button(void)
/****************************************/
{
disable_interrupt();
PB_FLAG=TRUE;
enable_interrupt();
}

/**********************************************/
void main(void)
/**********************************************/
```

```
{
float total_val,calibrate_val;
int raw_upper,raw_lower,strength;
float length,length_val,voltage;
int sample=0;
char main_val,point_val;


RS232_int_occurred=FALSE;              /* reset RS232 interrupt flag      */

PM0=PM0|0x01;
PMC0=0x00;
PM3=0xff;
PMC3=0x00;

MK0L.2=0;       /* interrupt processing enabled INTP1 */
ISM0L.2=0;      /* vectored interrupt */
CSE0L.2=0;      /* Vectored interrupt processing for INTP1 */
INTM0.4=0;      /* INTP1 valid on falling edge */

MK1L.0 = 0;              /* enable xxxx xxx0 INTST */
ISM1L.0 = 0;    /* Vectored handling of INTST */
CSE1L.0 = 0;    /* Vectored handling of INTST */


PB_FLAG=FALSE;
CALIBRATE_FLAG=FALSE;

ADM=0xd0;                       /* ANI0 - P7.0, 144 states, select mode 1 buffer */
init_lcd();
RS232_inis();

strcpy(txt1,"Calibrate Sensor");
print(TRUE,0,0,0,txt1);

enable_interrupt();

for(;;)
        {
        if (RS232_int_occurred==TRUE)     /* if transmission */   .
                {
                if(RS232_val>=128)
                        raw_upper=(RS232_val-128);
                else
                        raw_lower=RS232_val;

                total_val=(raw_upper<<7)|raw_lower;
                strength=ADCR0;
                voltage=strength*5.0/1023.0;

                if(PB_FLAG==TRUE)
                        {
                        calibrate_val=total_val;
                        CALIBRATE_FLAG=TRUE;
                        strcpy(txt1,"Calibrated");
                        print(TRUE,0,0,0,txt1);
                        PB_FLAG=FALSE;
                        }
                if((CALIBRATE_FLAG==TRUE)&&(voltage>2.0))     /* if calibrated,  and Detect
strong enough */
                        {
                        length_val=total_val/calibrate_val;
                        length=length_val*SENSOR_LENGTH;
                        sprintf(txt1,"L%.2f S%d V%.2f",length,sample,voltage);
                        print(FALSE,0,0,0,txt1);

                        main_val=((int)length)|0x80;
                        point_val=(int)((length-main_val)*100);

/*                      if((sample%2)==0)*/
                        TXS=main_val;
/*                      else*/
                        wait_trans_end();
                        TXS=point_val;
                        wait_trans_end();
```

```
                        sample++;
                    }
/*              TXS=RS232_val;*/

                RS232_int_occurred=FALSE;
                PB_FLAG=FALSE;
                }
        }
}
/**********************************/
void wait_trans_end(void)
/**********************************/
{
while(trans_end == 0)
        {
        }               /* Wait until transmission completed (INTST) */
trans_end = 0;
}


/********************/
void RS232_inis(void)   /* RS232 initialization processing        */
/********************/
{
int brgg;                       /* baud rate generator                  */

MK0H = MK0H & 0x3f;             /* enable 0011 1111 INTSR,INTSER          */
PMC3=PMC3|0x03;
ASIM = 0xf8;            /* even_parity, 8bits, 1stopbit, int baudrate */
brgg = 103;
BRG = brgg;             /* BAUD RATE = 2400, page 6-109          */
BRGM = BRGM | 0x88;
}
```

APPENDIX C
CIRCUIT DIAGRAMS

174
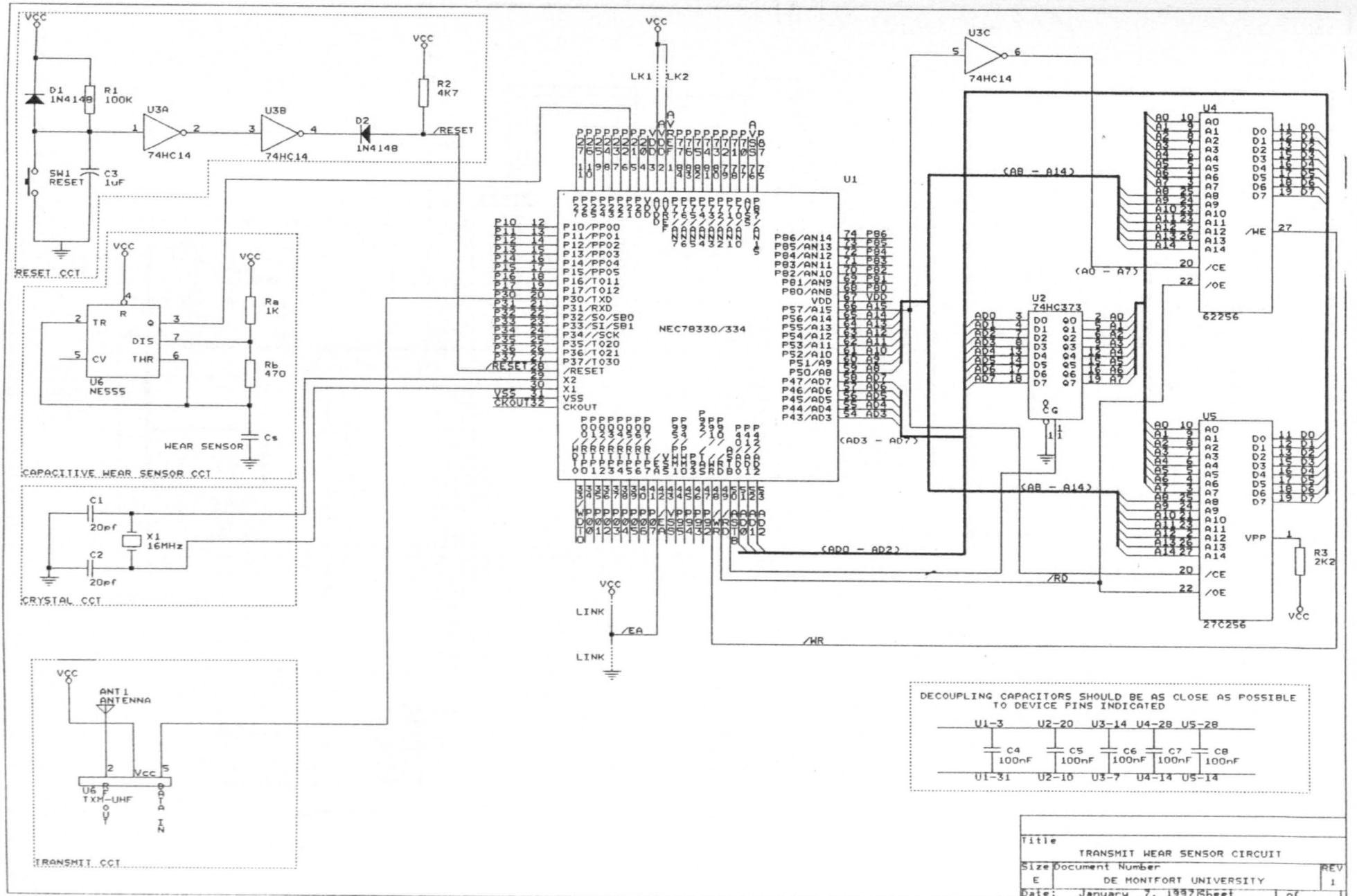
Title
TRANSMIT WEAR SENSOR CIRCUIT
Size | Document Number | REV
E | DE MONTFORT UNIVERSITY | 1
Date: January 7, 1997 | Sheet 1 of 1

APPENDIX C - CIRCUIT DIAGRAMS
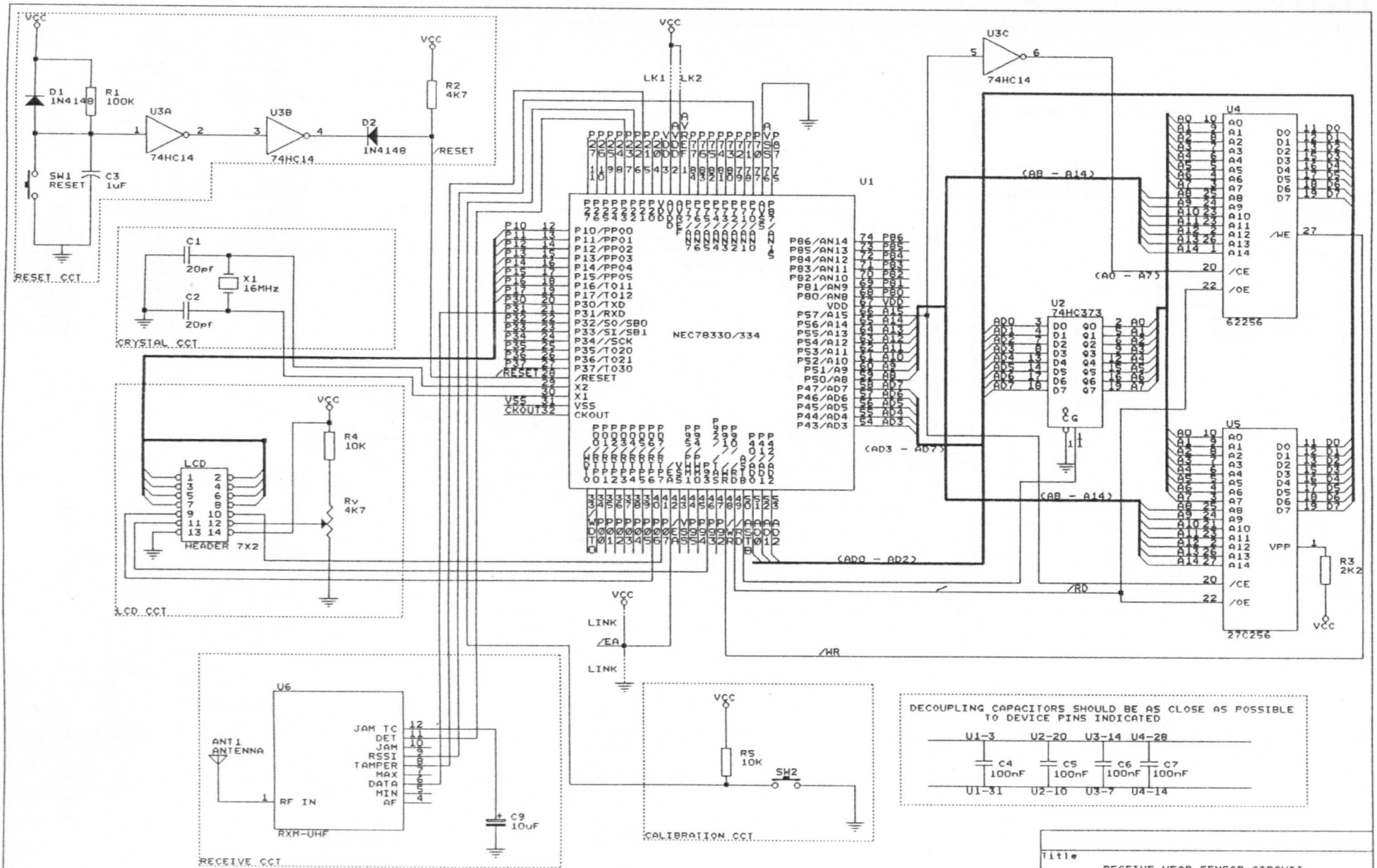
Title		
	RECEIVE WEAR SENSOR CIRCUIT	
Size	Document Number	REV
E	DE MONTFORT UNIVERSITY	1
Date:	January 7, 1997 Sheet 1 of 1	

176
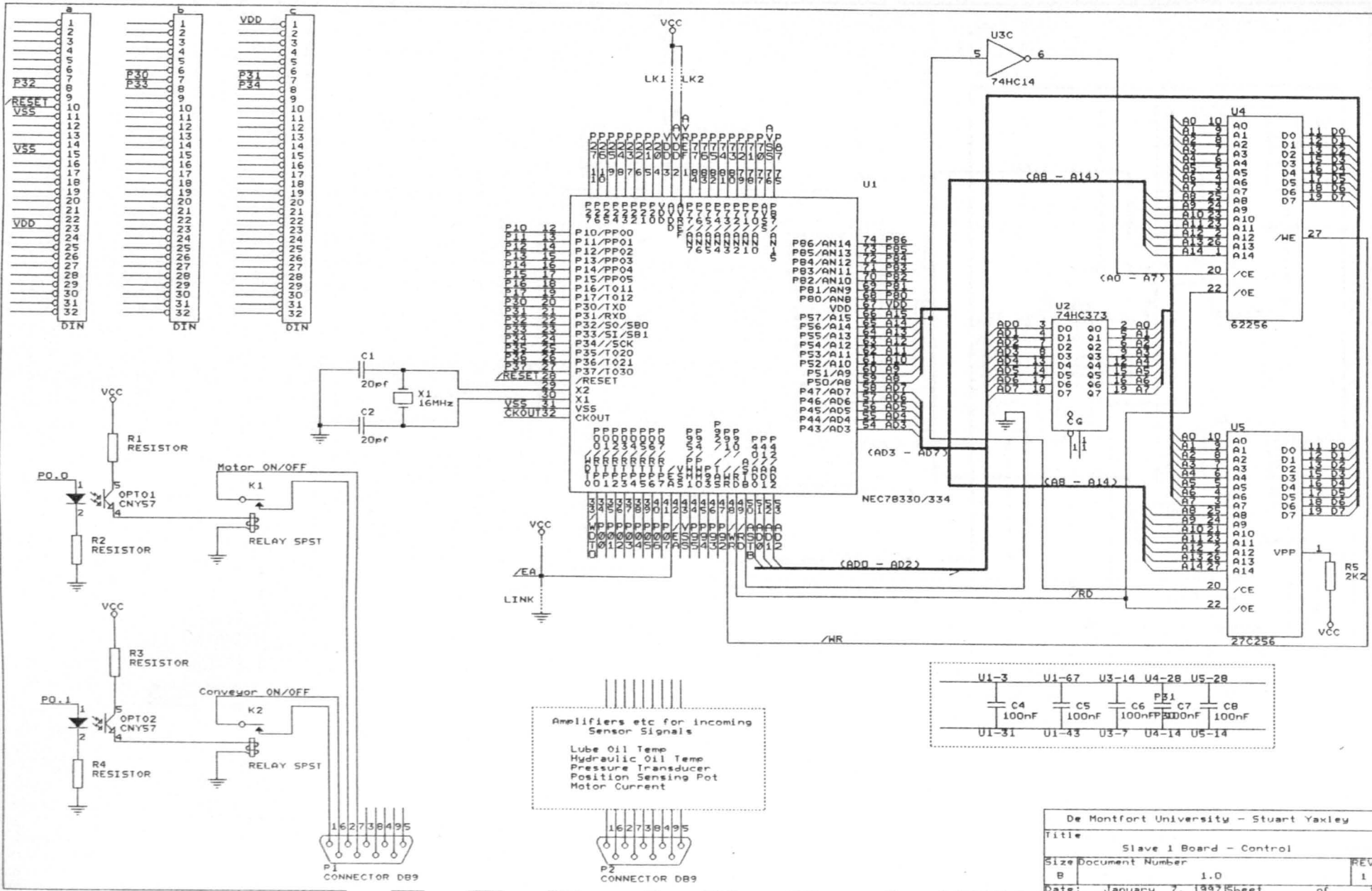
De Montfort University — Stuart Yaxley

Title
Slave 2 Board — Wear Sensor Input

Size: B Document Number: 1.0 REV: 1

Date: January 7, 1997 Sheet of 1

APPENDIX D

LIST OF PUBLISHED PAPERS

APPENDIX D - LIST OF PUBLISHED PAPERS

1. **Parkin R., Yaxley S.** (1994). Improving System Efficiency Through Mechatronics. Electro Technology. Institite of Electronics and Electrical Incorporated Engineers. Volume 5 Number 4. August/September ISSN 0306-8552 pp26-27

2. **Parkin R., Yaxley S., Bearman R.** (1994) Real Time Distributed Control for the Intelligent Operation of Rock Crushers. Proceedings International Workshop on Mechatronic Design and Modelling. Middle East Technical University, Ankara, Turkey, 15-19 Nov 1993 Ed. A Erden, M Erkman, A Tekkaya. August pp1-9

3. **Parkin R., Yaxley S., Bearman R.** (1994). Intelligent Control of Rock Crushers Using Real Time Distributed Systems and KBS. ESDA Second Biennial European Joint Conference on Engineering Systems Design and Analysis. Queen Mary and Westfield College, University of London, July 4-7 ISBN 0 7918 1280 4, pp153-158

4. **Yaxley S., Parkin R.** (1994), Distributed Control Systems Within a Mechatronic Environment. Proceedings, Joint Hungarian - British Mechatronics Conference, Sept 21-23 Budapest.

5. **Yaxley S., Parkin R., Bearman R.** (1993) Condition Monitoring of Cone Crushers Using Distributed Intelligence. Proceedings, COMADEM 93. 5th International Congress on Condition Monitoring and Diagnostic Engineering Management, 21-23 July University of the West of England, Bristol. Ed Raj Rao, Trmal ISBN 1 871056 72 1, pp485-490

6. **Yaxley S., Parkin R., Bearman R.** (1993), Distributed Control for Cone Crushers. Proceedings Int. Workshop on Mechatronical Computer Systems for Perception and Action, 1-3 June Halmstad University, Sweden. ISBN 91 630 1847 0 p39-44

7. **Yaxley S., Parkin R., Bearman R.** (1994). Heirarchical Distributed Intelligence Within a Quarrying Environment. Proceedings ICMA 94. International Conference on Machine Automation - Mechatronics Spells Profitability, Tampere, Finland. Feb 15-18, ISBN 951 722 107 X, pp745-757

PAGES NOT SCANNED AT

THE REQUEST OF THE

UNIVERSITY

SEE ORIGINAL COPY OF

THE THESIS FOR THIS

MATERIAL
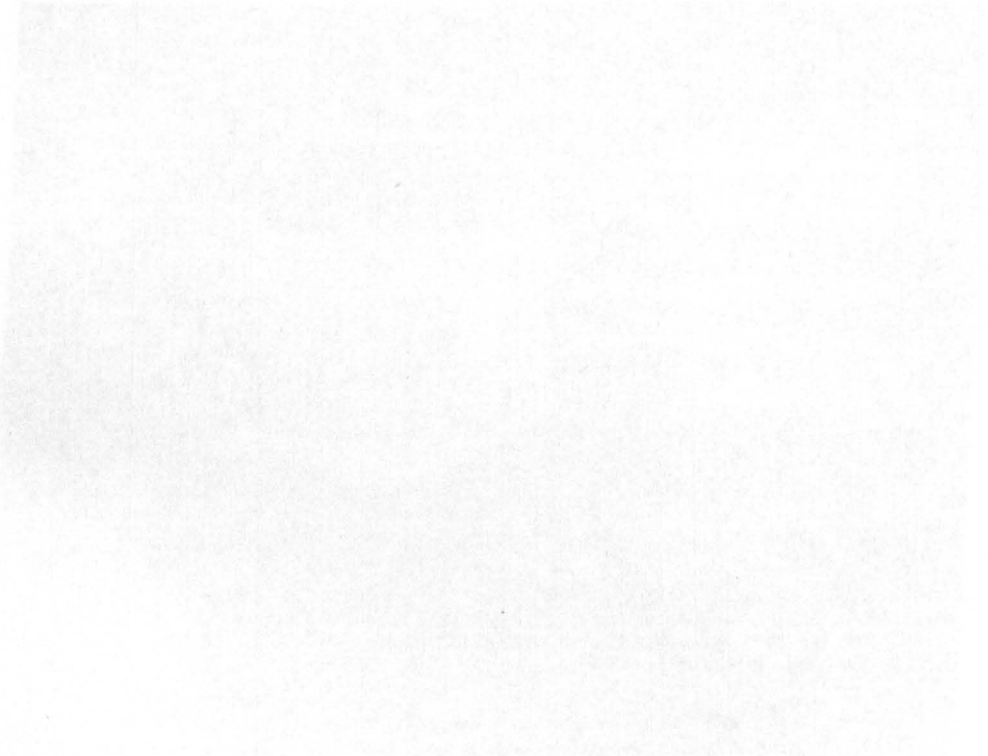
PAGE/PAGES EXCLUDED

UNDER INSTRUCTION

FROM THE UNIVERSITY