Microprocessor-based elastic feed systems for sewing applications

Marshall, Stephen J. L.

How to cite:

Marshall, Stephen J. L. (1990) *Microprocessor-based elastic feed systems for sewing applications*, Durham theses, Durham University. Available at Durham E-Theses Online: http://etheses.dur.ac.uk/6290/

ABSTRACT

Microprocessor-based Elastic Feed Systems for Sewing Applications

by Stephen J. L. Marshall

MSc Thesis 1990

The manufacture of certain garments, such as underwear and T-shirts, requires the attachment of tensioned elastics and tapes to workpieces which make up the garment. If the tape is incorrectly tensioned, when it returns to its original length after attachment to the workpiece the final garment will be too tight or too loose, and a sub-standard garment is produced.

This thesis describes the design and developmant of two microprcessor-based systems which control the feed of elastic tape to sewing machines for such operations. The first uses an open-loop control approach, and can maintain tension to within 2% of that required for correct workpiece sizing during normal operation. However, any loss of tension due to outside factors, such as slippage of the elastic through the feed mechanism, cannot be recovered sufficiently quickly to prevent incorrect sizing of one or more workpieces.

To overcome this problem, a second system was developed which employs closed-loop control to maintain the correct tension. A transducer senses the tension in the elastic, and provides feedback which allows the control algorithm to compensate rapidly for changes in tension. An adaptive control loop is also employed to compensate for practical problems encountered such as operator workpiece feed rate, elastic feed path friction, and variations in the physical characteristics of the elastic tape itself. Workpiece sizing can be maintained to within a pre-defined tolerance, usually ± 10mm.

Test data from laboratory and factory tests is included, and the performance of the two systems is compared.

Microprocessor-based Elastic Feed Systems for Sewing Applications

by

Stephen J. L. Marshall

A Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

School of Engineering and Applied Science

The University of Durham
1990

DECLARATION

The work contained in this thesis has not been submitted elsewhere for any other degree or qualification and unless referenced it is the author's own work.

ACKNOWLEDGEMENTS

This project was funded jointly by the ACME (Applications of Computers to Manufacturing Engineering) directorate of the SERC (Science and Engineering Research Council) and Courtaulds Clothing Brands, Gateshead (formerly Jockey UK Ltd and Lyle and Scott Ltd).

The help of the personnel in the School of Engineering and Applied Science at Durham University was greatly appreciated. In particular, the help of Mr. T. Nancarrow for his assistance with computing problems, Mr. P. Friend for assistance with electrical equipment, and Mr. B. Blackburn for the construction of mechanical components. I am grateful for the help and encouragement of the two supervisors of this project, Dr. J. E. L. Simmons and in particular Dr. M. Sarhadi, who originally suggested that I should undertake this project. Finally, thanks go to Mr. J. Brown of Courtaulds Clothing Brands for his help in arranging visits to the factory in Gateshead, and for supplying a sewing machine, material and elastic tape for testing of the feed systems.

CONTENTS

CHAPTER 1 - INTRODUCTION

CHAPTER 2 - THE OPEN-LOOP CONTROL SYSTEM

CHAPTER 3 - THE CLOSED-LOOP CONTROL SYSTEM

CHAPTER 4 - THE STAND-ALONE CONTROL SYSTEM

APPENDICES

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 The attachment of elastic tape to garments

Clothing retailers and wholesalers expect a very high standard of product quality from the garment manufacturing industry. But when nearly every stage of production involves the handling and processing of flexible materials, there are many assembly operations in which a small error can result in a sub-standard garment. One such operation is the attachment of pre-tensioned elastic tape or rubber tape to workpieces which make up garments such as underwear, sportswear and T-shirts.

The pre-tensioned tape is normally fed in to the sewing head from above through a slot in the presser foot. The tape is pulled in to the sewing head, along with the workpiece to which it is being sewn, by the action of the presser foot on the feed dogs as each stitch is formed, as shown in fig. 1. The feed dogs are positioned directly below the needle, and are driven via a camshaft connected to the main sewing machine drive. The path followed by the feed dogs is nearly circular. At the top of the path, the serrated surface of the feed dogs protrudes above the flat working surface of the sewing machine. The presser foot is spring-loaded, and is positioned directly above the feed dogs. The needle passes through a cut-out at the side of the presser foot as it moves up and down to form each stitch. The workpiece

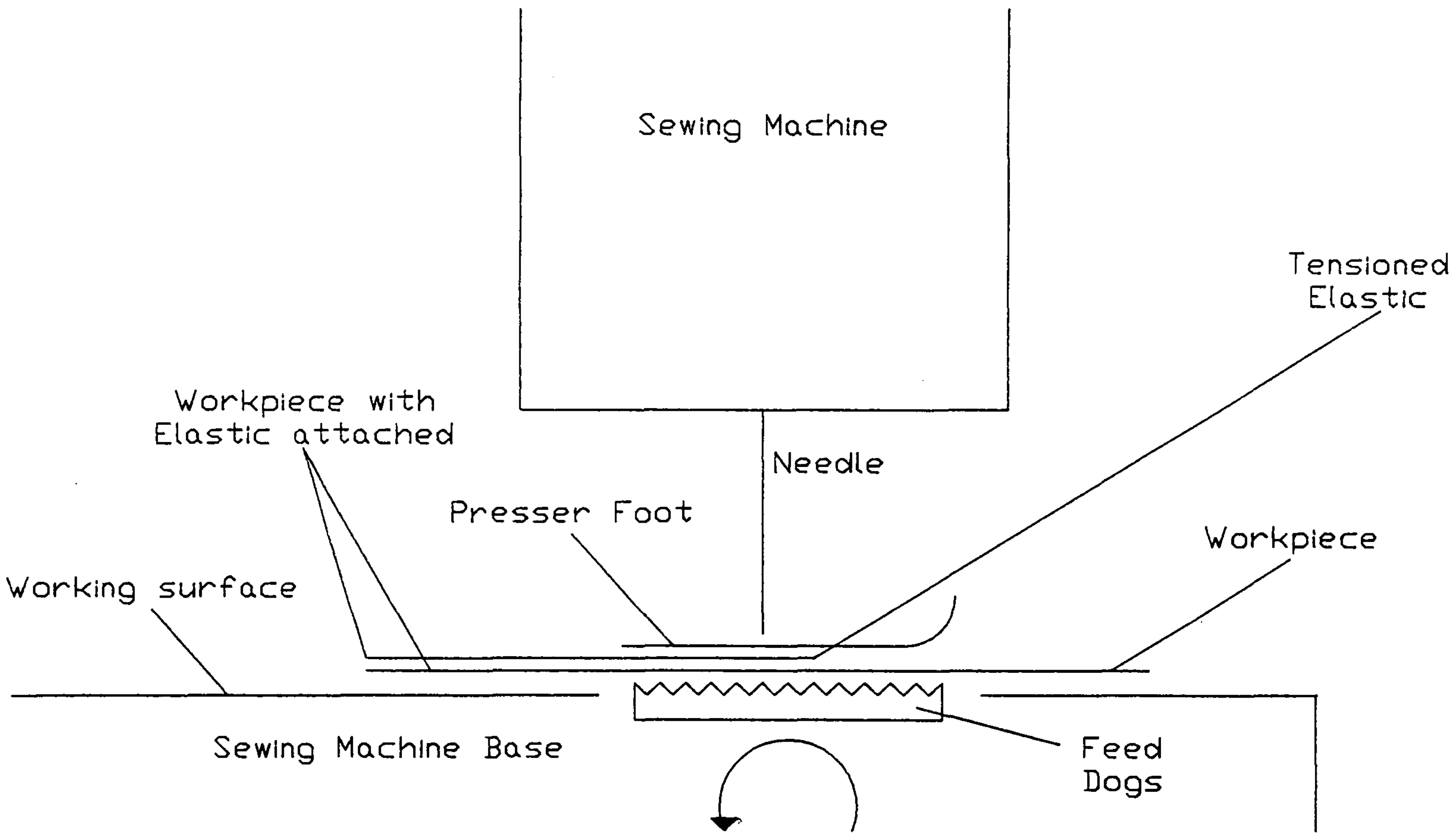

Figure 1. Sewing head of a typical sewing machine.

is fed between the presser foot and the feed dogs by the operator. As the feed dogs reach the highest point of their path, they grip the workpiece against the spring pressure of the presser foot, and so pull the workpiece through the sewing head as they continue to rotate. The proportion of the stitching cycle for which the feed dogs are above the surface of the sewing area determines the length of the workpiece which is pulled through the sewing machine as each stitch is formed. Thus the number of stitches per inch sewn on each workpiece can be set to the desired level by adjusting this parameter. This adjustment is usually carried out by a mechanic when setting up the machine for a new garment. The number of stitches per inch is usually determined by the type of fabric being used, and the aesthetic effect desired.

Once attached and free of the sewing head, the tape returns to its original length, resulting in a particular size of garment being produced. For different garments, different degrees of tension need to be imparted to the elastic tape being attached. The amount of tension required in the elastic is usually expressed in terms of the desired percentage stretch to be applied to the elastic. A desired 10% stretch in the elastic implies a length after tensioning of 110% of the original length of the elastic. Once attached to the workpiece, the elastic returns to its original length, reducing the workpiece dimension to 91% of its original length. The operator is usually told the original length of the workpieces being processed, and the required reduced length after attachment of the elastic, from which the desired percentage stretch in the elastic can be easily calculated as follows :

$$L_u = L_r$$

where L_r is the required reduced length of the workpiece, and L_u is the unstretched length of elastic. The extension required, e, is thus given by :

$$e = L_o - L_u$$

where L_o is the original length of the workpiece. Thus the required percentage stretch, T, is :

$$T = (e/L_u) * 100; \qquad (1.1)$$

To ensure correct sizing of the garment, control of the tape feed to the sewing machine is thus needed to maintain the elastic stretch at the required level.

1.2 Methods of elastic feed control

The earliest methods of tensioning the elastic tape consisted of a pair of spring-loaded nip rollers between which the untensioned tape was fed. Fig. 2 shows such an arrangement. Variations in tension were achieved by varying the spring pressure on the rollers. This was a very inaccurate and unreliable method of tensioning the elastic, since it relied on the tension in the elastic overcoming the spring pressure on the rollers. This tended to result in a stick-slip effect, causing an uneven distribution of tension in the elastic. In order to identify sub-standard garments, labour-intensive quality control techniques were employed. Samples from each batch of processed workpieces were taken at random. Visual inspection and measurement of these workpieces were carried out by an inspector, and unsatisfactory batches were re-processed if possible, or

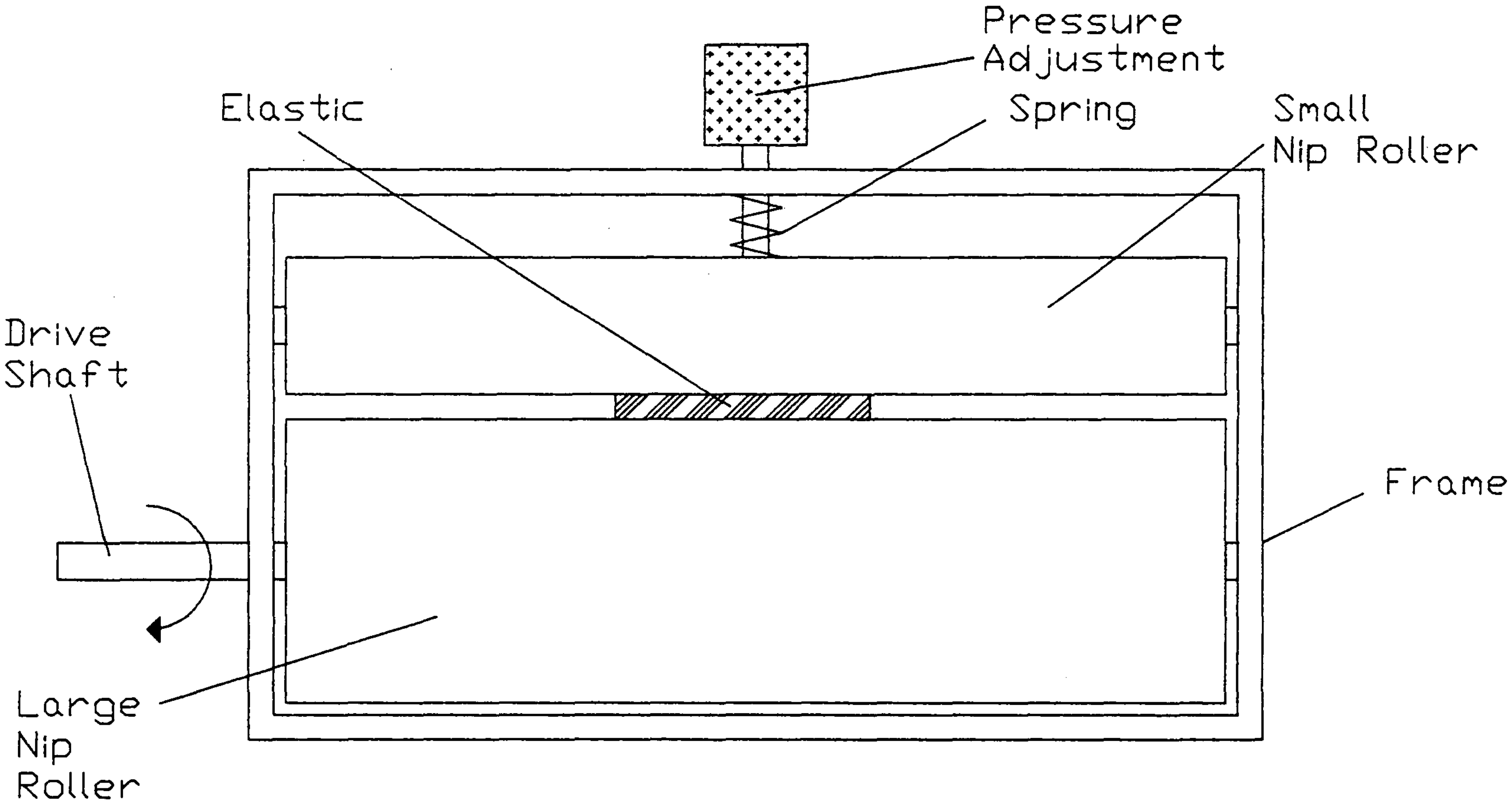

Figure 2. Spring-loaded nip rollers.

rejected.

Performance improvements were made by fitting a mechanical linkage between the larger of the two nip rollers and the sewing machine drive shaft via a gearbox. The rollers were driven via a system of gears, so that the length of untensioned tape fed between them as each stitch was formed was less than the length of workpiece fed through the sewing head. Tension in the tape between the rollers and the sewing head was thus produced by the difference in feed rates. This purely mechanical method of elastic tension control was, however, bulky and setting the desired tension was difficult and time-consuming.

With the advent of cheap, mass-produced microprocessors at the end of the 'Seventies, the opportunity arose for the development of more accurate electronic control systems for elastic feed. Such systems, apart from improving product quality and consistency, would need less maintainance and would reduce change-over times for different garments, important factors in an industry which has high volume production but more and more has to follow fashion trends rapidly to remain competitive with overseas imports.

Durham University was approached by Courtaulds Clothing Brands, then Lyle and Scott, in Gateshead, to design and develop an electronic elastic feed control system incorporating the features outlined in the previous section. The design specification also required the unit to be cheap, and simple to use, minimising the time required to train sewing machine operators and mechanics who have little or no electronics knowledge.

An open-loop electronic elastic feed control system was then developed. Elastic tension is maintained by controlling a stepper motor which drives a pair of nip rollers at a rate relative to the sewing machine speed. A sensor is attached to the machine to provide data on its speed. The sensor and motor effectively replace the mechanical linkage previously used. Normally, such systems do not 'know' what the absolute tension in the elastic is. They simply allow the operator to vary the rate at which the elastic is fed to the sewing machine by means of inputting a 'figure of merit' to the control system. The higher the number, the lower the feed rate relative to the machine speed, and thus the greater the elastic tension produced. Thus, when the operator changes to a new size, or a new garment, he has to run test pieces through his machine and adjust the 'figure of merit' on the control unit until the correct size is produced. The Durham system tried to overcome this problem by developing a control algorithm which allows the absolute tension in the elastic to be controlled.

It is believed that the open-loop control system developed at Durham [1] and currently in use in the garment industry, was the first of its kind. A block diagram of the system is shown in fig. 3. For correct operation,the control algorithm executed by the microprocessor requires accurate physical placement of the nip rollers at a known distance from the sewing head, and accurate data on the number of stitches per inch at which the sewing machine has been set to operate. The operator can then input the desired tension, and the algorithm will step the motor at a rate relative to the sewing machine speed to maintain the

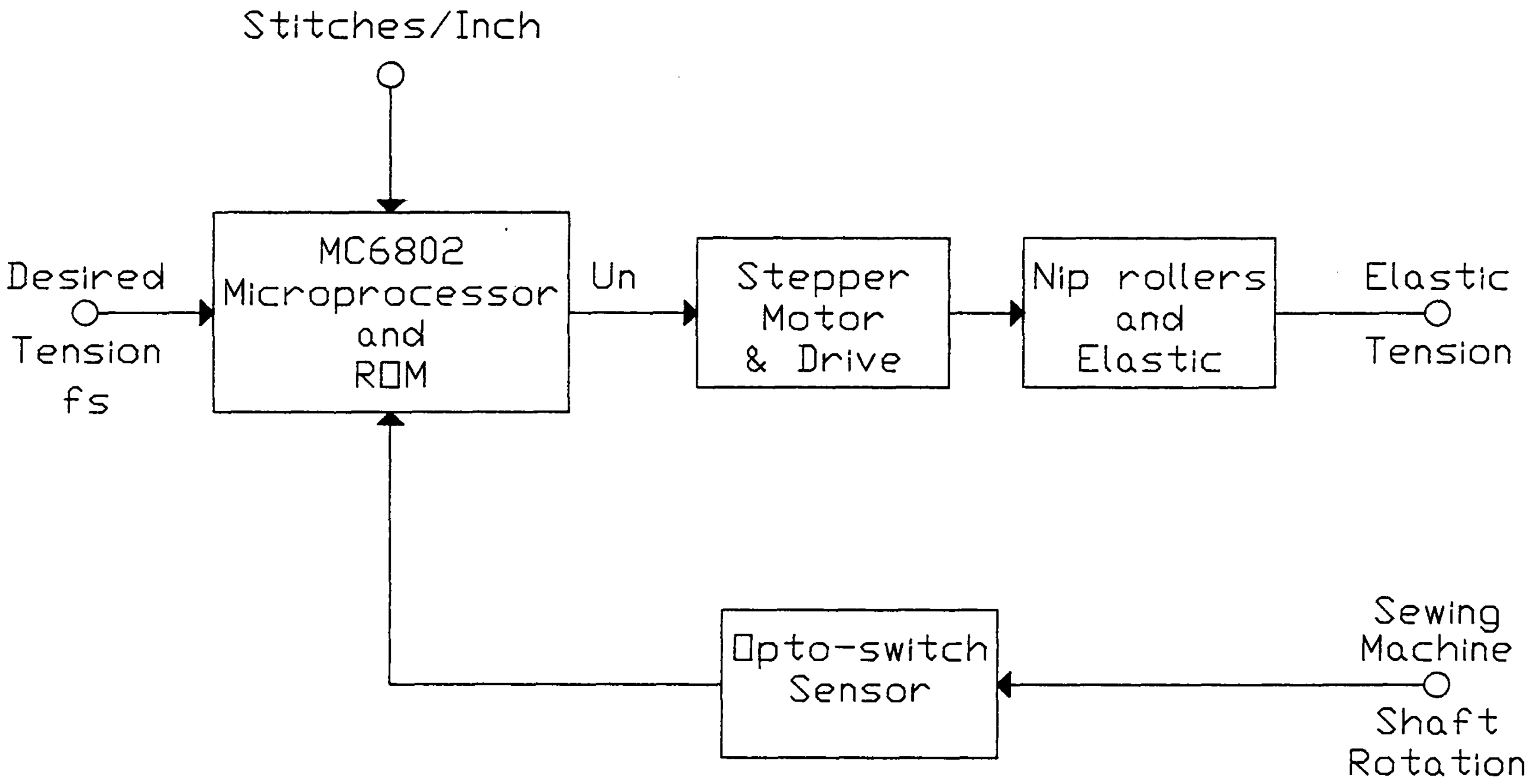

Figure 3. Block diagram of the open-loop control system.

tension.

This system, and other open-loop systems such as that manufactured by SAHL, represent the current standard of elastic feed control systems available on the market. Significant improvements have been made in type quality control of garments in whose manufacture these systems are used. Open-loop elastic feed systems, however, suffer from two main problems, which cause loss of tension in the elastic tape and hence an incorrectly sized garment. The first is slippage of the elastic tape through the nip rollers. This is caused by a build-up on the nip rollers of the powder with which the elastic tape is coated at manufacture to prevent it sticking to itself in its box, and to facilitate smooth feed through the nip rollers. The second problem is electrical noise causing false feed of elastic. It takes a certain time for open loop systems to recover from these conditions, resulting in one or more incorrectly sized garments depending on the severity of the error.

1.3 Layout of the thesis

The work undertaken for this thesis was aimed at the development of a closed-loop control system to overcome these problems. The work can be split in to four sections. Firstly, a thorough analysis was made of the open-loop system developed at Durham to enable its shortcomings to be fully understood. The knowledge gained could then be used in the development of the closed-loop system to overcome these shortcomings. This work is described in Chapter 2, and includes the derivation of the control algorithm used by the open-loop system, and analysis of practical data obtained from the system.

Secondly, a closed-loop control system was developed. The design features an algorithm which converts the desired elastic percentage stretch to a force. A tension control algorithm then uses that force as a set point for maintaining the elastic tension.

A general block diagram of the closed-loop system is shown in fig. 4. Feedback is provided by a specially-developed force transducer, which measures the axial force in the elastic tape being fed to the sewing machine. The tension control algorithm compares the measured force to the set-point force computed earlier. The resulting error signal is subsequently used to control a stepper motor, which drives a pair of nip rollers similar to those used by the open-loop system through which the elastic tape is fed, thus achieving closed-loop control. Chapter 3 contains a full description of this work, and includes discussion of the performance of the system based on test data.

The third stage was the development of a stand-alone unit which could be installed in the Courtaulds factory and tested in an industrial environment by their machine operators. The specification for the system called for economic construction, high reliability in a factory atmosphere which contains a high level of dust and fibres, a simple operator interface, and precise tension control independent from sewing machine parameters such as stitches per inch and physical positioning relative to the sewing head. A Motorola 68000 microprocessor was chosen to increase the processing ability of the system, allowing extra features such as automatic size-checking to be included

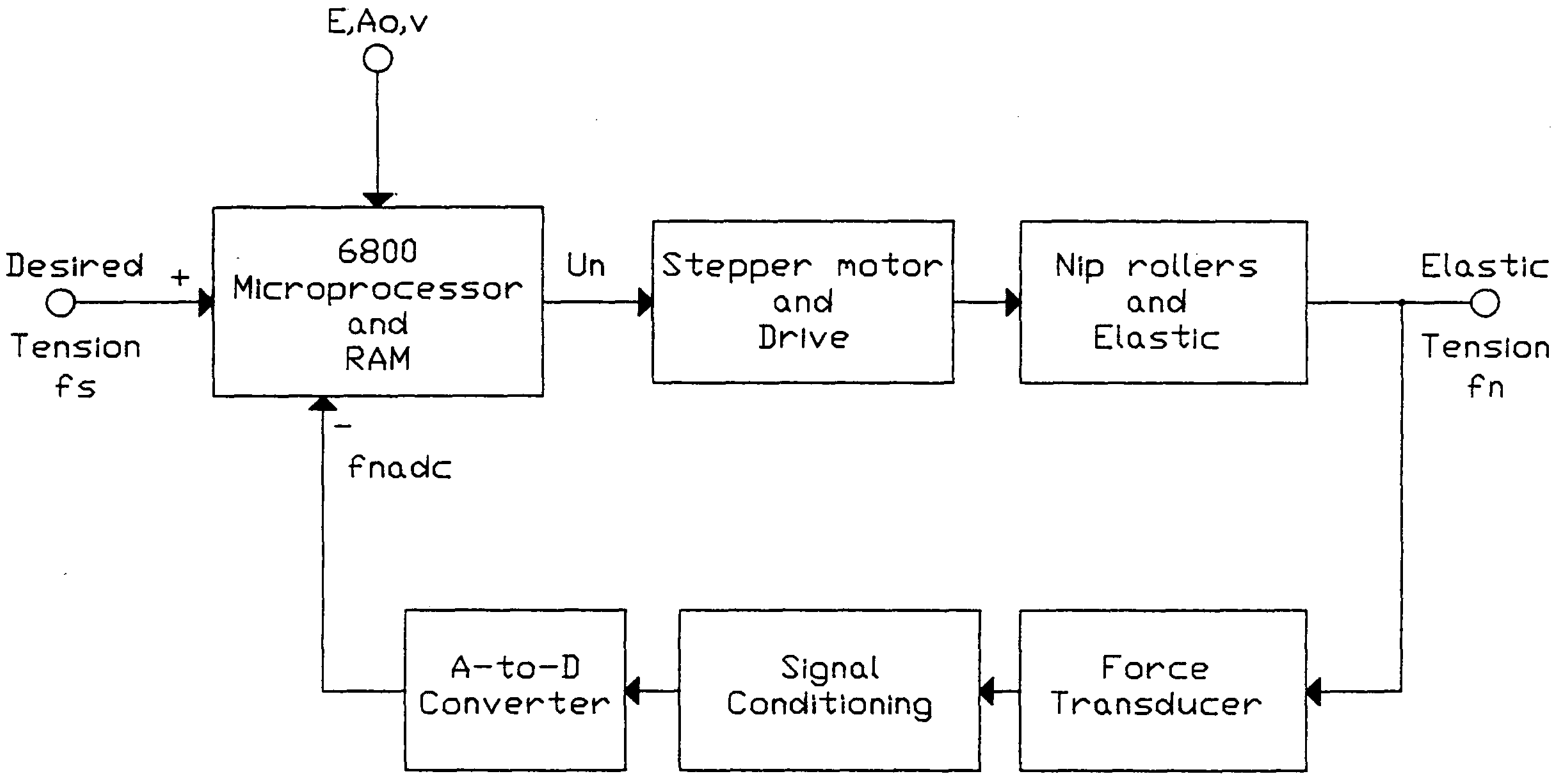

Figure 4. Block diagram of the closed-loop control system.

without compromising the tension control performance. A simple data terminal with an LCD display was used to provide the operator interface, allowing data to be input by the operator, and error messages and information to be displayed simply to the operator. The design and development of the stand-alone system is described in Chapter 4.

A comparison of the performances of the two systems is included in Chapter 5. Programs were written to simulate the elastic feed control performance of the two systems, allowing comparisons to be made over a range of operating parameters without having to alter the hardware. Comparisons of real data collected from operation of the two systems is also presented, and the benefits of using the closed-loop system are highlighted.

CHAPTER 2

THE OPEN-LOOP CONTROL SYSTEM

In this chapter, an open-loop control system for elastic feed is described. In section 2.1, the open-loop control algorithm for this application is derived and discussed. The following sections describe the software and hardware used to implement a commercial system based on the algorithm.

2.1 The control algorithm

An algorithm for the open-loop control of elastic feed was developed by I. Burdess [1]. It determines the rate of feed, R, of untensioned elastic to a sewing machine as a function of the following parameters:

R = fn(D,S,T,F)

D = distance from nip rollers to sewing head

S = number of stitches per inch produced by the sewing machine

T = required elastic percentage stretch for correct size of garment

1/F = feed of untensioned elastic through the nip rollers per feed operation

2.1.1 Derivation of the open-loop control algorithm

The distance D between the sewing head and the nip rollers is assumed to be 10".

Consider a piece of elastic pre-tensioned to a percentage stretch T. The strain in the elastic, e/L, is given by equation 1.1, which is

$$e/L = (T/100)$$

Therefore,

$$e = L.(T/100) \qquad (2.1)$$

Since the total length of the stretche delastic is 10",

$$L + e = 10'' \qquad (2.2)$$

So, substituting (2.1) in (2.2) gives

$$L + L.(T/100) = 10$$

$$L.(1 + (T/100)) = 10$$

$$L = 10.(1 / (1 + (T/100)))$$

When a stitch is formed, a length of tensioned elastic is attached to the material being processed. For a machine operating at C stitches per inch, this length, dD, is given by

$$dD = 1/C$$

The removal of this length dD of tensioned elastic causes an increase in the percentage stretch within the 10" of elastic remaining. The untensioned length, l, of dD is given by

$$l = (1 / (1 + T/100)).(1/C)$$

This corresponds to a decrease in the untensioned length of elastic which has been stretched to 10". Thus the new untensioned length of the 10" of elastic, L', is

$$L' = (10.P) - ((1/C).P)$$

$$L' = (10 - (1/C)).P$$

where

$$P = (1 / (1 + (T/100)))$$

and the new extension, e', is

$$e' = 10 - L'$$

Therefore, the new percentage stretch, T', is given by

$$T' = (e'/L').100$$

To return the elastic to the desired percentage stretch, T, the correct amount of tensioned elastic needs to be fed in by the control system to return the new untensioned length L' of the 10" of elastic to L. The amount of tensioned elastic needed to achieve this is the difference between L and L', given by

$$L - L' = (10.P) - (10.P - (1/C).P)$$

The two terms $(10.P)$ cancel, leaving

$$L - L' = (1/C).P$$

This is the length of elastic at percentage stretch T required to restore the stretch from T' to T. This corresponds to an untensioned length of elastic being fed in to the nip rollers of

$$(L - L').P = (1/C).P^2 \qquad (2.3)$$

Thus, if one motor step feeds (1/F)" of untensioned elastic in to the nip rollers, the required number of motor steps to maintain the percentage stretch at T is

$$\begin{aligned} S &= ((1/C).P^2) / (1/F) \\ &= (F/C).P^2 \\ &= (F.P^2)/C \qquad (2.4) \end{aligned}$$

However, the number of motor steps to be executed is not likely to be an integer number, so an equation had to be found based on equation (2.4) which would take in to account the fractions of a step required for maintaining the stretch at the correct level.

If, instead of calculating the number of motor steps required per stitch formed, a fixed number of steps were executed for each feed operation and a counter incremented each time a stitch were formed, the counter value could be used to determine whether a feed operation was necessary. When the counter overflowed, sufficient motor steps would have been executed to keep the elastic tension at the correct level. To ensure maximum accuracy, the counter should use the maximum resolution available for the system, which, in the case of the 6802 processor being used, is 8 bits (0 - 255).

The boundary condition for correct operation is the minimum number of stitches per inch at which the system can maintain the elastic percentage stretch at 0% if a feed operation is executed every time a stitch is formed, with the sewing machine running at its maximum speed. This is determined by the dimensions of the nip rollers used, and the number of steps executed per feed operation.

The diameter of the roller driven by the stepper motor is 1.25". The motor performs 200 steps/rev, therefore the feed of elastic per step is

(1.25 * Pi) / 200 = 0.0196"

If three steps are executed by each feed operation, then

3 * 0.0196" = 0.059"

of untensioned elastic would be fed through the nip rollers. Two such feed operations are executed per stitch, since the opto-switch attached to the sewing machine senses both the upward and downward motion of the needle bar, therefore the total feed per stitch is

$$2 * 0.059" = 0.118"$$

With a maximum sewing speed of 5200 rpm, or 86.7 rps, the nip rollers will feed in

$$86.7 * 0.118" = 10.23"$$

of elastic per stitch. With the machine running at 86.7 rps, the minimum number of stitches per inch at which the system can maintain the elastic tension at 0% is

$$(86.7/10.23) = 8.48 \text{ stitches/inch.}$$

So, under these conditions, a feed operation must be executed every time a signal is received from the opto-switch. For this to happen, the counter mentioned earlier must be incremented by 0 after each feed operation. Now, equation (2.4) has its maximum value under these conditions, since F is a constant, C is a minimum therefore 1/C is a maximum,and

$$P = (1/(1+T/100))$$

$$= 1 \text{ for } T = 0\%.$$

so

$$P^2 = 1$$

Thus

$$(F.P^2)/C = 1$$

To make the counter increment 0, 1 must be sutracted from the result of this equation. This gives us a general equation for the increment, which takes the form

$$\text{increment} = 1 - (F.P^2)/C$$

which gives an increment <= to 1 for all operating conditions.

To normalise this to the maximum resolution of the 6802, both sides need to be multiplied by 255, giving the final form of the equation,

$$V2 = V1.(1 - ((F.P^2)/C)) \tag{2.5}$$

where V2 is the increment constant

V1 is the base value, in this case 255 (hex FF).

The P^2 term has a value of <= 1, which is then multiplied by 255 to give values which are stored in a look-up table. The maximum value is that for 0% stretch, which gives $FF in the table. The minimum value used is that for 50% stretch, which is the greatest stretch at which the system can operate. It gives a value for P^2 of 0.444, corresponding to the last entry in the table, $72.

The (F/C) term has its maximum value of 1 when the sewing machine is set at the minimum number of stitches/inch at which the system can maintain the elastic stretch. A second look-up table was obtained by multiplying (F/C) by 255.

As well as containing the P^2 data, the same look-up table also contains values for the number of motor steps necessary to pre-tension the elastic, i.e. to bring the elastic stretch up to the required percentage assuming 0% stretch to start with, and to de-tension the elastic, i.e. to reduce the tension to 0% stretch assuming it is at the desired level to start with.

The number of de-tensioning steps required is calculated in the following way. Assuming the elastic to be at a desired percentage stretch T, the length of elastic, L', which has been stretched by T is given by

$$L' = D / (1 + (T/100))$$

where D is the distance between the sewing head and the nip rollers. For the tension in the elastic to be reduced to 0%, L′ must be increased until it is equal to D. Thus the length of untensioned elastic, L_u, which must be fed through the nip rollers is

$$L_u = D - L'$$

and so the number of motor steps, N_d, which need to be executed to feed L_u" of untensioned elastic is

$$N_d = L_u / (1/F)$$

The values for P^2, N_p, and N_d are stored in groups of three in the same look-up table.

From the derivation, it can be seen that the greater the increment constant, the smaller the number of steps executed by the stepper motor in relation to the sewing machine operation, and therefore the greater the tension at which the elastic is fed.

All the calculations of rate of feed rely on a precise physical relationship between the nip rollers and the sewing head. If the distance, D, between them is not exactly what it should be, the elastic tension will either be too large, if the distance is to great, or too small, if the distance is too small, as the elastic is fed into the sewing head.

An accurate value of the number of stitches per inch at which the sewing machine to which elastic is to be fed is operating must be available to the algorithm. If it is not, then the effect is the same as the incorrect setting of D.

The elastic fed to the nip rollers is assumed by the algorithm to be untensioned. If the elastic snags on anything between its box and the nip rollers, it has some tension imparted to it. The calculation of equation (2.3) is thus incorrect, and less elastic is fed in through the nip rollers than required. Thus the tension in the elastic being fed to the sewing machine is greater than that desired, and incorrect sizing could result.

One of the major advantages of this algorithm, however, is that its operation is totally independant of the type of elastic being used. It therefore requires no information on the physical proerties of the elastic, and once set up accurately for the machine to which it is attached, should need no adjustment other than changing the tension setting for different garment sizes or types of elastic.

2.2 System Implementation

2.2.1 Hardware Description

The open-loop system architecture is shown in fig. 5, and th installation of the system is shown in fig. 7. Read-only memory (ROM) stores the program which directs the actions of the microprocessor. An operator interface allows data to be input to the unit and warnings to be displayed to the operator. Input interfacing allows data to be read in by the microprocessor from the operator and from the sewing machine, from which control signals can be computed by the control algorithm. Output interfacing allows these signals to be passed on to external devices, along with any necessary warnings to the operator. The interrupt interface allows control of the elastic tension to be switched on and off.

The microprocessor used is a Motorola MC6802 [2]. At the

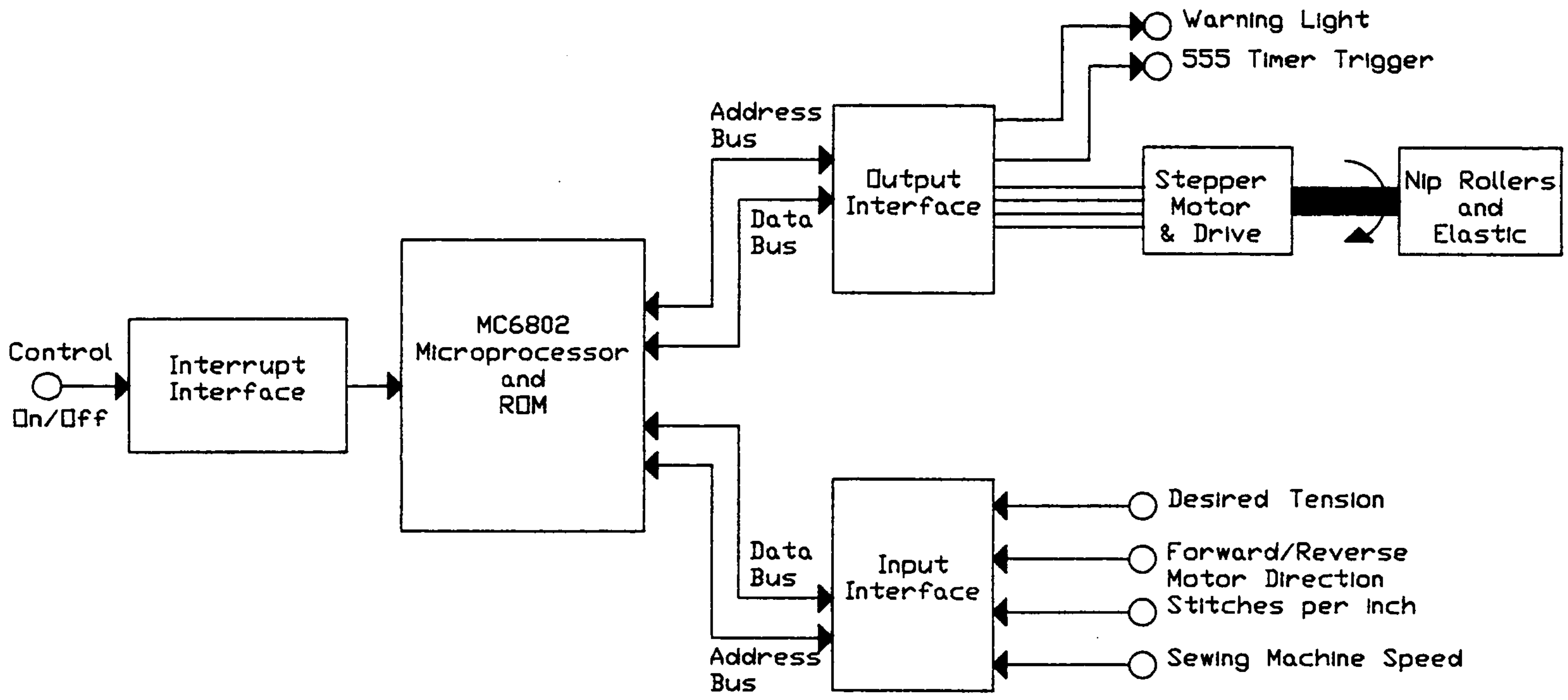

Figure 5. Open-loop control system architecture.

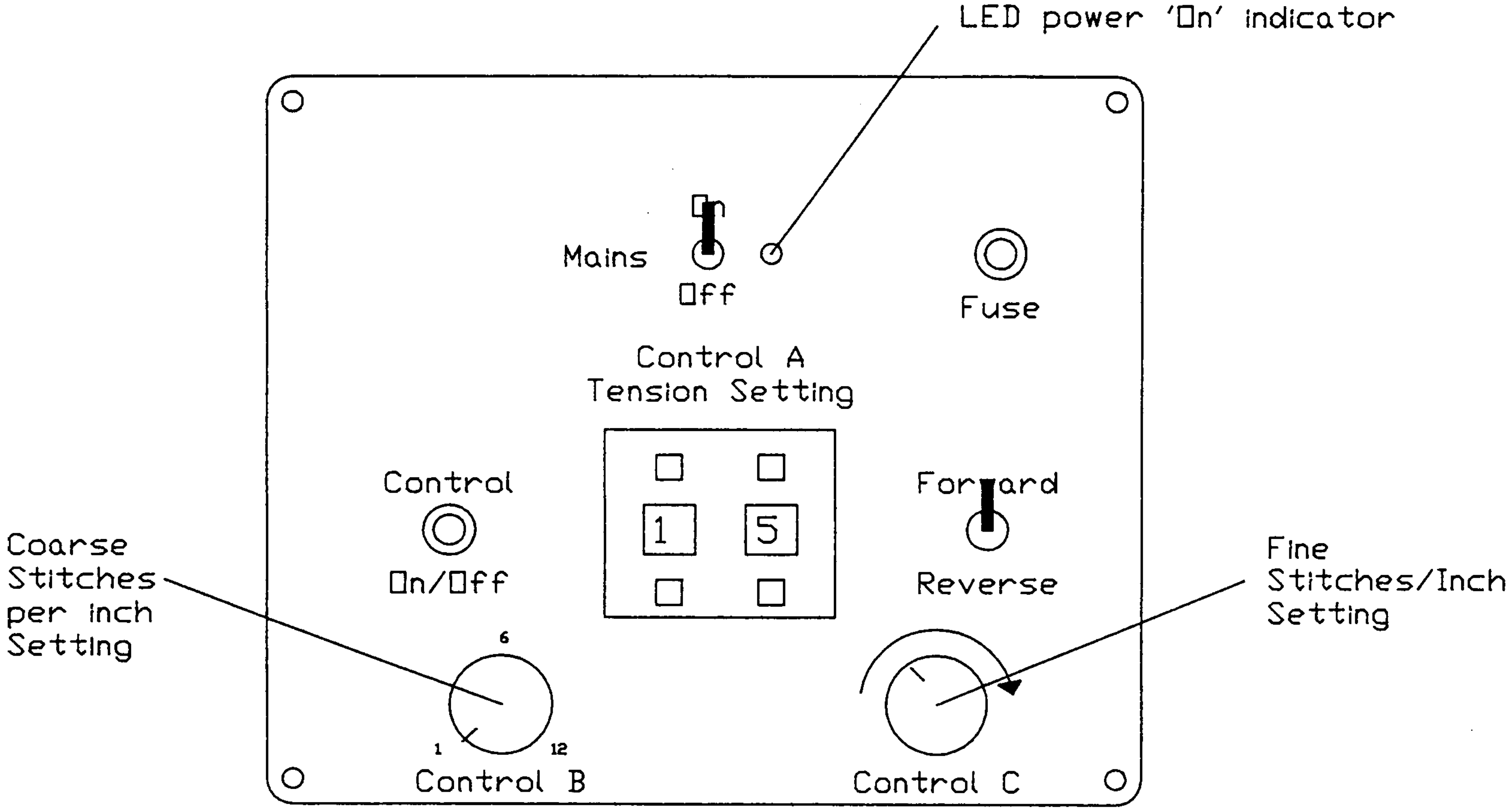

Figure 6. Front panel of the open-loop control system.

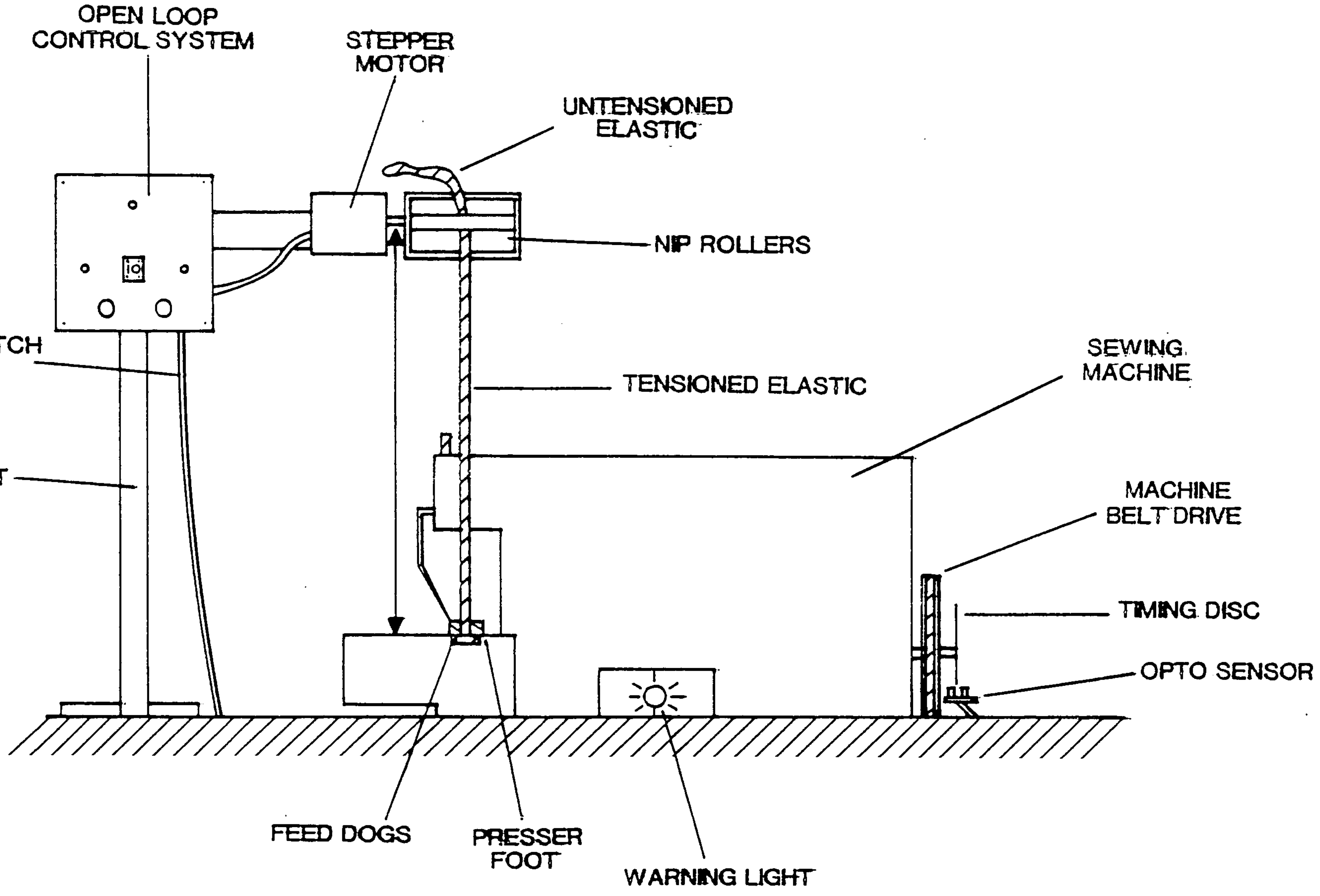

Figure 7. Installation of the open-loop control system.

time of development of this system, it was the best device available for real-time control applications at an economic price. It has 128 * 8 bits of on-board volatile random-access memory (RAM) which is used to store data during operation, and an on-board clock which, with an external crystal, provides a 1 Mhz operating frequency. Program memory is a 4k * 8-bit EPROM, type TMS 2532.

A major design feature is the operator interface. It consists of the controls provided on the front panel of the unit, and a separate warning light which can be mounted on the sewing machine in the most convenient position for it to be seen by the operator whilst sewing garments.

The front panel, shown in fig. 6, has six controls. Four are used by the operator, and two are used by a mechanic. The four operator controls consist of a mains on/off switch with an LED power 'on' indicator, a pair of binary-coded decimal (BCD) thumbwheel switches for setting the required percentage stetch in the elastic for the size of garment being produced, a tension control on/off push-button switch which allows pre-tensioning and control or de-tensioning of the elastic or manual control of the stepper motor, and a forward/reverse select switch for the stepper motor direction under manual control. Once the system has been installed by a mechanic, two further controls allow him to input to the system the number of stitches per inch produced by the sewing machine to which the system is attached.

The BCD thumbwheel switches allow the operator to tell the system what percentage stretch in the elastic isrequired. Each thumbwheel switch contains 4 smaller switches, each of which represent one bit of a BCD digit. The switches are normally open, so 10k pull-up resistors are needed on the output of each one. When the switches are closed, the outputs are pulled down to ground. The correct combination of switches are opened or closed to present the correct combination of bits on the outputs to give the BCD digit corresponding to the digit visible to the operator on the front of the unit.

The tension control on/off switch is the only device connected to the interrupt interface. When operated, an interrupt request is sent to the 6802. This allows the elastic to be tensioned and controlled or de-tensioned with alternate single operations of the switch at any time during operation. If the switch is held down, the motor is driven continuously in the direction indicated by the direction switch.

The warning light tells the operator whether the elastic has been pre-tensioned and is ready to control the elastic tension, or if the sewing machine has been operated before the elastic was pre-tensioned and is thus receiving elastic at the wrong tension.

The simplicity of the operator interface made the introduction of the unit into a largely manual production line quick and easy, since the operator needed only the minimum of instruction on the use of the unit.

The two controls for setting the stitches per inch at which the sewing machine is running operate as follows. Control B in fig. 7 is a twelve-position switch which is connected to a resistor ladder, consisting of twelve 2k2 resistors. This gives a coarse adjustment of 1 to 12 stitches per inch by switching in one or more of the resistors in the ladder. A 2k potentiometer, control C, is connected in series with the coarse control to give fine adjustment of the stitches per inch setting. The resulting resistance obtained from the coarse and fine adjustments is used to determine the length of a single shot pulse from a 555 timer operating in monostable mode, by altering the time constant of an R-C network [9]. The monostable can be triggered by the control program, and the output pulse fed back to the control program. The length of the pulse can then be measured, giving an indication of the stitches per inch setting to the control program.

For monostable operation of the 555 timer, the pulse length is given by :

$$T = (1.1 * R_f * C) \text{ seconds} \qquad (2.7)$$

where R_f is the resistance set by the two controls, and C is the value of a capacitor. The minimum value of R_f is 1k, to ensure a pulse, and the maximum value is 27.2k. C is a 20nf capacitor, so C = 20nf. From equation (2.7), the minimum and maximum pulse lengths are

T_{min} = 22 microseconds

T_{max} = 598.4 microseconds.

The length of the pulse is monitored by the control program by incrementing a counter until the output state of the 555 timer changes. The counter value is used to reference a look-up table stored in the ROM from which the stitches/inch setting of the sewing machine is obtained. The values in the look-up table are again pre-calculated based on the pulse times calculated from equation 2.7.

The stepper motor drive consists of 4 independant, identical circuits, one for each phase of the motor. Each circuit contains a TIP121 Darlington complimentary power transistor. When switched on under program control, the transistor allows current to flow through the motor winding towhich it is connected, and so energises that phase. When the correct combination of phases is switched on or off, the motor can be stepped clockwise or anticlockwise as desired.

The input and output interfacing of these devices to the processor is provided by the two 8-bit ports, A and B, of a Motorola M6821 peripheral interface adapter (PIA) [2]. There are six outputs, bits 2 to 7 (PB2-PB7) of port B. PB2-PB5 are used to convey the control signals for the stepper motor phases to the stepper motor drive circuits. PB5 controls the operator warning light, and PB7 controls the 555 timer, and so allows the number of stitches per inch setting to be read under software control. All the outputs are buffered by the six inverters on a 74LS16 logic buffering chip. The inverters have open collector outputs, so 10k pull-up resistors are included for each ciruit.

There are twelve inputs, PB0, PB1, CA1, CA2 and PA0-7. PA0-6 are connected to the two thumbwheel switches. PB0 is connected to the motor direction select switch. The output of the 555 timer giving the stitches per inch switch settings is fed into PB1. CA2 is connected to a slotted opto-switch which senses each revolution of the sewing machine drive shaft and hence each stitch as it is made.

The tension control push-button is connected to both PA7 and CA1. A change of state on CA1 causes the 6821 to send an interrupt request to the 6802. If interrupts have been enabled, the interupt service routine is then entered. The signal level on PA7 can be read as a normal input at any time.

A +5v regulated power supply is provided for the TTL circuits, and a +24v supply for the warning lamp and the stepper motor drive circuits. Both are full wave rectified.

A second 555 timer is used to generate a reset pulse when power is applied to the unit. Again, the monostable mode of operation is employed. An R-C network, consisting of a 330k resistor and a 100nf capacitor, ensures that the trigger input of the 555 is held below 1/3 of its + supply rail,thus triggering the monostable. The time constant of the R-C network is 33ms, so the trigger input is returned to > 1/3 of the + supply before the output pulse can complete, ensuring correct operation. Using equation (2.7), with Rf now 330k and C 470nf, a pulse length of 170 ms is obtained. It is inverted to give the logical low signal necessary to reset the logic devices. The reset signal is connected to the reset pins on the 6821 and the 6802. On the low-to-high transition of the reset line at the end of the monostable pulse, the 6802 loads the address stored in the last two memory locations it can access, FFFE and FFFF, in to its program counter. Program execution then begins at the address in the program counter, which is C000 hex.

2.2.2 The Program

Software for the system was developed on a Cifer 2864 computer in using 6800 assembly language, and S-format machine code object files were generated by an Avocet Systems 6800 cross-assembler V2.0. A Stag PROM programmer was used to blow the machine code in to the TMS2532 EPROMs. A flow chart for the program is given in figs. 8a and 8b.

On power-up, all the temporary registers are set to their initial conditions. The stack pointer is loaded with the last available RAM location, $7F. Each bit of the PIA ports A and B is initialised to either an input or an output as required by the description in section 2.2.1. The port A Control Register (CRA) is initialised for the tension control push-button and the opto-switch inputs, CA1 and CA2 respectively, to be positive edge-sensitive, and the CA1 interrupt request is enabled. Any positive edge detected on either of these inputs will set a flag corresponding to that input in CRA, which can be read by the program. In addition, the active transition on CA1 sends an interrupt request to the processor.

The first stage of the control program is then entered. The warning light is flashed on and off to indicate that the control program is running, but the elastic has not been tensioned ready for feeding. The CA1 and CA2 flags in CRA are monitored, waiting for either the control push-button to be pressed, or for a positive edge from the sensor detecting the sewing machine operation.

If the push-button is pressed first, the system reads in the desired percentage stretch from the thumbwheel switches set by the operator. The required number of motor steps to tension the elastic to the desired percentage stretch is obtained from the pre-calculated data in the relevant look-up table. The warning light is switched off while the motor is stepped to tension the elastic. Once the tensioning is complete, the

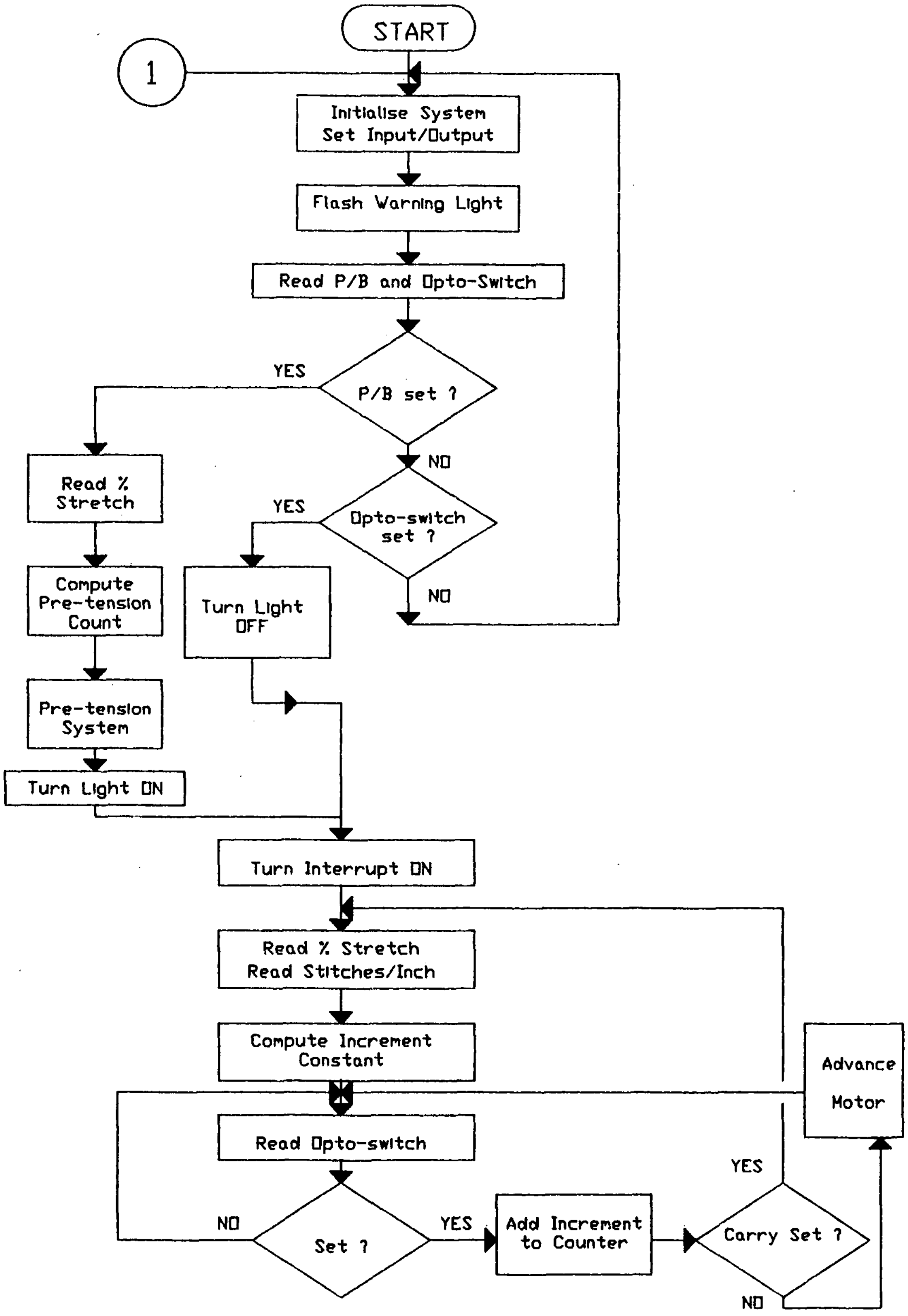

Figure 8a. Open-loop main control program flow chart.

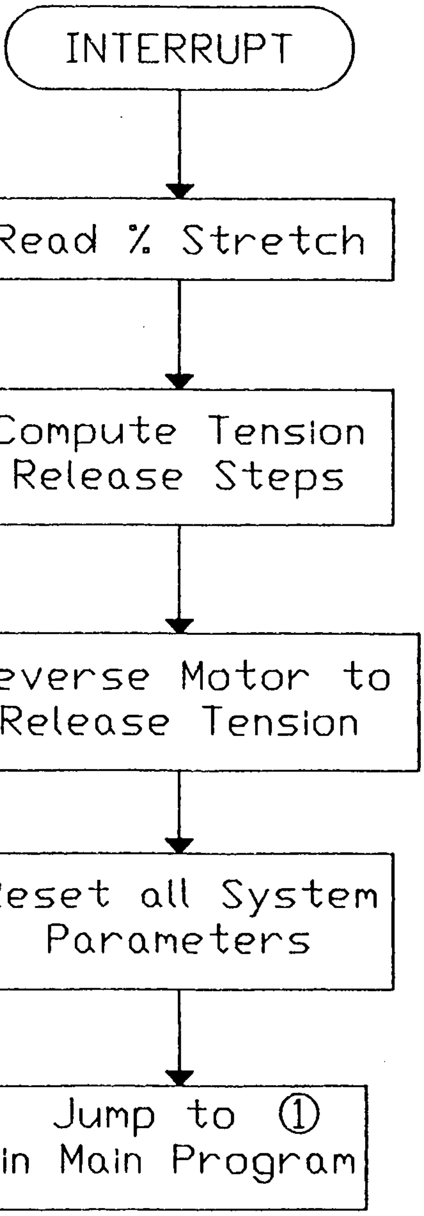

Figure 8b. Open-loop control program interrupt handling flow chart.

warning light is switched on permanently to indicate that the elastic has been tensioned and the system is ready to begin controlling the feed rate.

The program then enters the feed control algorithm, described in section 2.1, which decides whether or not to perform a feed operation to maintain the elastic percentage stretch. Sewing machine operation is indicated by the opto-switch sensor. The tension setting thumbwheel switches are also regularly monitored for any change, and a new increment constant is calculated accordingly.

When the program enters the feed control algorithm, the 6802 interrupt line is enabled. Thus, if the tension control push-button is pressed, the background program is interrupted by the interrupt request from the PIA. An interrupt service routine is entered, which determines the action to be taken. A delay loop is executed, after which the state of PA7 is checked. If it is still high, the push-button is being held down. The status of the forward/reverse switch is then read in, and the motor is stepped continually in the direction indicated until the push-button is released. Processing is then returned to the first stage of the control program, and the warning light starts flashing.

However, if on leaving the delay loop PA7 is low, the push button must have been released. The tension release subroutine is then executed. The number of motor steps necessary to reduce the elastic stretch from the current level to 0% is read in from the look-up table. The warning light is switched off while the motor is stepped to release the tension in the elastic. Again, processing is returned to the first stage of the control program.

If the operator attempts to attach elastic to a workpiece before pressing the tension control push-button to pre-tension the elastic, an error condition is encountered. Sewing machine operation is sensed by the opto-switch, and the program jumps to the feed control algorithm immediately. The warning light is switched off so the operator knows that the tension has not been set, and the feed control algorithm attempts to feed elastic at the stretch indicated by the thumbwheel switches. This action prevents the elastic from becoming overstretched and possibly breaking.

2.3 System performance

There are a number of these systems in use both at Coutaulds Clothing Brands in Gateshead and at factories in the United States. The feedback from Gateshead has provided useful information on the operation of this system and the problems encountered with its use in an industrial environment.

Initial tests in the laboratory used a second pair of nip rollers driven by an independantly controlled stepper motor to simulate the sewing operation, which is an intermittent feed operation. The second nip roller pair was stepped in such a way as to simulate a sewing machine stitching at 10 stitches/inch.

The range of percentage stretch required for garment manufacture is generally between 10-30% as described in Chapter 1. Although the unit was capable of controlling stretch up to 50%, it was decided that the upper limit was never going to be used, so tests were carried out mainly using values between 10 and 40%.

These results can be seen in tables 1 and 3. For the results in table 1, a grey-coloured elastic was used. For the results in table 3, a white-coloured elastic was used. It can be seen that the stretch could be controlled to within 2%. This would correspond to a variation in garment size, for a typical underpant front or back panel waistband of 410mm having elastic attached at a nominal 28% stretch, of less than 5mm. This is well within the normal Courtaulds Clothing Brands tolerance for garments of ± 10mm.

Systems installed at Courtaulds, however, suffered from two problems. The first was 'chalking', which affects rubber tape. The tape is stored with a coating of chalk to prevent it sticking to adjacent tape under the weight of tape above. During operation, this chalk builds up on the nip rollers as the tape is fed through. This causes slippage of the elastic through the nip rollers, and a corresponding loss of tension in the tape fed to the sewing machine. An incorrectly-sized garment could thus be produced, which would not necessarily be spotted during quality control inspection. To counter this problem, the rollers had to be cleaned frequently, requiring the operator to stop work, thus losing production.

The second problem was not discovered until during the development of the closed-loop system because of the way the system was used in the factory. When changing elastic or garment size, the operator would set the desired tension on the unit, andfeed a workpiece through the sewing machine and attach elastic to it. If the size produced was incorrect, the tension setting

Sample	Percentage Stretch (%)				
	Desired	Initial	@ 150mm	@ 300mm	@ 450mm
A	10	10	9	9	11
B	20	20	20	20	20
C	30	30	28	28	30
D	30	30	29	29	30
E	40	40	39	40	39

Table 1. Open-loop system results with grey elastic.

Sample	Percentage Stretch (%)				
	Desired	Initial	@ 150mm	@ 300mm	@ 450mm
A	10	10	10	9	10
B	20	20	20	20	20
C	30	27	28	30	30
D	30	30	30	30	30
E	40	40	41	39	40

Elastic : Grey - E * Ao = 6.24, v = 0.385

Table 2. Closed-loop system results with grey elastic.

Sample	Percentage Stretch (%)				
	Desired	Initial	@ 150mm	@ 300mm	@ 450mm
A	10	10	9	10	10
B	20	20	20	20	20
C	30	26	30	29	30
D	30	29	30	30	30
E	40	42	38	39	39

Table 3. Open-Loop System Results with White Elastic.

	Percentage Stretch (%)				
Sample	Desired	Initial	@ 150mm	@ 300mm	@ 450mm
A	10	10	10	9	10
B	20	20	20	20	20
C	30	29	30	30	30
D	30	30	30	30	31
E	40	40	40	40	41

Elastic : White - E * Ao = 7.86, v = 0.40

Table 4. Closed-loop system results with white elastic.

would be increased or decreased to compensate for the error, and another workpiece fed through. This process would be repeated until the correct size was achieved. The final tension setting on the unit did not necessarily correspond to the desired tension. The tension control then became merely a 'figure of merit', which did not directly correspond to the actual tension in the processed workpiece. The cause of this error is tension imparted to the elastic as it is fed into and through the sewing machine by friction between the elastic and the feed slot in the top of the presser foot. The elastic attached to the workpiece is thus at a greater tension than that fed in to the sewing amchine, and the operator has to reduce the tensionsitting on the unit to compensate. measurements made both from the open-loop systems and the closed-loop system in the laboratory bore this out. The difference between the actual tension of the attached tape and the tension of the tape fed in to the sewing head was as much as 30 %.

A comparison of the open-loop and closed-loop system preformances follows in Chapter 5.

CHAPTER 3

THE CLOSED-LOOP CONTROL SYSTEM

To overcome the problems encountered with the open-loop system, it was decided to develop a closed-loop feed control system. The derivation of the control algorithm is presented in section 3.1. The following sections describe the design and development of a system to implement the algorithm, and the development of a force transducer to provide feedback of the elastic stretch.

3.1. The control algorithm

For closed-loop control to be achieved, it is necessary to have some form of feedback of the parameter to be controlled, or of a parameter which is related to it by some expression. In this case, the parameter to be controlled was the percetage stretch in the elastic tape being fed in to a sewing machine.

The first feedback method to be considered was to mark the elastic in some way at known intervals, and use some form of transducer to sense the markings as the elastic was fed in to the sewing machine. The stretch in the elastic between the nip rollers and the sewing head could then be calculated from the number of motor steps executed between each pair of marks and the diameter of the nip rollers.

Four problems were identified which caused this approach to be rejected. Firstly, it would be difficult to mark the elastic at precise intervals in its unstretched state whilst feeding it to the nip rollers without imparting some tension to it. It would be possible to mark the elastic as part of the manufacturing process, but this would require a change in the manufacturing process, and a corresponding increase in the price of the elastic for the extra process. Since elastic is used in large quantities, any cost increase would be unacceptable.

Secondly, the time constant associated with adjusting for variations in the stretch would be determined by the separation of the marks on the elastic. Any variation in stretch could only be recorded as each mark was sensed. If the marks were too far apart, the system would be slow to respond in changing the feed rate. If the marks were too close, vibration and other noise factors could cause instability and possibly oscillation of the control loop.

Thirdly, some elastic tapes, such as that used on waistbands of underpants, is visible, and so any markings would have to be invisible to the naked eye.

FInally, whatever was used to mark the elastic would have to be colourfast to prevent discolouration of the garment when washed.

The method chosen was to monitor the axial force in the elastic with a specially-designed force transducer, and to use the feedback from it as the basis for calculating the instantaneous strain and hence stretch in the elastic. It was considered that this method would provide a much faster response to variations than the previous technique, and could be implemented using similar hardware to the open-loop system with the addition of a force transducer.

There are two types of elasticated material which are attached to garments. One is rubber tape, which is normally sewn in to the binding or seam on waistbands and cuffs to provide a good fit, and was the material with which this project was mainly concerned, since Courtaulds sew it in to the legs and waistbands of their 'Y'-front range of underwear. The second is woven to form a material which behaves in much the same way as the rubber tape, but looks better and can therefore be used on exposed surfaces of the garment. Both materials behave as elastomers, which have high elasticity and hence their cross-sectional area reduces significantly when stretched. Hooke's law relates stress and strain of a material, but assumes no significant reduction in cross-sectional area of the material under stress.

An algorithm was therefore developed for this purpose which relates the axial force, f, in the elastic to the percentage stretch T. The relationship is given by :

$$f = E.A_o.(T/100).(1 - v.(T/100))^2 \qquad (3.1)$$

where :

E = Young's modulus for the elastic in N/m^2

Ao = the unstretched cross-sectional area of the elastic in m^2

v = Poisson's ratio for the elastic

3.1.1 Derivation of the closed-loop control algorithm

The parameter to be controlled is the percentage stretch, T, in the elastic being fed to the sewing machine, as described in

Chapter 1. The strain required to produce this stretch is given by :

$e/L = T/100$ (3.2)

where

L = original untensioned length of the elastic

e = extension of elastic

The stress present in the elastic, S, is given by :

$S = f/A$ (3.3)

where

f = axial force applied to the elastic

A = cross-sectional area

For a Hookean material, Young's modulus (reference 3) is defined as the ratio of stress to strain, and is given by :

$E = (f.L) / (A.e)$ (3.4)

where

E = Young's modulus for the material

f,L,A, and e are as above.

For elastomeric materials, however, any strain is accompanied by a significant reduction in cross-sectional area, and thus Hooke's law does not apply directly [3]. Poisson's ratio (reference 3) is defined as the ratio of lateral contraction to longnitudinal strain, and is given by :

$v = ((W_o - W)/W_o) / (e/L)$ (3.5)

where

W_o = unstretched width of the material

W = width of material with strain applied

e and L are as for equation 3.2

Rearranging equation 3.5 gives

$$v.(e/L) = 1 - (W/W_o)$$

Therefore

$$W / W_o = 1 - (v.(e/L))$$

Multiplying both sides by Wo thus gives

$$W = W_o.(1 - (v.(e/L))) \qquad (3.6)$$

This equation gives a relationship between the reduction in width of the elastic to the strain applied to it. The thickness of the material also reduces, therefore to give the reduction in area, this term is squared, giving

$$A = A_o.(1 - (v.(e/L)))^2 \qquad (3.7)$$

Substituting equation 3.7 in equation 3.4 gives :

$$E = (f.L) / (A_o.(1 - (v.(e/L)))^2.e)$$

Substituting equation 3.2 in the above equation :

$$E = (f.\frac{100}{T}) / (A_o.(1 - (v.(T/100)))^2)$$

Rearranging to give f :

$$f = E.A_o.(1 - (v.(T/100)))^2\frac{.T}{100} \qquad (3.8)$$

This gives a relationship between the axial force, f, applied to the elastic, and the percentage stretch, T, required to give a correctly-dimensioned workpiece after attacment of the elastic. E, A_o and v are constants, therefore given values for them, f can be calculated for any desired T.

Typical values for Poisson's ratio, v, are between 0.27 and 0.41 for rubber tape. Values for Young's modudlus, E, range from $2.7 * 10^6$ N/m^2 to $7 * 10^6$ N/m^2 for rubber tape [3].

The axial force f, calculated from equation 3.1, is then used as a set point for the feedback control of the elastic tension, being compared to the actual axial force in the elastic as sensed by a the force transducer. An error signal is produced and used to drive the stepper motor forwards or backwards to compensate for changes in the axial force, and hence in the percentage stretch, from the set point.

The correct operation of this algorithm is independent of the distance between the nip rollers and the sewing head, and of the stitches per inch setting of the sewing machine. The device can therefore be easily moved from one machine to another, with a minimum of set-up required. Any tension imparted to the elastic being fed into the nip rollers causes no increase in tension in the elastic fed to the sewing machine, since the absolute tension is controlled between the nip rollers and the sewing head.

The algorithm does, however, require information to be given to it each time a new type of elastic is used. Information on E, A_0, and v is provided by the manufacturer of the elastic, since these are factors in deciding which type of elastic is to be used for a certain garment. The information can, therefore, be easily input by the operator each time a new elastic is used.

3.2 The 6800 development system

3.2.1 Hardware description

The closed-loop system architecture is shown in fig. 9. It is based around a Motorola M6800 development system [2], which was chosen so that useful software already developed for the open-loop control program could be re-used. Program development was carried out on the same equipment as for the open-loop system. The object code produced was downloaded into the 4k * 8-bits of RAM provided in the development system using the 'load'

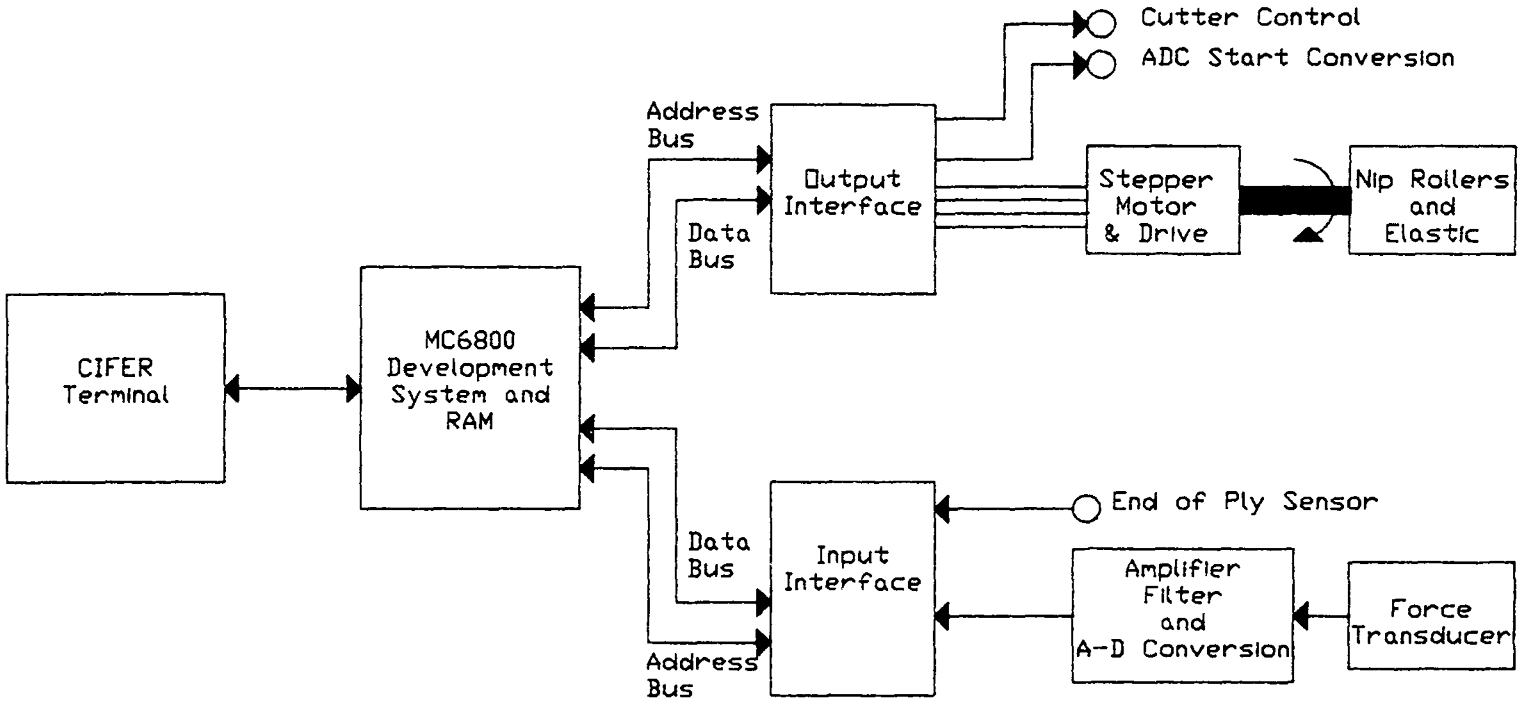

Figure 9. Closed-loop control system architecture.

command in the MIKBUG [2] monitor program resident in ROM in the development system. The program can then be executed by the development system. The Cifer terminal behaves as an operator interface, allowing data to be input to, and messages to be displayed from the development system under program control. No interrupts are used.

The 6800 development system includes two 6821 PIA's, the same as those used in the open loop system, for the input/output interfacing, and an MC6850 asynchronous communications interface adapter (ACIA) [2] for serial communications with the Cifer terminal.

A stepper motor drive circuit was required to drive the motor attached to the nip rollers. In order to improve the nip-roller performance over the open-loop system, a 200-steps per rev motor was used in conjunction with a chopper drive circuit developed previously at Durham. This combination produced much higher holding torque and much improved pull-in and pull-out rates. The motor could easily be run at 700 steps/rev, which corresponded to the maximum rate achievable by the open loop system.

3.2.2 Force transducer development

As was mentioned in section 3.1, some form of transducer was required to to produce an electrical signal proportional to the axial force in the elastic. The requirements for the transducer were that it should be simple and accurate, and fairly compact so that it would take up as little space as possible around the sewing machine.

One of the cheapest and simplest devices for producing an electrical signal proportional to a force is a strain guage [4]. Deformation of a strain guage in either tension or compression causes an increase or decrease in its resistance. If the strain guage is included in a resistance bridge with a reference voltage, V_{ref}, applied to it as in fig. 10, changes in its resistance cause changes in V_O, which is the sum of V1 and V2, proprtional to the force causing deformation of the guage. To produce the largest change in V_O for a given force, and therefore the most sensitive bridge, all four arms of the bridge are replaced by strain guages, forming a full bridge. To give the maximum sensitivity, arms A and C are mounted on one side of a bar, and B and D on the other. If arms A and C are in tension, then arms B and D must be in compression. Thus A and C's resistances increase, and B and D's decrease. V1 then decreases, and V2 increases, giving a change in V_O of (V2 - V1). The full bridge arrangement gives a fourfold increase of the changes in V_O over using a single strain guage and three fixed value close tolerance resistors in a quarter bridge.

However, for the strain guages to be deformed, they must be attached to some object on which the force to be measured is acting. The first attempt to satisfy this criterion, a thin bar with one curved face and one flat face was attached to the frame holding the nip rollers, so that the elastic had to pass over the curved face immediately after being fed through the rollers, shown in fig. 11. The tension in the elastic would therefore cause the bar to deflect. The strain guage bridge was attached to the bar, with two arms on the flat surface, and two arms on a

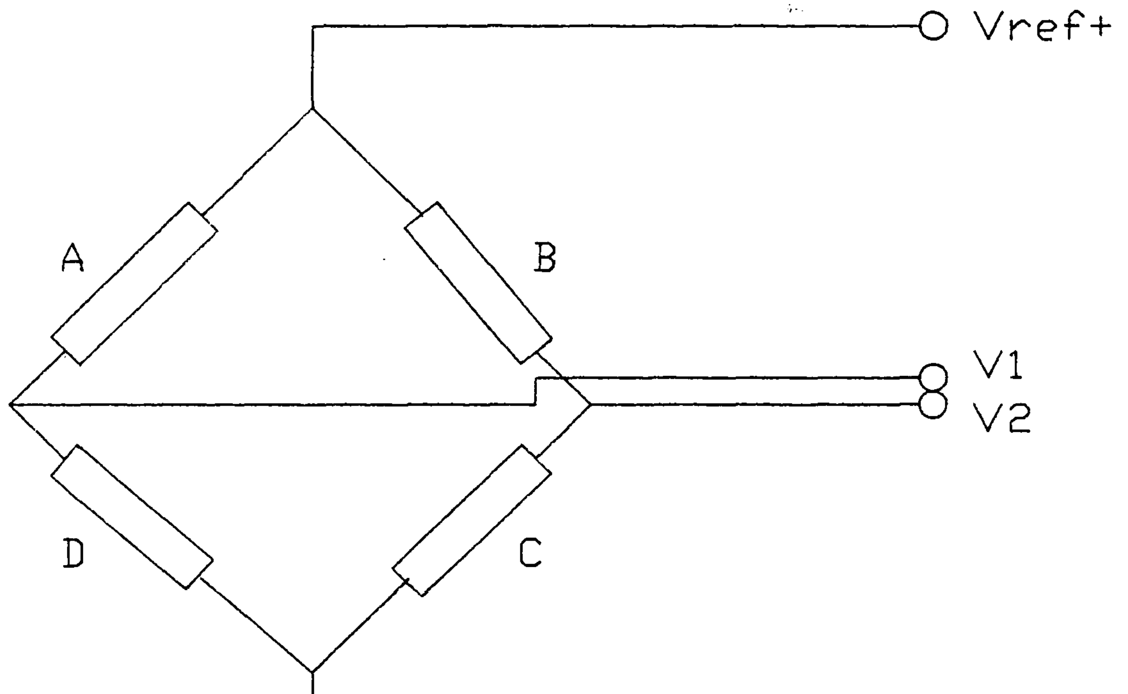

Figure 10. Full bridge strain guage arrangement.

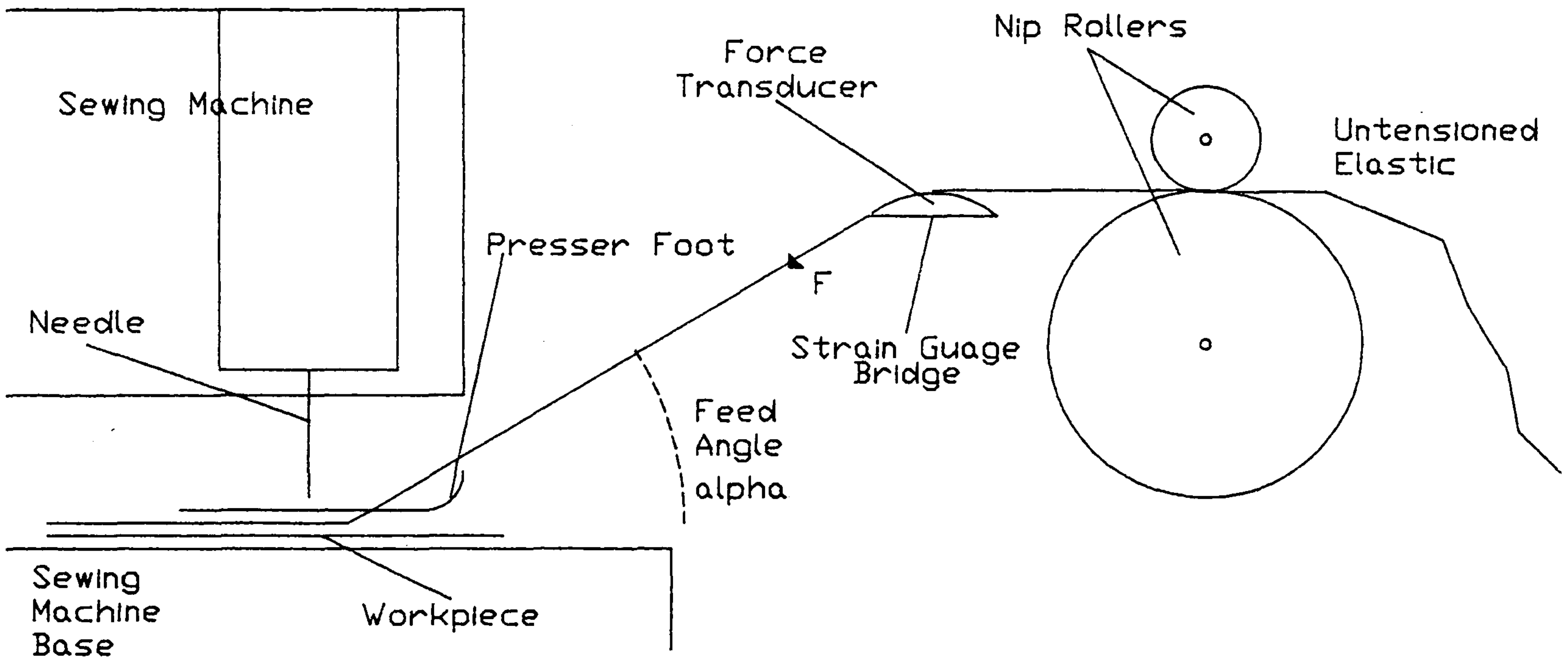

Figure 11. Single bar force transducer.

flattened section of the curved surface.

For the output voltage V_o to represent the axial force in the elastic, it was necessary to know the angle at which the elastic was being fed in to the sewing head, alpha, shown in fig. 11. The force sensed by the transducer is related to the axial force in the elastic by the expression:

$$F_{axial} = F_m / \sin(alpha)$$

For alpha = 30 degrees, this gives a force on the bar of 0.5 * F_{axial}.

The transducer was calibrated by hanging a series of accurate weights from it, and the ADC outputs obtained were used to create a look-up table of forces relating to the ADC outputs. The above equation was used in the production of the look-up table which thus contained the axial forces in the elastic relating to the transducer output as represented by the ADC output. For the 6800 system, this was a much more efficient method of obtaining the axial force than calculation, since operations such as multiplication and division are very slow, and would therefore slow the response of the system to changes in tension. The precise relationship between the applied force and the ADC value read by the control algorithm depends on the strain guages used. Any small variations in resistance between one transducer system and another, and/or between the gain of the amplification and filtering stages, and the calibration curve will be slightly different. Thus each transducer would have to be calibrated individually.

This transducer was used for initial testing of the control algorithm, and worked well if the elastic was only being tensioned or de-tensioned. If, however, the elastic was being fed to a sewing machine, problems arose with friction between the elastic and the bar causing jerky and innaccurate feed, and a lack of sensitivity in the transducer, due to the force acting on it being smaller than the axial force in the elastic which was being controlled.

To overcome these problems, a new transducer was designed, shown in fig. 12. The transducer consists of three bars with rectangular cross-section rigidly mounted at one end forming cantilevers. On the free end of each bar is mounted a roller, which rotates on small bearings. The triangular positioning of the three bars is such that the axial force in the elastic is resolved vertically on to the central bar, to which is attached a full bridge strain guage arrangement. Thus the force on the central bar is independent of the angle of exit of the elastic from the transducer as long as the elastic passes over the exit roller. The most convenient mounting position for the transducer and nip rollers can then be used, since physical positioning relative to the sewing head is not critical. The deflection of the bar resulting from the force exerted by the elastic tape passing over the rollers causes an output to be generated by the strain guage bridge. The equation relating the force on the bar to the axial force in the elastic is:

$$F_{axial} = F_m / (2\cos(alpha))$$

For alpha = 30 degrees, this gives a force on the bar which is 1.732 * F_{axial}. This, combined with the greater deflection of the central bar for a given force due to it being a cantilever and therefore not fixed at both ends, means the transducer

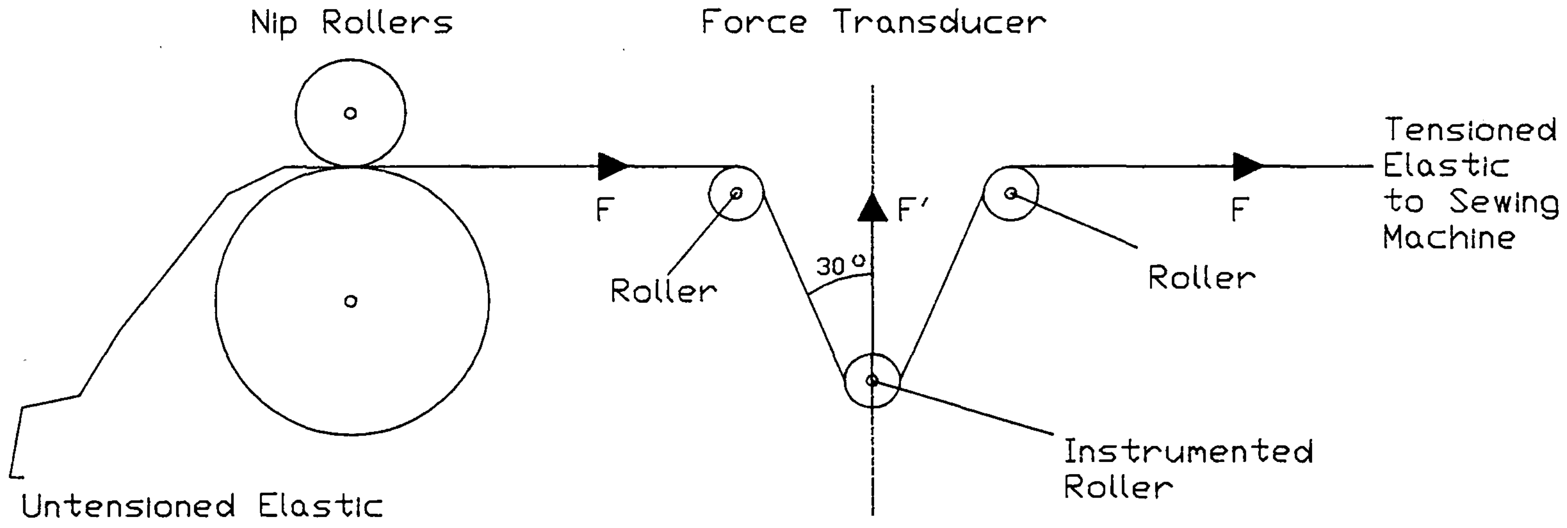

Figure 12. Three bar force transducer.

produces a much greater output than the first design. Consequently, to use the full input range of the ADC, the gain of the amplification stage of the transducer interface was reduced, improving noise performance. This is the design which has been used for all subsequent work.

Calibration of this transducer was carried out in the same way as for the first design, and a typical calibration curve is shown in fig. 13. It can be seen that for the small forces required to stretch the elastic, the response of the transducer is almost linear.

3.2.3 Transducer interface

An interface was required between the transducer and the development system to present the output in a form whuch could be accessed read by the control program. It consisted of three stages : amplification, filtering and A-to-D conversion . Fig. 24 shows the amplification and filtering stages of the interface.

To provide a full voltage range input for the ADC, the gain of the first stage amplification had to be > 800, which is high for a normal 741 type operational amplifier and could cause instability problems. The gain needed to be high because the output from the transducer is in the order of millivolts. To achieve the necessary gain and minimise these problems, a 7652 chopper-stabilised operational amplifier was chosen. This is an instrumentation amplifier, and features extremely low offset voltage, very low time and temperature drift, very high gain and CMRR, low intermodulation effects, and a wide bandwidth. The low offset voltage is achieved by comparing the inverting and

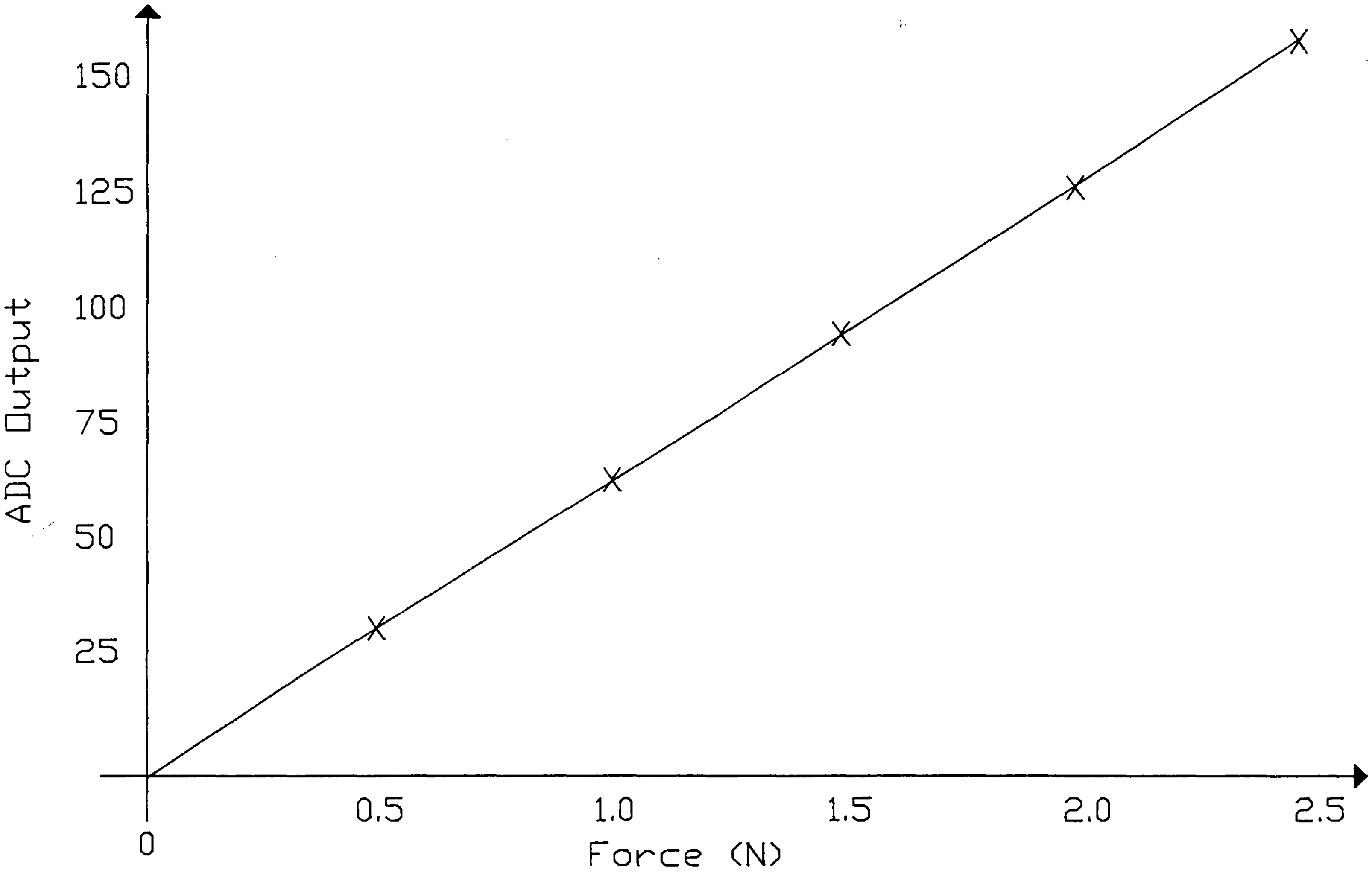

Figure 13. A typical calibration curve for the force transducer

non-inverting input voltages in an on-board nulling amplifier which spends alternate clock phases nulling itself and the main amplifier. Two external capacitors store the correcting potentials on the two amplifier nulling inputs which are then used to reduce the offset voltage of the whole amplifier. The amplifier was configured in differential mode [4].

The second stage consisted of a second-order low-pass Butterworth filter [4]. This type of filter was chosen because it has a very flat passband response, with a sharp transition from passband to stopband at the cut-off frequency. The time constant for the filter was critical, as the filter had to have a fast response to be able to cope with the small changes in the elastic tension, as it is pulled in to the sewing head, without lagging the input, thus causing instability and even oscillation. The time constant for the filter is given by

$$T_C = R.C$$

and the cutoff frequency (-3db point in the response) is given by

$$f_C = 1 \; / \; (2.\text{Đ}.R.C)$$

Initially, a cut-off frequency of 5Hz was chosen, but the slow response of the system with a time constant of 31.8ms caused serious oscillation in the control loop in response to almost any impulse or attempt to feed the elastic. A value of 3.3ms was then chosen, and this gave a very stable steady-state response to impulses applied to the transducer bar. The time constant consideration was traded off with the cut-off frequency, which for this particular value was 48.2Hz.

An A-to-D converter (ADC) was needed to convert the output from the transducer providing the feedback of the axial force in the elastic to a digital signal which could be read by the control program running on the 6800 system. Since the 6800 is an 8-bit processor, a ZN427E 8-bit ADC was chosen. It was configured to operate in unipolar mode with an input range of 0v to +5v using the internal 2.5v reference voltage. Control of the A-to-D conversions was achieved by using one bit of Port A of PIA 1 on the 6800 system to start an ADC cycle, and another bit to monitor the conversion complete output of the ADC. The ADC was clocked at 25KHz, giving a conversion cycle time of 0.44 ms.

Also included in the interface was a refrence voltage for the strain guage bridge, V_{ref}. Two ICL8069 1.2v voltage reference IC's were used, giving 2.4v across the bridge. With each arm of the bridge having a resistance of 120ohms, the output voltage change for a 10% change in the strain guage bridge resistance would be 2mv, and the output from the amplification and filtering stages would be 1.67v.

3.3 The program

A flow chart for the closed-loop tension control program is shown in fig. 14. The first stage of the program execution initialises the PIA ports to the appropriate inputs and outputs. The operator is prompted for information on the type of elastic to be used. The default value is the previously used data. Once this has been input, the control algorithm computes the axial force required in the elastic to produce the desired percentage stretch. A look-up table, obtained from the calibration of the transducer, is then accessed to obtain the transducer ADC reading which relates to this force.

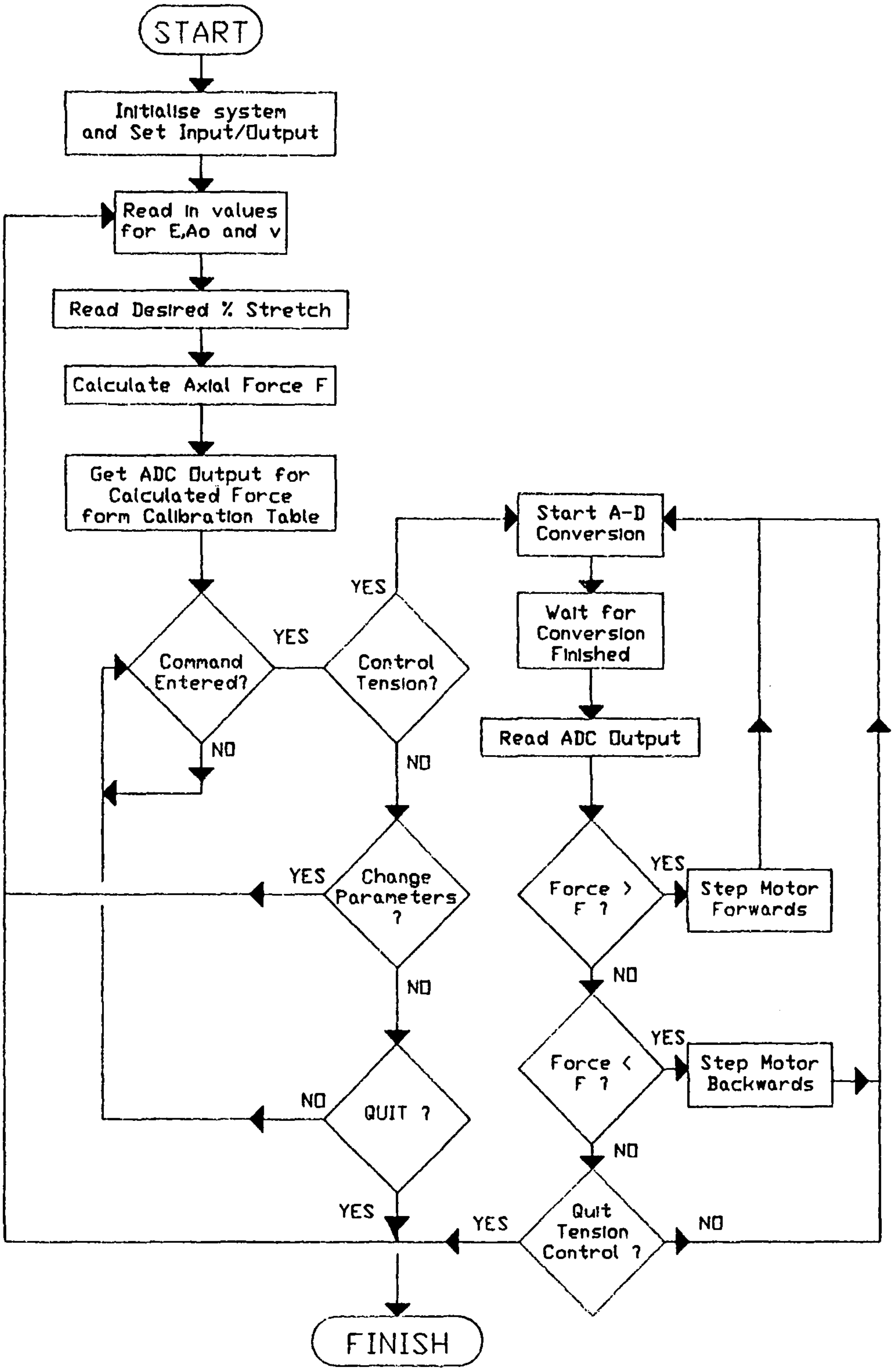

Figure 14. Closed-loop control program flow chart.

A menu is displayed showing the operator the options available. The program the allows the operator to change any of the parameters E, A_0, v or desired percentage stretch, to pre-tension or de-tension the elastic, or to start continous control the elastic feed.

Pre-tensioning the elastic simply steps the motor in such a way as to stretch the elastic until the axial force read in by the control algorithm, as indicated by the ADC output, is equal to the set-point calculated earlier. De-tensioning steps the motor in such a way as to release the tension in the elastic until the axial force is zero. When either of these tasks is completed, the program returns to the option menu. The feed control algorithm maintains the the tension in the elastic by stepping the motor backwards or forwards as necessary as the elastic is pulled into the sewing head by the feed dogs and the presser foot.

There are two parameters which have to be controlled to achieve the correct size for a given workpiece. One is the percentage stretch in the elastic, and the other is the rate of feed of the workpiece to the sewing machine. The desired stretch in the elastic is achieved by the automated control of the elastic tension. The rate of feed of the workpiece usually involves a human operator, thus requiring a man-machine interface to close the loop. An additional feature was included in the system to provide the operator with information on the amount of untensioned elastic attached to each garment. Since the rate of workpiece feed is directly proportional to the amount of untensioned elastic, the operator can use the displayed information to adjust his or her rate of feed to produce the correct garment size. This was achieved in the following manner.

While the program is in the elastic feed control mode, the status of the beginning/end of ply sensor is monitored. The number of forward and reverse motor steps during attachment of elastic to the ply is counted and from that, the amount of untensioned elastic which has been fed into the ply can be calculated. This can then be compared to the desired amount of elastic which should have been fed in to the particular size of garment being produced. An indication is then given to the operator of whether the correct size of garment has been produced within pre-determined tolerances. Any appropriate action can then be taken to re-process or discard the incorrectly-sized garment.

3.4 System performance

The closed-loop system was first tested in the same manner as the open-loop system, with a second independently-driven pair of nip-rollers simulating a sewing machine. The results of these tests for two different types of rubber tape are shown in tables 2 and 4. For the results in table 2, a grey-coloured rubber tape was used with $v = 0.385$, and $E * A_o = 6.24$. For the results in table 4, a white-coloured rubber tape was used, with $v = 0.400$, $E * A_o = 7.86$. For both types of elastic, the percentage stretch was maintained to ± 1% for virtually all settings. The variation in the 30% setting for the grey tape was due to a length which had a slightly greater cross-sectional area than the rest of the tape. This type of variation in the manufacture of elastic tape is common, but the 3% error involved is not large enough for a garment to be incorrectly sized. A decision based on these encouraging results was then made to try the system on a sewing machine.

An industrial overlock sewing machine was obtained on loan from the Courtaulds factory in Gateshead. The tape used was grey rubber tape. It was attached to knitted cotton material such as that found in T-shirts and underwear. The range of reduction in size required for these garments varies between about 5 and 20%, thus requiring tape with a similar stretch to be attached to the material to provide the necessary reduction.

Once information on the type of elastic being used, such as E, v and A_0, had been given to the system, it was found that the elastic stretch could be controlled to within 1% as it was being fed into the sewing machine. Problems arose, however, when the garment produced was measured to check that the correct size had been produced. It was found that with the elastic being fed in to the top of the presser foot from above through a slot, the standard feed method currently in use, the size was much smaller than anticipated, implying that the elastic was over-tensioned when it was attached to the material. The stretch in the elastic being fed to the sewing head was checked first, and was found to be at the correct tension. The problem therefore seemed to be caused by the movement of the elastic into and under the presser foot imparting extra tension to the elastic.

To find whether the feed through the slot was causing the problem, the nip rollers were positioned so that the elastic wase fed directly in from the front of the presser foot parallel to the machine plate. The sizes produced form this arrangement were found to be correct, thus confirming that the problem was with the feed in through the presser foot slot. Friction effects between the elastic and the lip of the slot, and the action of the feed dogs on small sections of the elastic as it is pulled under the needle to be stitched were the cause. These results were also found when measurements were taken in an industrial enviroment on the open-loop system, as described in section 2. i.e. that the tension in the elastic being fed in to the sewing head was less than the value calculated for the size required, to compensate for this effect.

Two approaches were tried to solve this problem. Firstly, feeding from the front of the presser foot was investigated. Although this worked well on occasions, because there was no guide to keep the elastic lined up with the presser foot, it tended to drift from the desired line, and foul the needle. Secondly, a roller was attached to the front of the presser foot, and the elastic fed from above around the roller and in to the front of the foot. The effect was reduced, but the roller obstructed the operator when feeding the beginning and end of the ply through the sewing head.

From these results, it became clear that a new approach would have to be taken to compensate for these effects.

3.5 The adaptive control approach

The new approach involved a software/system approach which would not require any alterations to existing presser foot designs.

Measurement of the untensioned elastic being fed in to the sewing head using the opto-switch were found to be accurate to better than 1mm, giving a precision in readings of normal garment sizes of 300mm of between 0.25 and 0.50%. An alternative solution was achieved based on this measurement.

When starting to process a new batch of workpieces, the operator is prompted by the system for the original length of the workpieces to be processed, and the desired length after attachment of the pre-tensioned elastic. From these two quantities, the length of untensioned elastic to be fed in to the sewing head is calculated, and hence the required percentage stretch in the pre-tensioned elastic can be obtained. The equation 3.1 is then used to convert the percentage stretch in to an initial set-point for the closed-loop feed control of the elastic. The system then pre-tensions the elastic.

The operator then feeds a workpiece through the sewing machine. The system checks the finished size against that desired by the operator. The size is usually too small, due to friction effects as described above, thus giving an error.

An adaptive control or performance algorithm is then used to calculate a new set-point for the feed control algorithm. A diagram of this arrangement is shown in fig. 15. The set-point is either increased or decreased as necessary based on the error in size being too large or too small. The operator then feeds in a second workpiece and the process is repeated. The workpiece sizes converge on the desired size after between 2 and 4 attachment operations, as shown in figs. 16 and 17. The effect of this approach is to add an outer closed loop to the control

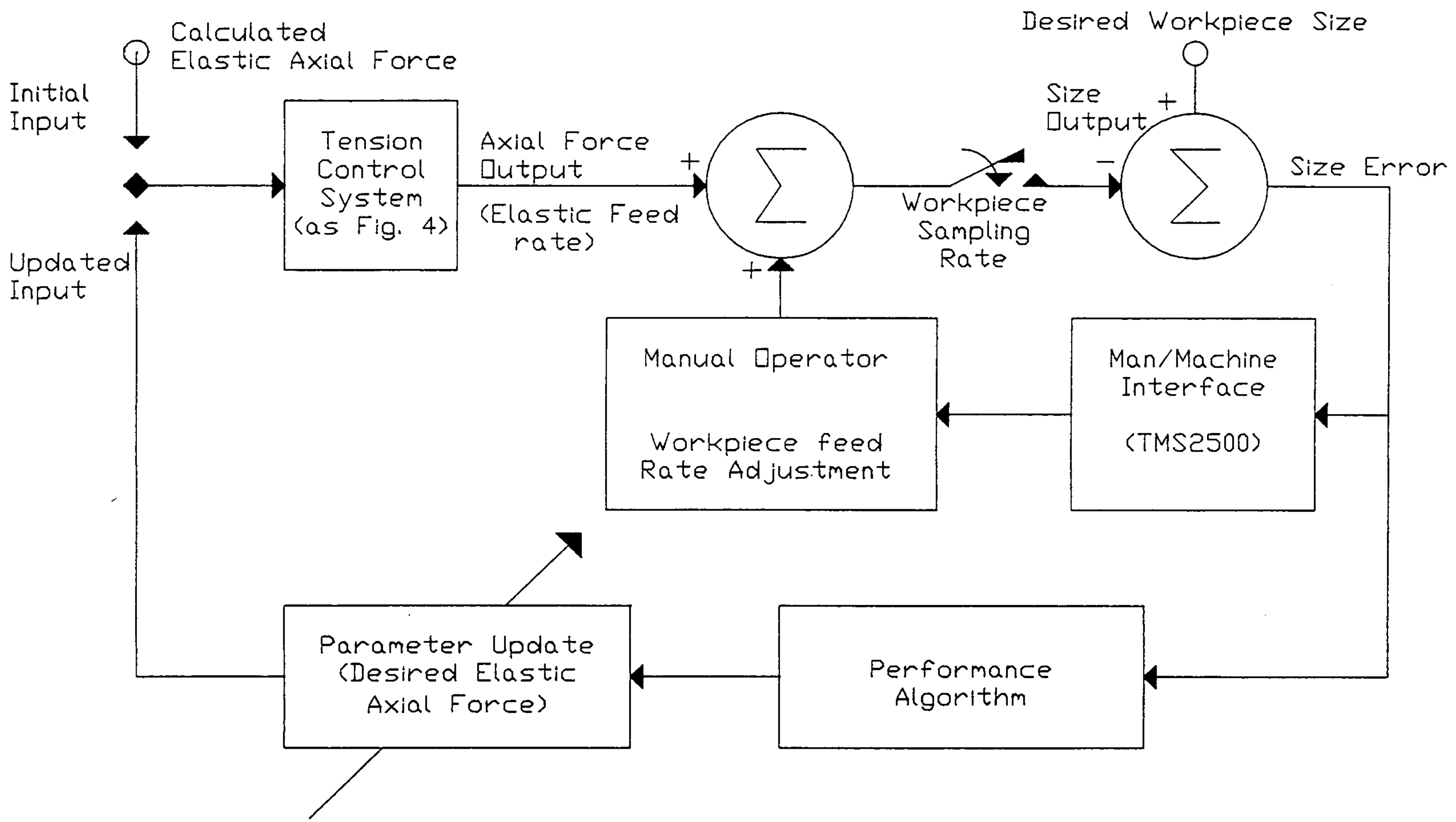

Figure 15. Block diagram of the adaptive control loop

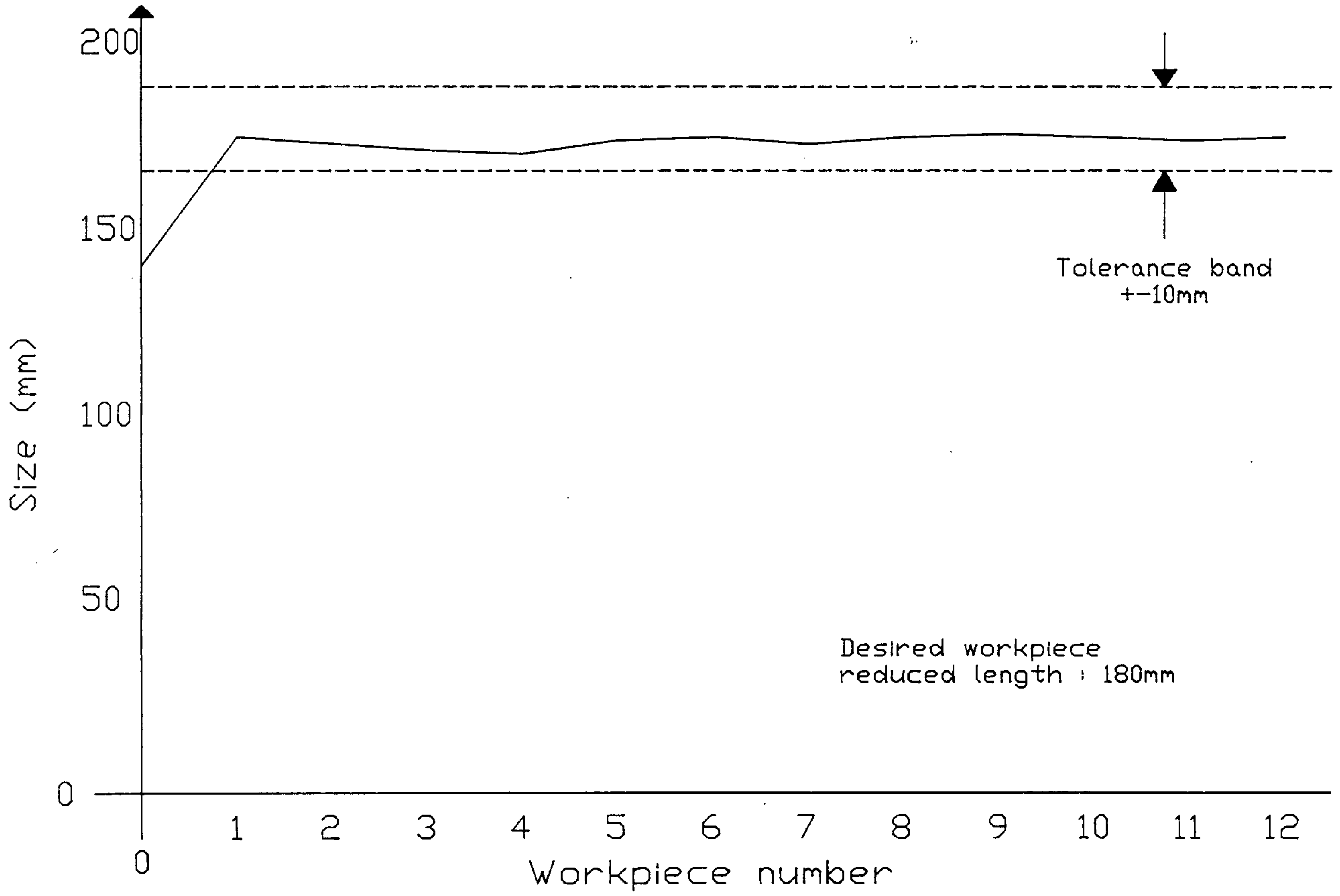

Figure 16. Adaptive control loop response.

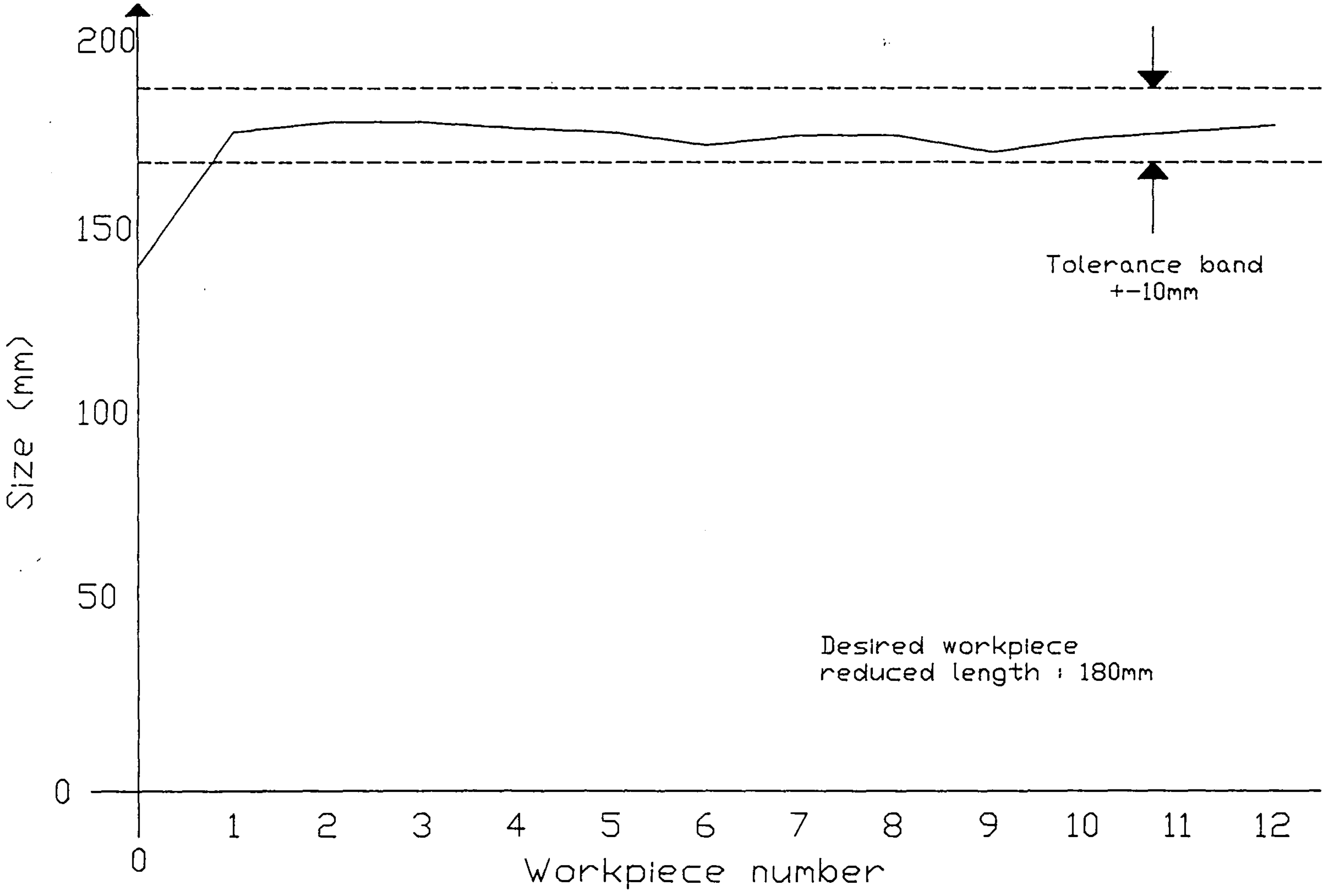

Figure 17. Adaptive control loop response (2).

system. The original closed-loop feed control of the elastic as it is attached to a workpiece is unaltered. However, the set-point for the control loop is now supplied by the adaptive algorithm after each workpiece has been processed, instead of the algorithm described in section 3.1.

Several advantages to this approach over the earlier method can be seen. Firstly, control is independent of the physical properties of the type of elastic being used. Any variations in the elastic properties, such as those noted in section 3.4 are also compensated for since every workpiece is checked for correct sizing, and any necessary changes are made to the elastic tension before the next workpiece is processed. The operator is given an indication of the size of the workpiece just processed in terms of a ± deviation from the desired size. An audible warning is given if the error is greater than the required tolerance. The operator therefore has no need to look at the terminal after every operation. If the warning is heard, the workpiece size can be checked, and the workpiece reprocessed or discarded as desired.

This independence from elastic properties means that the operator only has to enter information to the system in terms of dimensions which are easy for him to relate to, instead of parameters such as Poisson's ratio which mean nothing to him.

A further advantage of this approach is the operator workpiece feed rate. The adaptive algorithm aims to correct the elastic feed rate assuming that there is no added tension in the workpiece. If the operator is imparting tension in to the workpiece as she feeds or guides it through the sewing head, the measured dimensions will be small all the time, since the effect of this extra tension will be added to the effect of the elastic tension. If the sewing machine is correctly set up, it is possible to allow the workpiece to be pulled through the sewing head by the presser foot and feed dogs, with the operator only providing guidance of the workpiece to prevent rucking.

Since the results obtained from this approach were vey encouraging on the sewing machine, it was decided to develop a stand-alone unit which could be taken to the Courtaulds factory and tested in situ. This system is described in the next chapter.

CHAPTER 4

THE STAND-ALONE CONTROL SYSTEM

The next stage of the project was the construction of a stand-alone system which could be tested in an industrial enviroment. It incorporates features other than the basic elastic tension control algorithm, including size checking, a versatile but simple operator interface, and control of a pneumatic cutter attached to the sewing machine to minimise waste of elastic tape. The aim of this unit was not to produce a commercially viable unit, but to allow testing of the control algorithm in an industrial enviroment over a prolonged period, whilst retaining the facility for further development of the system in the light of any problems encountered during testing.

The system uses the force transducer developed in the earlier stages of the project to provide feedback of the elastic tension. All the other electronics were either replaced or upgraded to improve the versatility and robustness of the system and its performance at sewing speeds up to 6000 stitches per second.

4.1 Hardware description

4.1.1 MC68000 development board

In order to facilitate accurate control at higher sewing speeds and to incorporate extra features in the stand-alone system, a more powerful microprocessor architecture than the 6800 family was needed. A Flight-68K single board computer [6], built around a Motorola MC68000 16-bit microprocessor, was chosen to be the heart of the system. Apart form the 68000, the board contains a Motorola MC68230 Peripheral Interface/Timer (PIT) [6] and an MC68681 Dual Asynchronous Receiver/Transmitter (DUART) [6].

4.1.2 MC68230 peripheral interface/timer

The 68230 PIT provides all the necessary digital input/output for the system. Three 8-bit ports A, B, and C are provided. All three ports are used in the bit input/output mode.

The 4 least significant bits of port A, PA0-3, monitor and control the ADC. PA1 is an input, PA0,2 and 3 are outputs. PA0 provides the Start Conversion (SC) signal. PA1 monitors the End Of Conversion (EOC) output of the ADC. PA2 and 3 control what data is presented on the output pins of the ADC.

The stepper motor drive is controlled by PA4 and 5, which are connected to the motor direction and clock inputs of the drive respectively.

The end of ply sensor output is connected to PA7.

All 8 bits of port B are inputs, and are connected to the outputs of the ADC.

The least significant bit of port C is an output for the cutter control signal.

The timer/counter is used to generate a square wave which is then used as the clock signal for the ADC. The output of the timer/counter appears on pin PC3/TOUT, which is bit 4 of port c.

4.1.3 MC68681 dual asynchronous receiver/transmitter

The 68681 DUART provides 2 asynchronous serial communications channels, ports A and B. The TTL inputs and outputs of the DUART are buffered by a 1488 and 1489 line driver and receiver. A DC-DC converter on the board produces a split 12v DC supply for the inputs and outputs of the 1488 and 1489, to give RS232 voltage levels on the serial lines.

One channel allows communications with a terminal, in this ase an Ampex A210 terminal emulating a TVI920, and the other allows communications with a host computer or some other device, in this case a TM2500 data terminal [8] and the CMC Vitesse host computer.

Port A was used solely for communications with the A210, allowing diagnostic messages to be displayed using system calls provided by the monitor program from a user program running on the 68000 board. The data format and Baud rate for port A communications is determined by the monitor program when the system is booted up. The port is initilaised for a transmission format of 8 data bits, 1 stop bit, and no parity, and the A210 was set up to accomodate this. A series of carriage returns are then sent from the terminal, and the 68000 board synchronises its Baud rate from these to that of the terminal. The Baud rate used was 9600, since the communications line to the host computers operates at that speed.

Port B was used for two purposes : communications with a host computer, and with the TM2500 data terminal. Slight alterations had to be made to the link area for port B on the

68000 board itself in order to accomodate the different hardware handshaking connections used by the host, so that the cable to the host would have standard termination, thus aiding any fault-finding necessary. The 68681 uses DTR (input) and CTS (output) but the host uses DSR (output) and CTS (input), so the link area had to be changed to connect host DSR to 68681 DTR and host CTS to host DTR. All other connections were unchanged. The data format was the same as for port A.

For use with the data terminal, a custom lead had to be constructed to give the required connections, since the data terminal has a 9-pin D-connector for RS232C serial communications. The terminal requires CTS (input) and DTR (output).

4.2 Signal Conditioning

Conditioning of the full bridge strain guage output from the force transducer was again provided in two stages. A 7652 chopper-stabilised operational amplifier again provided the high gain first amplification stage. However, the second stage amplification and filtering was altered to reduce the time constant of the filter to 1.5ms, to accomodate the faster cycle time for sampling with the 68000 processor.

The output from the second conditioning stage was then fed to a ZN502E 10-bit ADC, operating in bipolar mode with an input voltage range of -5v to +5v. Bipolar mode was chosen to allow automatic zeroing of the transducer output to compensate for any drift in the transducer output. The accuracy of the conversion tage was thus incresed by a factor of 2 over the 6800 system since the ADC had four times the resolution of the 8-bit ZN427E, but the operating range was increased to 10v.

A retro-reflective opto-switch was attached to the sewing machine at the rear of the presser foot to sense the beginning and end of workpieces as they are fed through the sewing machine. The output of the opto-switch is fed to one input of an LM393 dual comparator, and a refernece signal to the other input. The reference voltage can thus be used to help adjust the sensitivity of the opto-switch circuit for correct operation. The output of the comparator is filtered using a simple R-C network to avoid oscillation of the output when the input voltages are similar. The filtered sgnal is then used by the control algorithm in the calculation of the amount of untensioned elastic fed to the workpiece as it is processed.

4.3 Stepper motor drive

The stepper motor drive used in the stand-alone system was a commercially available Parker-Digiplan PK 2. It is a chopper current control drive, able to drive a 2.2" outer diameter 200 step/rev motor at speeds up to 1000 steps/sec without stalling with no acceleration. With a nip roller diameter of 1.2 inches, this means the system can feed elastic at 0% stretch to a sewing machine set to 14 stitches/inch at machine speeds up to 6000 stitches/min. Two control signals are needed to control the PK 2 drive. The state of the direction input determines whether the four motor phase outputs are energised to step the motor clockwise or anticlockwise, and the clock input determines the number of steps in the desired direction, one cycle on the clock input producing one step of the motor. The clock and direction inputs to the PK2 use 12v logic levels, so the outpus from the 68230 PIT are optically isolated using a dual opto-isolator.

4.4 Data Terminal

One of the main requirements for the stand-alone system was the inclusion of an operator interface which would allow messages to be displayed to the operator by the system,and for the operator to be able to alter easily the parameters controlled by the system.

A Burr-Brown TMS 2500 Microterminal [8] was chosen to perform this task. The front panel is shown in fig. 18. It effectively replaced the Cifer used in the 6800 system. It is very robust, an essential quality in a industrial enviroment, and the keyboard is sealed, protecting it from the large amounts of dust and fibres present in the atmosphere around processes involving textiles.

The terminal features a 16-digit LCD display with variable viewing angle. There are 24 keys, including a numeric keypad and six back-lit function keys. Most features of the terminal are under program control from the 68000 board using escape sequences.

Communications with the terminal are via an RS232C interface, using one of the serial ports provided by the 68681 DUART. The terminal requires a five-wire RS232 interface, using receive, transmit, ground, Clear-To-Send (CTS), and Data-Terminal-Ready (DTR). The communications protocol used is 7-bit, no parity and one stop bit, at a data transfer rate of 9600 baud. The terminal is set up to operate in block mode.

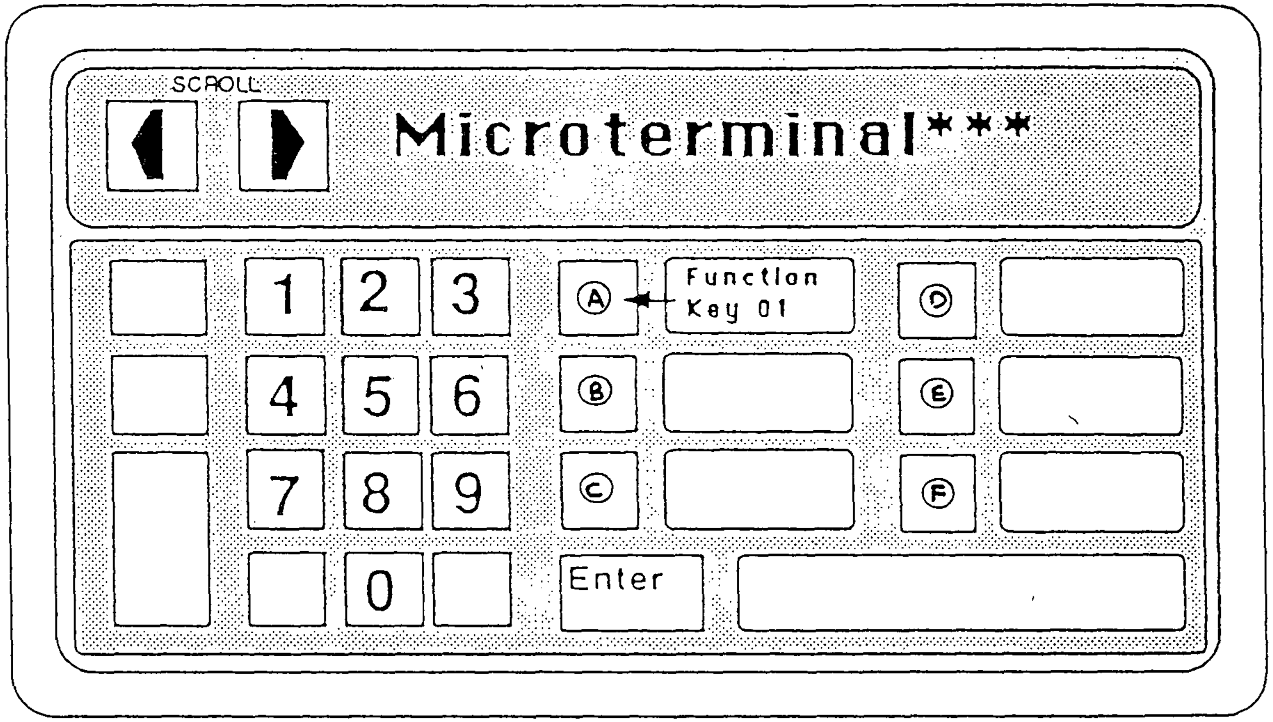

Figure 18. TMS2500 layout.

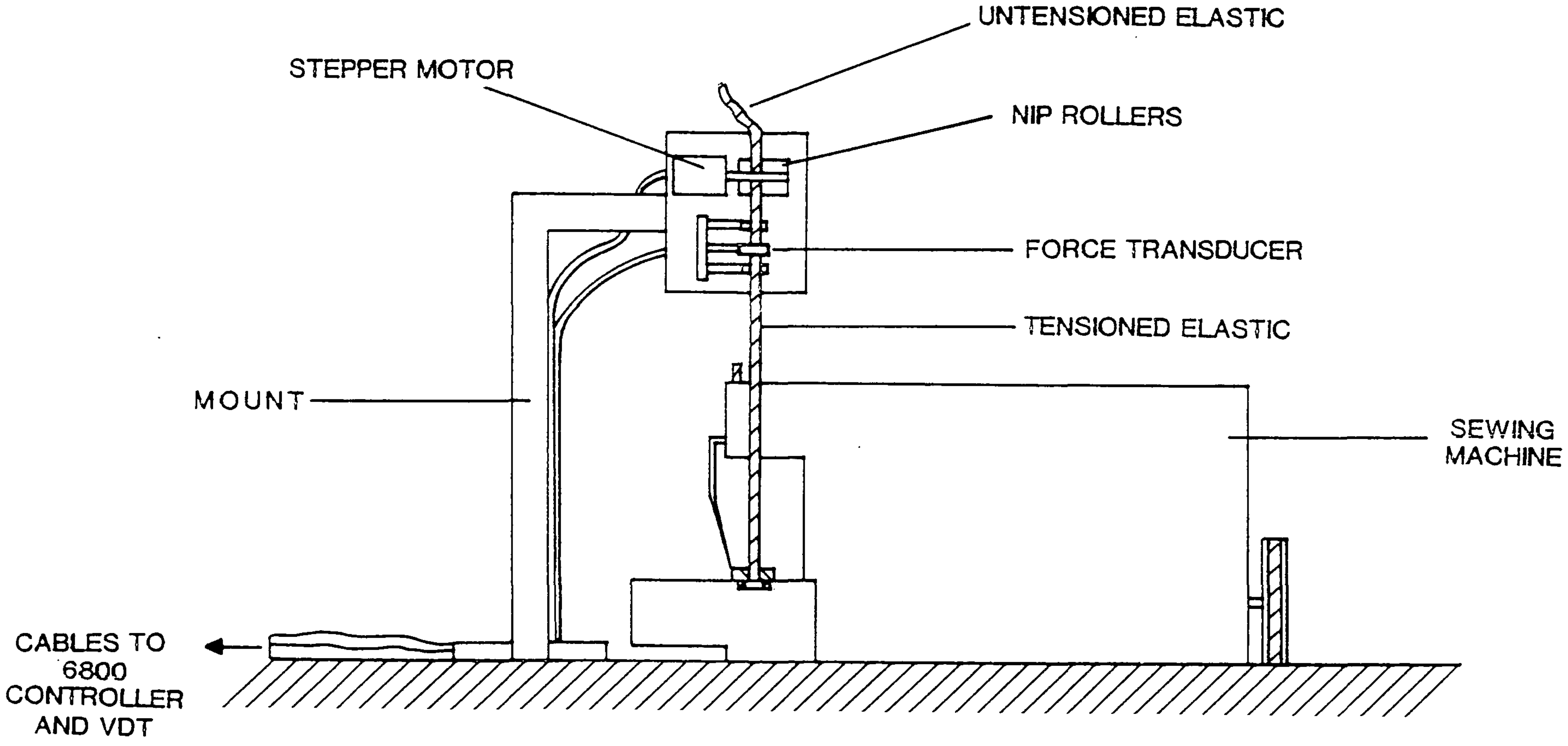

Figure 19. Installation of the closed-loop system.

Thus the editing facilities on the terminal can be used by the operator to change any mistakes made whilst entering data before the data is sent to the 68000 board. The terminal provides a 'delete' key which deletes one character at a time, and a 'clear' key which clears the entire entry without sending any characters to the 68000 board. Each block of data entered must be terminated by pressing the enter key, which sends the hex ASCII control character 03. This character was chosen so that carriage returns could be included in blocks of data sent to the 68000, if required. The control program recognises this character as the termination of the block of data, and can then continue and process the data as required.

The six function keys are programmable from software. A string of up to 4 characters can be assigned to each key, so that when a key is pressed, the string is sent to the host. The keys were assigned the ASCII characters 'A' to 'F', followed by the termination character 03. Each time a function key is pressed, therefore, the ASCII character associated with that key and the termination character are sent to the host, so there is no need to use the 'enter' key when using functions. Local echo is also switched off on the terminal so that the function key string does not appear on the terminal's LCD display.

Once a function key string has been received by the program, an escape sequence is sent to the terminal which switches on the backlight for the function key that has been pressed. Once the task initiated by the key has been completed or cancelled the backlight is switched off again. This gives the operator an instant visual indication of what task is being performed, backing up the messages displayed by the terminal.

Next to each function key is a space where a label can be placed to explain the operation of the corresponding key.

There are other features of the terminal which are under program control, such as key click, key repeat, keyboard lock, and a bell. Key click sounds the bell whenever a key is pressed to give an audible confirmation of correct operation. Key repeat operates if a key is held down for more than a second. Keyboard lock prevents any key from functioning until the keyboard is unlocked again. The bell can be sounded to give an audible indication of an event.

4.5 Software description

The monitor program for the Flight-68K system resides in 32k of EPROM and is supplied with the board [6]. User program memory is provided by 16k of RAM, which can be expanded to 128K. For operation as a stand-alone unit in a factory, the monitor program EPROMs were removed and replaced by EEPROMs containing the control program. Thus on power-up or reset, the board boots up into the control program rather than the monitor program.

Program development for the system was carried out on a CMC Vitesse mini-computer, running the Uniplus V2.1 operating system. A Durham Microprocessor Centre 68000 cross-assembler was used to produce relocatable machine code, and so, using a command provided in the 68000 monitor program for down-loading code from a host, was easily down-loaded from the Vitesse to the development board's RAM.

4.5.1 The Program

A listing for this control program is in Appendix A. The program firstly initializes the PIT and DUART to operate as described in sections 4.1.2 and 4.1.3. The program then enters command mode.

The operator is prompted via messages displayed on the TM2500's LCD display to enter commands or data via the keypad. There are six options available to the operator, via the six backlit function keys on the TM2500.

Function key 'A' in fig. 18, when selected, displays a series of help messages explaing the operation of the other five function keys. The corresponding function key is backlit while its help message is on the display. The operator simply hits the 'Enter' key to view the next help message. Once the help is completed, the operator is returned to the command mode.

Function key 'B' prompts the operator for the original and reduced lengths of the workpiece to be processed in mm. This data is then used by the adaptive control algorithm to monitor the workpiece sizing whilst in tension control mode (Function 'F').

Function key 'C' allows the operator to set the length in mm of elastic to be left after the end of the workpiece before it is cut by the pneumatic cutter.

Function key 'D' prompts the operator for values of Young's modulus, Poisson's ratio and the unstretched cross-sectional area of the elastic being used. This information is used to calculate the necessary axial force in the elastic for the desired percentage stretch calculated from the original and reduced lengths of the workpiece. The axial force is then used to calculate the set point for the tension control algorithm if no workpieces have been processed, since the necessary tension will not be known until the adaptive control algorithm has changed the set point to give the correct size. If the information on the elastic properties has not been entered, a default set point is used, which still normally allows the adaptive algorithm to achieve correct sizing within 2 or 3 workpieces.

Function key 'E' toggles between pretensioning and detensioning the elastic, starting with pretensioning. The program returns immediately to the command prompt upon completion.

Function key 'F' initiates feed control. Elastic tension is maintained by the tension control loop shown in fig. 4, and the correct sizing of workpieces is maintained by the adaptive control loop shown in fig. 15. To exit this function, the operator simply needs to hit the 'Enter' key on the TM2500, and is then returned to the command mode.

During the execution of any function, if the operator enters an illegal or out-of-range value, an error message is displayed indicating the error, and the 'bell' is sounded on the terminal.

As can be seen, operation of the system is very easy. In its simplest mode of operation, the operator would enter the original and reduced lengths of the workpiece to be processed using function key 'B', and then enter tension control mode ussing function key 'F' and begin working.

The closed-loop system results discussed in section 3.5 were in fact obtained from this system.

CHAPTER 5

SYSTEM SIMULATIONS AND COMPARISONS

Two computer programs were written on the Durham MTS system in PASCAL to simulate the performance of the tension control algorithms of the open and closed-loop systems when feeding elastic into a sewing machine under a variety of conditions. This was done to allow plots of the system responses under these conditions to be obtained and compared much more easily than attempting to measure the small dynamic changes in tension of the elastic in a real feed operation.

The programs contain common procedures written to simulate the action of the sewing machine as it pulls elastic into the sewing head, the changes in elastic tension following the execution of motor steps, and the effect of electrical noise and slippage of elastic through the nip rollers. In both cases the nip rollers are assumed to be the same distance from the sewing head, 203.2mm, of the same dimensions, and to have the same stepper motor and drive. The sewing machine is assumed to be operating at 10.0 stitches per inch, and at a constant speed of 5200 rpm (86.7 stitches per second), which is the maximum speed for which the open loop system was designed to operate, and is unlikely to be achived during normal operation. Appendix A contains a listing of the closed-loop simulation program.

Both systems are of the digital non-linear type. The main problem in the analysis came form the non-linear behaviour of the elastic fed through the nip rollers. When stepping forwards to feed elastic to the sewing head, untensioned elastic is fed through the rollers, giving the same length feed for each motor step. However, when stepping backwards to tension the elastic, each step pulls tensioned elastic through the rollers, and the tension is increased with each step. Thus the length of elastic pulled through depends on the tension already in the elastic when a step is executed, and changes with each one. This problem was overcome by having different algorithms to calculate the elastic tension for motor steps in each direction. Procedures were written to simulate the action of the two control algorithms in response to the operation of the sewing machine. For the closed-loop system, procedures to simulate the filter response and ADC quantisation were also written.

5.1 Closed-loop control system simulation.

To simulate the closed-loop system, it was first necessary to identify the loop elements involved. Fig. 4 shows a block diagram of the tension control loop. The set-point force, f_s, is the axial force necessary in the elastic to produce the required percentage stretch, and is calculated from equation 3.1. To allow the control program to compare this force to the ADC output, f_s has to be quantised to give the nearest integer value, f_{sq}, less than f_s. The output, U_n, of the controller after n sampling intervals is simply a single motor step forwards or backwards depending on the error, E_n, between the axial force in the elastic as indicated by the ADC output,

f_{nadc}, and the set-point force, f_{sq}. Thus

$$U_n = E_n \,/\, |E_n| \qquad (5.1)$$

where

$$E_n = f_{sq} - f_{nadc}$$

The change in position of the stepper motor and hence the nip rollers alters the actual tension, f_n, in the elastic between the nip rollers and the sewing head by an amount df_n. The current value of f_n is thus given by

$$f_n = f_{nm1} + U_n.df_n \qquad (5.2)$$

where f_{nm1} is the force in the elastic at the previous sample instant. To evaluate this expression, df_n is needed. From equation 3.1 again,

$$df_n = E.A_o.((e/L)_n.(1 - (e/L)_n.v)^2 - \\ (e/L)_{nm1}.(1 - e/L)_{nm1}.v)^2) \qquad (5.3)$$

where $(e/L)_n$ is the current strain, and $(e/L)_{nm1}$ is the previous strain, and E, A_o and v are as described in Section 3.1.

However, to evaluate the strain $(e/L)_n$, we need to know L_n, the new length of untensioned elastic stretched to L_o (203.2mm) beteen the nip rollers and the sewing head. If the elastic tension is too low, the nip rollers are stepped in such a way as to pull some of the elastic between the nip rollers and the sewing head out, thus reducing the amount of elastic between the nip rollers and sewing head and increasing the tension. In this case, L_n is given by

$$L_n = (L_o \,/\, (L_{nm1} - (1/(1 + (e/L)_{nm1}).dX))) - 1 \qquad (5.4)$$

where dX is the angular displacement of the nip rollers caused by one step of the stepper motor. If the elastic is being de-tensioned, i.e. untensioned elastic is being fed in through the nip rollers to the sewing head, the new length L_n is gibven by

$$L_n = L_{nm1} + dX \qquad (5.5)$$

From equations 5.4 and 5.5, the value of $(e/L)_n$ can be calculated as follows

$$(e/l)_n = (L_o \,/\, L_n) - 1 \qquad (5.6)$$

Once this has been obtained, a value for f_n can be calculated using equations 5.2 and 5.3.

However, for the closed-loop system, the output of the transducer, related to f_n via the calibration curve given in fig. 13, has to be amplified, filtered and converted to a digital value that can be read by the control program. The amplification and filtering is shown in fig. 24. The amplification stage of the interface board is assumed to only apply a gain to the output of the transducer. This gain, and any gain added by the filtering stage, are accounted for in the calibration curve. Since the calibration curve is effectively a straight line, this gain is the gradient of the curve, and the value used for the simulations was 70.313.

The filtering stage causes a phase lag in the output of the filter compared to the input. The Laplace transfer function for a second-order Butterworth filter is :

$$\frac{\dfrac{1}{C_1C_2R_1R_2}}{s_2 + s(1/C_1R_1+1/C_2R_2) + 1/C_1C_2R_1R_2} \qquad (5.7)$$

The numerator is a constant, i.e. a gain factor, and is therefore contained in the calibration curve gradient. So, to simulate the response of the filter, the roots of the denominator need to be found, the denominator being

$$s_2 + s(1/C_1R_1+1/C_2R_2) + 1/C_1C_2R_1R_2 \qquad (5.8)$$

The roots can be calculated using the standard equation for a quadratic [5], which is

$$\text{roots} = \frac{-b \pm \text{sqrt}(b^2 - 4ac)}{2a}$$

where a, b, and c are the coefficients of the quadratic equation. Since, for the filter, $C_1 = 2.C_2$, and $R_1 = R_2$, the values for a, b, and c are

$a = 1$

$b = 3/(2CR)$

$c = 1/(2C_2R_2)$

The values for R and C used for the filter were 1Mohm and 1500pf. Substituting these values in the above equations for a, b, and c gives

$a = 1$

$b = 1000.0$

$c = 2.22 * 10^6$

Using equation 5.8 gives roots

root 1 = -666.67

root 2 = -1333.33

Thus equation 5.8 becomes

$(s + 666.67)(s + 1333.33)$

and the transfer function becomes

$$\frac{1}{(s + 666.67)(s + 1333.3)} \qquad (5.9)$$

Form the table of Laplace transforms in Chapter 5 of Kreysig, [5], this transform can be converted to the time domain giving

$$f(t) = e^{at} - e^{bt} \qquad (5.10)$$

where a and b are the two roots 1 and 2 of equation 5.8. For a fixed sampling interval T of 2.3 ms, equation 5.10 becomes

$$f(t) = e^{-0.733} - e^{-1.467} \qquad (5.11)$$

The transducer output f_n is multiplied by this factor after each sample is taken, thus giving the lag in the input to the ADC.

The ADC quantization is simulated simply by rounding the force calculated at the output of the filter by the gradient of the calibration curve, in this case 70.313, and then rounding it to the nearest integer less than the filter output. This gives a value for f_{nadc}, which can then be compared to the similarly quantised f_{sq} calculated from the set-point force f_s.

Finally, the sewing machine can be simulated by reducing the length of untensioned elastic, L_n, between the sewing head and the nip rollers by an amount 1/C, where C is the number of stiches per inch formed by the sewing machine. This is done for each simulated stitch formation.

Using these equations allows a loop to be constructed which calculates the actual axial force f_n in the elastic at each sample interval of the control system, and the effect of the filter and A-D converter to be taken in to account in determining the force read in by the control system. The strain $(e/L)_n$ can then be plotted against time to give a graph of the performance if the control system whilst elastic is fed to a sewing machine.

5.2 Open-loop control system simulation.

The open-loop system response was simulated in a slightly different way. The calculation of f_n following a motor step is the same, but the filter and ADC simulation was not needed. Pre-tensioning the elastic simply used the number of motor steps calculated from equation (2.4), and calculated the axial force f_n after each step. Once sewing was assumed to have started, each execution of the simulation loop represented a pulse being received from the opto-switch attached to the sewing machine. For each stitch formed by the sewing machine, two pulses are received from the opto-switch. Equation (2.4) was used to calculate the increment constant V1, and a decision made on whether or not to step the motor. The new axial force f_n could then be calculated. Again, a plot of $(e/L)_n$ against time was obtained, with time being determined by the sewing machine speed. It was assumed that the system could perform the neccessary calculations in the time between pulses being received, since the simulation speed of the sewing machine was within the design operating range of the system.

5.3 Comparison of the two systems' performance.

There are two main areas in which the open-loop and closed-loop algorithms can be compared : the initial physical installation and information required for correct operation, and the practical problems encountered in the actual operation of the systems.

From the description of the two control algorithms in sections 2.1 and 3.1, it can be seen that the systems have quite different requirements for installation and operation. The physical installations are shown in figs. 7 and 19. The open-loop system needs to be accurately positioned physically with regard to the sewing machine, whereas the closed-loop system is independant of this positioning. The number of stitches per inch at which the sewing machine is set is also required by the open-loop system and not by the closed-loop system. The open-loop system has to have sensing of the sewing machine rotation, i.e. stitch formation, which the closed-loop system does not. The only information required by the closed-loop system are the original and reduced lengths of the workpiece to be precessed, which are available anyway, since that is the information given to the operator when she receives new workpieces. Normal tension control mode operation of the two systems is shown in figs. 20 and 21. The initial slope on both graphs show the systems pre-tensioning to the desired strain of 0.1 (10% stretch). Stitching starts after 0.25s, and elastic is fed to the sewing head. Both systems maintain the strain at within 1% of desired under these normal conditions.

However, there are two main causes of loss of tension in the elastic during otherwise normal sewing operations. The first is slippage of the elastic through the nip rollers due to 'chalking', as described in Chapter 1.2. The result of this can be seen in figs. 22 and 23. The figures show that the effect on

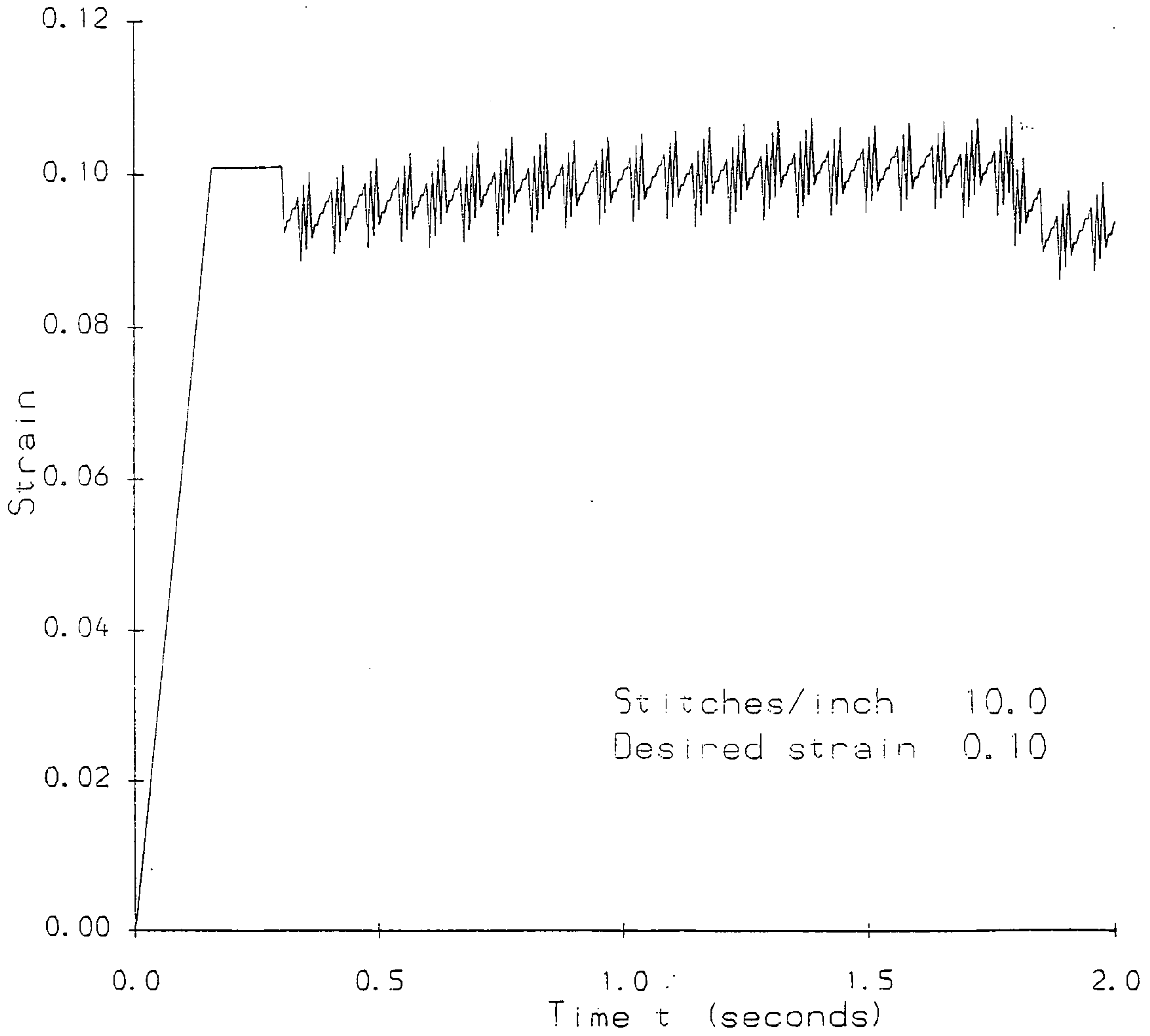

Figure 20. Simulated response of the open-loop system under normal operating conditions.

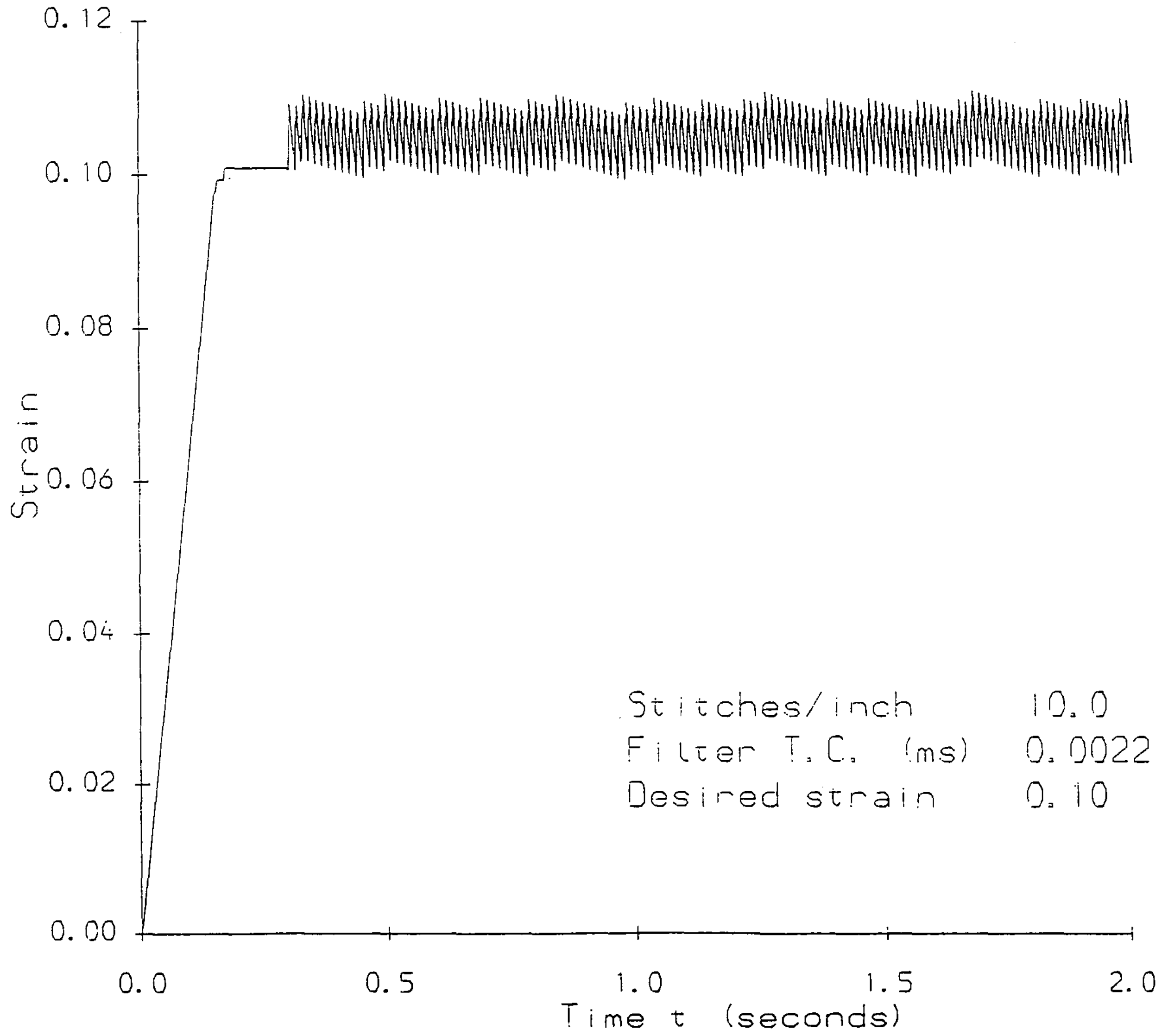

Figure 21. Simulated response of the closed-loop system under normal operating conditions.

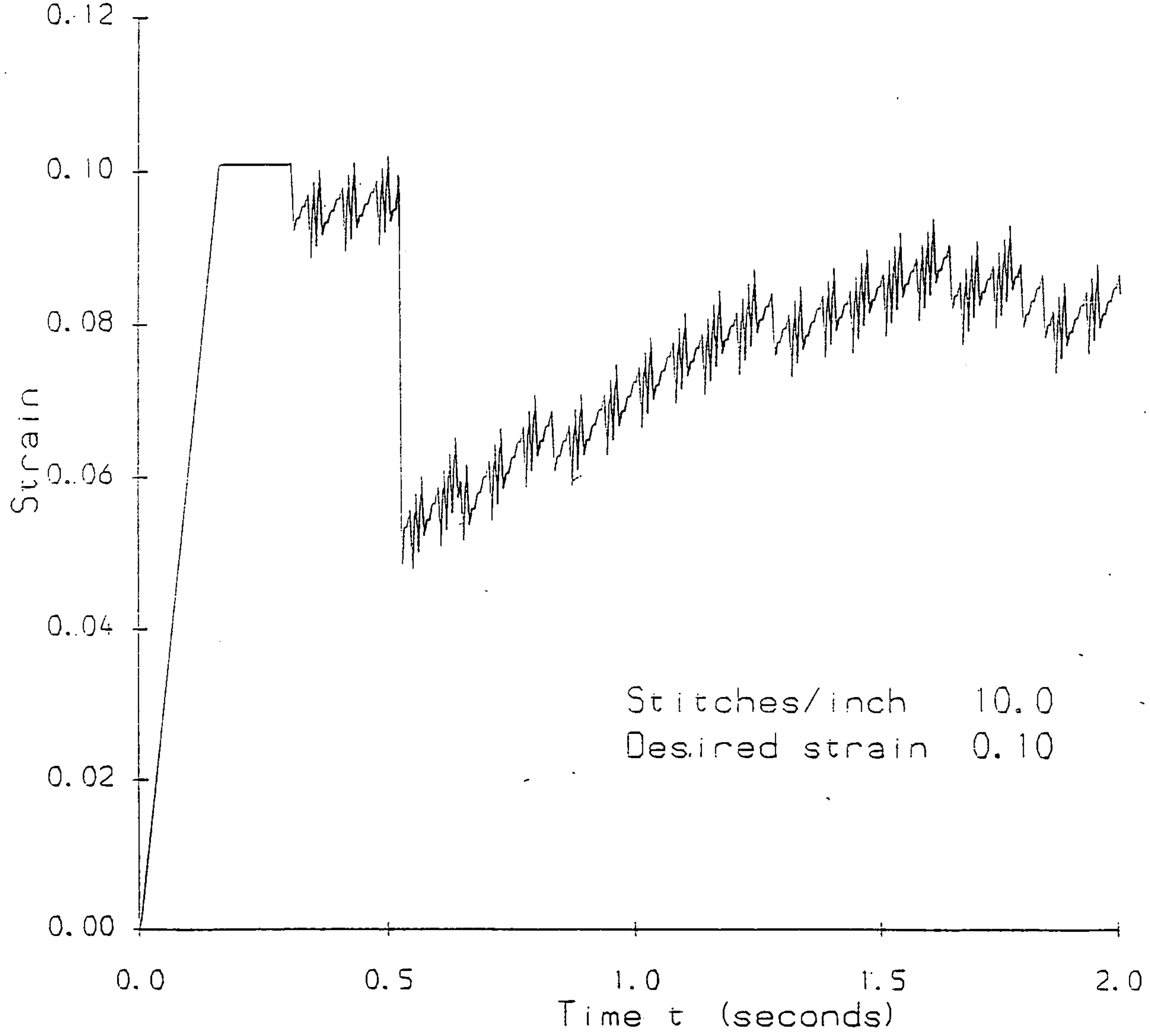

Figure 22. Simulated response of the open-loop system with elastic slippage through the nip rollers

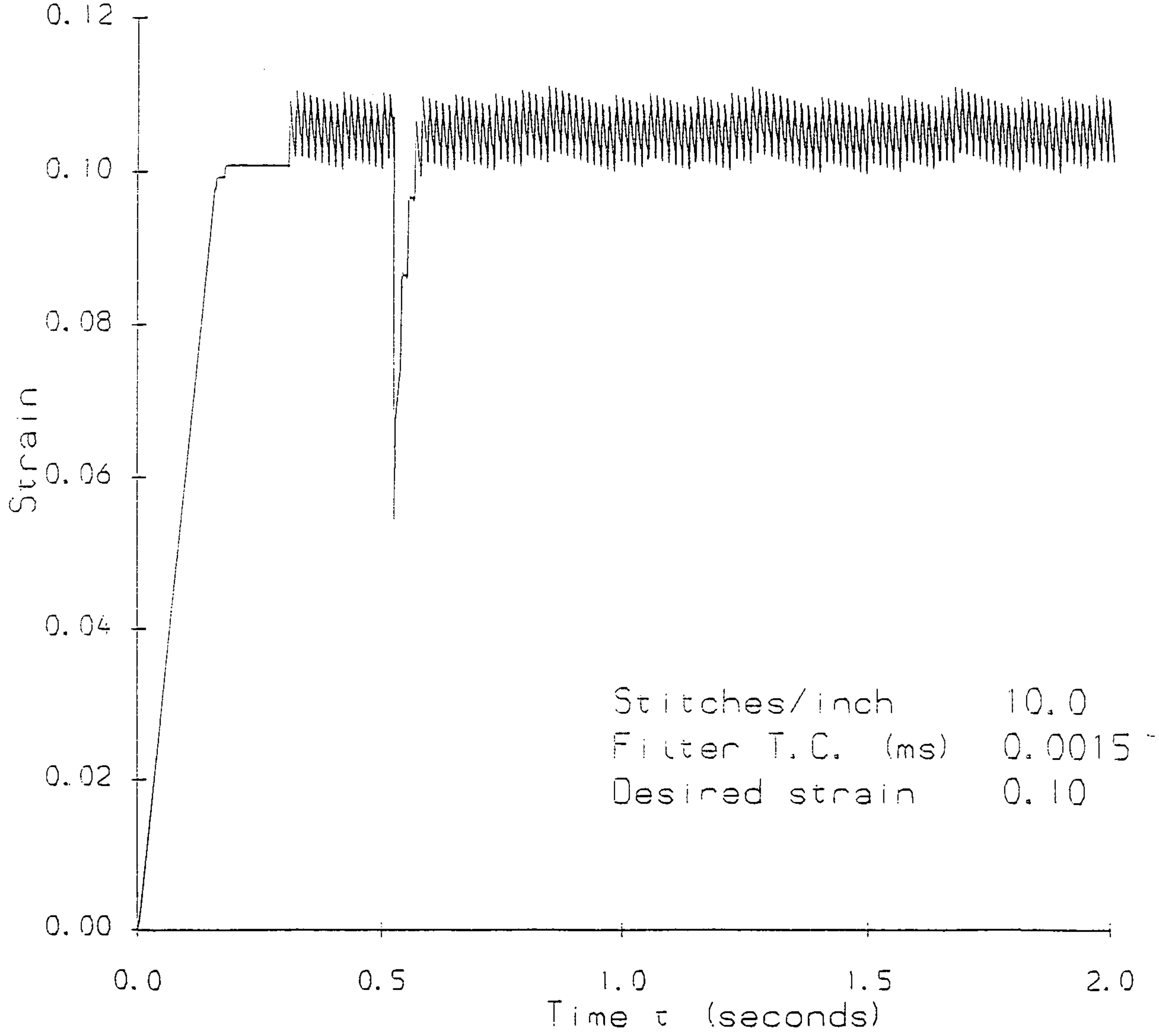

Figure 23. Simulated response of the closed-loop system with elastic slippage through the nip rollers.

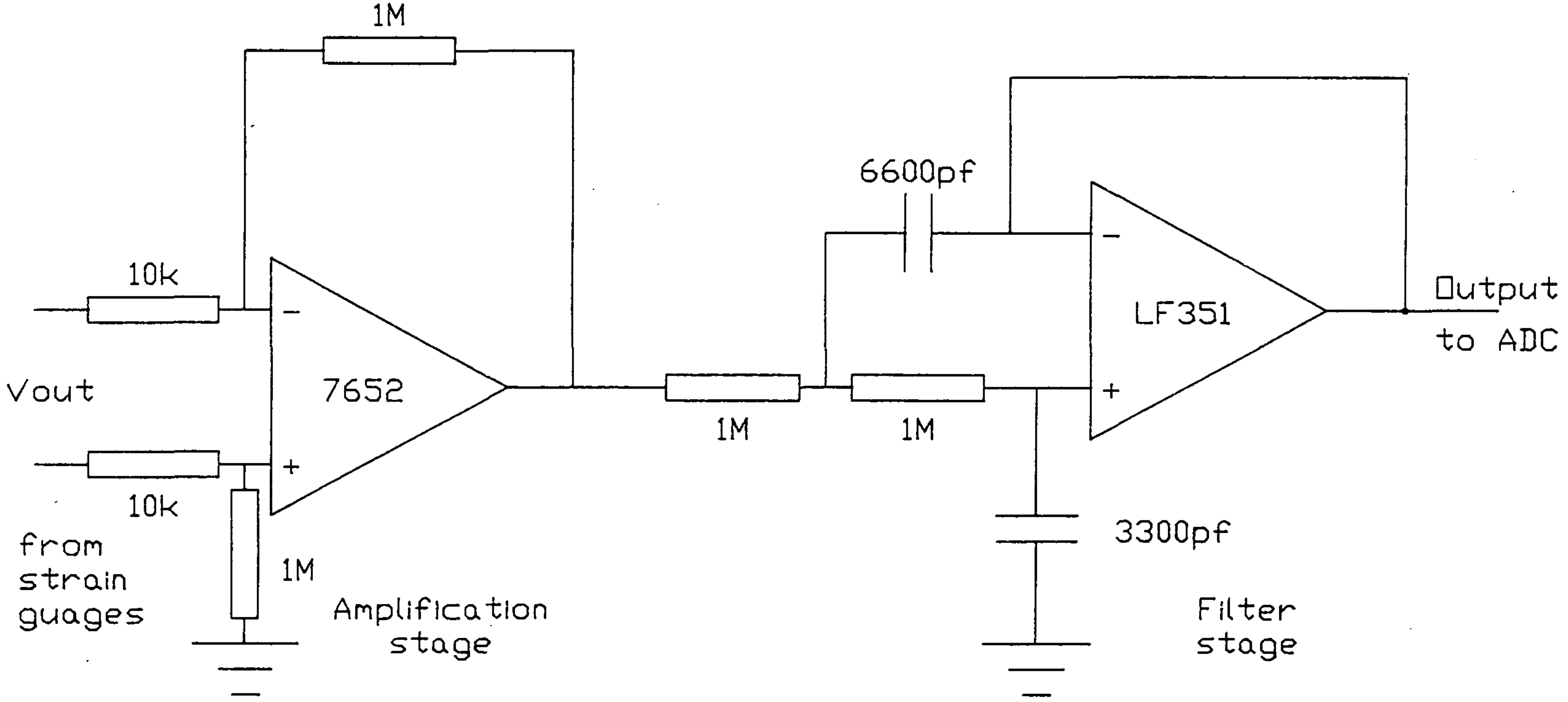

Figure 24. Transducer interface amplification and filtering.

the closed-loop system performance is considerably less than that on the open-loop system. The closed-loop system recovers the loss of tension within 3 stitches, whereas the open-loop system takes approximately 175 stitches to recover the loss. Both systems were feeding elastic at 10% stretch to a sewing machine operating as described earlier. The length of elastic to slip through the nip rollers was 10mm. In the case of the closed-loop system, it would be almost impossible to spot any variation in garment quality due to the slippage, but the open-loop system will have fed under-tensioned elastic for 444mm of stitching, thus producing a sub-standard garment.

The second effect which results in a loss of tension is electrical noise, which causes false information to be fed to the controller, or a false output to be produced. The effect is basically the same as slippage but on a smaller scale, with the closed-loop system responding so quickly that the effect is hardly noticeable, giving a better maintained tension, and hence more consistent quality garments than with the open-loop system.

The added advantage of the automatic size-checking and adaptive control of the workpiece sizing performed by the closed-loop system as described in Chapter 3. gives significant improvements in performance over the open-loop system.

CHAPTER 6

CONCLUSIONS

Two microprocessor-based systems have been developed for the feed of elastic tape at a precise tension for attachment to garments.

The first is an open-loop control system. Each stitching cycle of the sewing machine is sensed by the system, and the elastic is fed through a pair of stepper motor driven nip rollers at such a rate as to maintain the tension in the elastic at a pre-determined level input to the system. Tension can be maintained to within 2% of that desired, thus offering improvements over the mechanical tensioning systems previously available.

However, the response of open-loop control systems to problems such as slippage of the elastic tape through the nip rollers feeding it to the sewing machine, which cause a loss of tension in the elastic, is very slow. Sub-standard garments may then be produced, and may not be spotted by quality control, since it is not possible for every garment to be checked for correct sizing.

An adaptive closed-loop control system has been developed to overcome this problem. It uses a specially developed force transducer to sense the tension in the elastic. The control system then uses this value to vary the feed through the nip rollers in such a way as to maintain the tension at the desired level. Thus, any loss of tension is rapidly recovered, preventing incorrect sizing of the workpiece. Tension can be controlled to within 2% of that desired. However, a further control loop senses the beginning and end of each workpiece as it is processed, and calculates the amount of untensioned elastic attached to it. Given the original length and the required reduced length of the workpiece, the actual reduced length of the workpiece can be calculated and compared to that desired. The set-point for the elastic tension control loop is adjusted accordingly to maintain the correct dimensions of the next workpiece. Each workpiece is therefore checked for correct sizing as it is processed, reducing the chance of sub-standard garments being produced and hence improving quality control.

A simple man-machine interface futher enhances the system performance by enabling the operator to adjust her workpiece feed rate in to the sewing machine. Tension imparted to the workpiece is thus reduced, further improving garment quality.

Future work on thid project would include futher extensive testing of the stand-alone system in an industrial environment, and exploration in to the possibilities of producing a commercial product.

REFERENCES

[1] Burdess, I.R. "Feed Control Procedure." Patent Application number 8215999, June, 1982.

[2] Motorola "M6800 Application Manual." 1975 first edition. Motorola.

[3] Higgins, R.A. "Properties of Engineering Materials." 1985 revision. Hodder and Stoughton. London,Sydney,Auckland & Toronto. 289-295.

[4] Horowitz, P., Hill, W. "The Art of Electronics." 1985 edition. Cambridge University Press, Cambridge. 92-171.

[5] Kreyszig, E. "Advanced Engineering Mathmatics." Fourth edition. J. Wiley & Sons. New York, Chichester, Brisbane, Toronto. 200-248,759

[6] Coates, R.F. "Flight 68K User's Manual." Second edition 1985. Flight Electronics, Southampton.

[7] King, T., Knight, B. "Programming the M68000." Second edition 1987. Benjamin/Cummings Publishers Inc., Wokingham.

[8] Burr-Brown Corporation "TM2500/TM2700 OEM Microterminals Manual MA51414." Revision B June 1987. Burr-Brown Corporation.

[9] RS Components Ltd "IC Timers 555 and 556" Data sheet 2113 July 1984. RS Components Ltd, Corby.

[10] RS Components Ltd "RS427 Analog-to-digital Converter" Data sheet 4052 (discontinued). RS Components Ltd, Corby.

[11] Marshall, S.J.L., Sarhadi, M. "A Microprocessor-based Elastic Feed System." Proceedings of ACME grantholders conference, Nottingham University, September 1988.

[12] Marshall, S.J.L., Sarhadi, M. "Implementation of Metrology Systems for Quality Inspection of Garments." Xth International Conference on Production Research, Nottinham University, August 1989.

[13] Marshall, S.J.L., Sarhadi, M. "Elastic Feed Control Systems." IFAC Symposium on Low Cost Automation Techniques, Components and Instruments, Applications, Milan, November 1989.

APPENDIX A

LISTING OF 68000 ASSEMBLER ROUTINES FOR THE STAND-ALONE CLOSED-LOOP CONTROL SYSTEM

Appendix A.1

Listing of the routine PITDATA, which performs the tension control and adaption for the system.

```
          |routines for i/o control of the pit.

_convert: movb #12,d0                       |initialise conversion by sending a low
          addb stepos,d0
          movb d0,a1@(padr)                 |pulse via bit 0 of port a to the SC pin
          movb #13,d0                       |on the 502.
          addb stepos,d0
          movb d0,a1@(padr)
conl:     btst #1,a1@(padr)                 |check for end of conversion bit 1
          beq conl
          movb stepos,d1
          addb #9,d1
          movb d1,a1@(padr)                 |enable 8 MSB's of result
          movb a1@(pbdr),d0                 |read in to d0
          andw #0xff,d0                     |strip higher 8 bits of d0
          aslw #2,d0                        |shift left twice to form 10 bits.
          movb stepos,d1
          addb #5,d1
          movb d1,a1@(padr)                 |enable two LSB's
          movb a1@(pbdr),d1                 |read in to d1
          rolb #2,d1                        |transfer to 2 LSB's of d1
          andw #0x03,d1                     |strip top 14 bits of d1
          addw d1,d0                        |add the two to get one 10-bit result
          movw d0,reading                   |in d0,and store in reading.
          movb stepos,d1
          addb #13,d1
          movb d1,a1@(padr)
          rts


_testcon: movb #0x0c,d0                      |initialise conversion by sending a low
          movb d0,a1@(padr)                 |pulse via bit 0 of port a to the SC pin
```

```
           movb #0x0d,d0              |on the 502.
           movb d0,a1@(padr)
 tcon1:    btst #1,a1@(padr)         |check for end of conversion bit 1
           beq tcon1
           movb #9,d1
           movb d1,a1@(padr)         |enable 8 MSB's of result
           movb a1@(pbdr),d0         |read in to d0
           andw #0xff,d0             |strip higher 8 bits of d0
           aslw #2,d0                |shift left twice to form 10 bits.
           movb #5,d1
           movb d1,a1@(padr)         |enable two LSB's
           movb a1@(pbdr),d1         |read in to d1
           rolb #2,d1                |transfer to 2 LSB's of d1
           andw #0x03,d1             |strip top 14 bits of d1
           addw d1,d0                |add the two to get one 10-bit result
           movw d0,reading           |in d0,and store in reading.
           movb #13,d1
           movb d1,a1@(padr)
           rts

_advance:  movb #0x5d,d2             |advance motor by one step.
           movb d2,a1@(padr)         |bit 4 is motor direction (1-clockwise)
                                     |                          (0-anti- "  ).
           movb #0x1d,d2             |bit 5 is clock pulse to pk2 drive.
           movb d2,a1@(padr)
           andb #0x30,d2
           movb d2,stepos
           rts

_reverse:  movb #0x4d,d2             |reverse motor by one step.
           movb d2,a1@(padr)
           movb #0x0d,d2
           movb d2,a1@(padr)
           andb #0x30,d2
           movb d2,stepos
           rts

_plysens:  cmpb #0,cutflag
           beq plyex
           movb a1@(padr),d1      |routine to sense the beginning and end of
           andb #0x80,d1
           bne plysens1
           andb #0x80,prevstat    |plies fed through the sewing machine,and to
           bne plysens2           |calculate the number of steps performed.
           jsr _delay2
           bra plyex              |requires prevstat,stepcnt=0 before sensing
                                  |starts.
plysens1:  andb #1,dirflag
           bne plysen10
           movw revcnt,d0
           addw #1,d0
           movw d0,revcnt
           bra plysen11
plysen10:  movw stepcnt,d0        |if either present or previous state are 1,
           addw #1,d0             |then count continues.
                 movw d0,stepcnt      |if prevstat=1 and present state=0 then end
```

```
plysen11: movb d1,prevstat         |count and re-initialise prevstat.
          movb #0,endflag
          jsr _delay2
          bra plyex                |stepcnt and endflag are re-initialised in
                                        |the subroutine for calculating whether the
plysens2: movb #1,endflag          |correct amount of elastic has been fed in to
          movb #0,prevstat         |the ply.
          movb #0,cutflag          |clear flag for cutter control.
          movw stepcnt,d0
          movw revcnt,d1
          subw d1,d0
          movw d0,stepcnt
          cmpw sizeh,d0
          ble plysens3
          movl #sizemes1,a5
          jsr _outm
                bra plysens5
plysens3: cmpw sizel,d0
          bge plysens4
          movl #sizemes2,a5
          jsr _outm
                bra plysens5
plysens4: movl #sizemes3,a5
                   jsr _outm
plysens5: jsr _plyerr
          andb #1,calflag
          beq plysens8
          movw setpt1,d0
          subw zero,d0
          movw d0,setpt1
          movw setpt2,d0
          subw zero,d0
          movw d0,setpt2
          movw stepcnt,d0
          movw zsize,d1
          cmpw d1,d0
          bgt plysen50
          subw d0,d1
          movw d1,d0
                bra plysen51
plysen50:  subw  d1,d0               |compare stepcnt with no. of steps for 0%
plysen51: mulu #100,d0             |stretch.
          movw stepcnt,d1            |calculate the actual % stretch obtained
          jsr _div                 |in the elastic : result in d0.b
          movw d0,acstrain
          trap #11
          .word 0x0a
          trap #11
          .word 0x03
          movw strain,d0
          trap #11
          .word 0x0a
          trap #11
          .word 0x03
          movw stepcnt,d0
          movw sizeref,d1
```

```
           cmpw d1,d0          |compare desired steps with actual steps.
           blt plysens6
           beq plysens7
           subw d1,d0          |calculate sifference between sizeref & stepcnt
           mulu #100,d0        |desired > actual, increase setpts for control
           movw d1,d0
           movw sizeref,d1     |by a factor : ((stepcnt-sizeref)*100)/sizeref+100
           jsr _div            |result in d0.
           addw #100,d0
           movw setpt1,d1      |alter setpt 1 by factor
           mulu d0,d1
           movw d1,d0
           movw #100,d1
           jsr _div            |result in d0
           addw zero,d0
           movw d0,setpt1            |store new setpt value.
           trap #11
           .word 0x0a
           trap #11
           .word 0x03
           movw setpt1,d0
           subw #20,d0
           movw d0,setpt2            |store new setpt 2 value.
           addb #1,calcount
           bra plysens8
plysens6:  subw d0,d1          |actual > desired,decrease setpts for control
           mulu #200,d1
           movw d1,d0
           movw stepcnt,d1
           jsr _div            |result in d0.
           movw #100,d1
           subw d0,d1
           movw setpt1,d0      |alter setpt 1 by factor
           mulu d1,d0
           movw #100,d1
           jsr _div            |result in d0
           addw zero,d0
           movw d0,setpt1      |store new setpt value.
           trap #11
           .word 0x0a
           trap #11
           .word 0x03
           movw setpt1,d0
           subw #20,d0
           movw d0,setpt2            |store new setpt value.
           addb #1,calcount
                bra plysens8
plysens7:  movb #0,calflag
           movw setpt1,d0
           addw zero,d0
           cmpw zero,d0
           bgt plysen70
           movw zero,setpt1
                bra plysen71
plysen70:    movw  d0,setpt1
plysen71:  subw #20,d0
```

```
          cmpw zero,d0
          bgt plysen72
          movw zero,setpt2
               bra plysens8
plysen72:   movw  d0,setpt2
plysens8: movw stepcnt,d0
          movw #0,stepcnt
          jsr _delay3
             movw #0,revcnt
plyex:    rts

_plyerr:  movw sizeref,d2
          movw stepcnt,d1                 |sutract number of steps from desired number
          movw stepcnt,d0
          trap #11
          .word 0x0a
          trap #11
          .word 0x03
          movw d2,d0
          cmpw d0,d1
          bgt plyerr1
          subw d1,d0
          movl #sizemes5,a5
          jsr _outm
                 bra plyerr2
plyerr1:  subw d0,d1
          movl #sizemes6,a5
          jsr _outm
                  movw d1,d0
plyerr2:  mulu #100,d0
          movw #273,d1
          jsr _div
          jsr _outnum
          movl #sizemes4,a5
          jsr _outm
          movw stepcnt,d0
          movw sizeref,d1
          addw #6,d1
          cmpw d1,d0
          bgt plyerrx
          subw #12,d1
          cmpw d1,d0
          blt plyerrx
             movb #0,calflag
plyerrx:  rts


          |delay for required length before cutting elastic tape.

_cutcon:  andb #1,cutflag           |check if flag set for cutting
          bne cutex                 |if not then check number of steps
          movw stepcnt,d1           |done since end of ply
          cmpw length,d1            |and cut when
          ble cut1                  |the step count is greater than desired.
          jsr _cuton
          jsr _cutoff
```

```
         movw #0,stepcnt          |clear step count and set cut flag.
         movb #1,cutflag          |when cut done.
         bra cutex
cut1:    andb #1,dirflag
         bne cut2
         subw #1,d1
         bra cut3
cut2:     addw #1,d1              |increment step count if waiting to cut.
cut3:    movw d1,stepcnt
cutex:   rts

_cuton:  movb #0xff,a1@(pcdr)
         rts

_cutoff: movb #0x00,a1@(pcdr)
         rts

sizemes1: .ascii "\nToo big "
         .byte 7
         .byte 0
         .even

sizemes2: .ascii "\nToo small "
         .byte 7
         .byte 0
         .even

sizemes3: .ascii "\nO.K."
         .byte 0
         .even

sizemes4: .ascii "mm\r"
         .byte 0
         .even

sizemes5: .ascii "-"
         .byte 0
         .even

sizemes6: .ascii "+"
         .byte 0
         .even
%
```

Appendix A.2

Listing of the routine CALC, which calculates the initial set-point for the tension control algorithm based on initial values for E, A_o, v, and desired stretch in the elastic. Also included is the routine SIZESET, which calculates the number of motor steps required to feed the correct amount of untensioned elastic in to the sewing head for the size of garment being made.

```
         |Calculation of desired force from initial conditions:
         |   strain desired,E*Ao,v.
         |following the equation
         |
         |      F=E*Ao*e/L*(1-(e/L*v))^2
         |
_calc:  movw strain,d0          |check if strain is 0.
        bne calc1
        movw #0,d0
        addw zero,d0
        movw d0,setpt2
        addw #18,d0
        movw d0,setpt1
        bra calcex
        movl #calerr1,a5
        bra calc3
calc1:  movw v,d0               |check if v is 0.
        bne calc2
        movl #calerr2,a5
           bra calc3
calc2:  movw EAo,d0
        bne calc4
        movl #calerr3,a5
calc3:  jsr _outm               |output error message
        bra calcex
calc4:  movw strain,d0         |put strain value mantissa in d0.
        mulu v,d0              |multiply by Poisson's ratio
        movl #10,d1             |divide result by 10
        jsr _div
        movl #0x2710,d1         |subtract the result from 10000 to give
        subl d0,d1              |equivalent of (1-(e/L*v)).
        mulu d1,d1              |square (1-(e/L*v)).
        movl d1,d0              |put result in d0 to be divided.
```

```
          movl #0x2710,d1        |divisor 10000.
          jsr _div
          movl #10,d1            |divisor 10
          jsr _div               |now have (1-(e/L*v))^2 in d0.
          movw EAo,d3           |multiply E*Ao by strain.
          mulu strain,d3
          mulu d3,d0             |multiply strain*E*Ao by (1-(e/L*v))^2.
          movl #0x2710,d1        |divide result by 10000.
          jsr _div
          movl #10,d1            |divide again by 10.
          jsr _div               |have desired force now stored in d0.
          movw d0,sforce
          mulu #110,d0
          movl #100,d1
          jsr _div
          addw zero,d0
          movw d0,setpt1
          subw #0x14,d0
           movw d0,setpt2
calcex:   rts


          |division subroutine with rounding of result up or down.
          |divisor in d1,number to be divided in d0,result in d0.w.

_div:     divu d1,d0             |divide d0 by d1 - store result in d0.
          movl d0,d2             |copy d0 to d2.
          lsrl #8,d2             |shift d2 logically right 16 bits.
          lsrl #8,d2             |
          andl #0xffff,d2        |get rid of quotient.
          lsrl #1,d1             |halve d1 (divisor)
          cmpl d1,d2             |compare remainder with half of the divisor.
          blt divex              |leave result as it was (rounding down).
          addw #1,d0             |add 1 to result (rounding up)
divex:    andl #0xffff,d0        |remove remainder from d0.
          rts                    |return from sub-routine (rounded result in d0)


_divr:    divu d1,d0             |divide d0 by d1 - store result in d0.
          movl d0,d1
          lsrl #8,d1             |remainder in d1
          lsrl #8,d1
          andl #0xffff,d1        |strip quotient from d1
          andl #0xffff,d0        |remove remainder from d0.
          rts                    |return from sub-routine (result in d0)

           |routine to get size values for amount of elastic from tables.

_setsize: movl #size,a3
          movw #0,d2
sizel:    movw a3@(0,d2:w),d1
          cmpw sizedes,d1             |Find desired size entry in sizetble.
          beq size2
          addb #4,d2
```

```
        bra size1
size2:   movw a3@(2,d2:w),d0     |read number of steps for correct size
        mulu #100,d0            |at 0% stretch into d0.
        movw strain,d1
        addw #100,d1            |calculate actual number of steps at
        jsr _div                |the current strain setting.
        movw d0,sizeref
        trap #11
        .word 0x0a
        trap #11
        .word 0x03
        movw sizeref,d0         |add tolerances to the desired size.
        addw #25,d0
        movw d0,sizeh
        subw #50,d0
        movw d0,sizel
        rts                     |return from subroutine.

calerr1: .ascii "Error:strain=0\r"
        .byte 7
        .byte 3
        .byte 0

calerr2: .ascii "Error:v=0\r"
        .byte 7
        .byte 3
        .byte 0

calerr3: .ascii "Error:EAo=0\r"
        .byte 7
        .byte 3
        .byte 0
        .even
%
```

APPENDIX B

LISTING OF THE CLOSED-LOOP CONTROL SYSTEM

SIMULATION PROGRAM

```
PROGRAM  LOOPANAL(INPUT,OUTPUT);
CONST
 dX=0.399;
 L0=203.2;
 EA0=7.86;
 v=0.400;
 LIMIT=500;
 LITERAL_SIZE=255;

TYPE
 COORDINATES=ARRAY[1..LIMIT] OF SHORTREAL;
 LITERAL=PACKED ARRAY[1..LITERAL_SIZE] OF CHAR;

VAR
 MAXSTRAIN,STRAINn,STRAINnm1,Ln,Lnm1,dFn,Fn,Fnm1,Fnm2,
 Dnm1,Dnm2,Fs,Fk,Fkm1,Fkm2,R,Ca,a,b,c,ROOT1,ROOT2,
 FACTOR,T,POWER0,POWER1,POWER2,TIME_CONSTANT,STITCHES_PER_INCH :SHORTREAL;

 ERRORn,CONTROLLER_OUTPUT,LOOP,Fsq,
 Fnadc,MOTOR_STEP   :INTEGER;

 X,Y:COORDINATES;

 OPTION:CHAR;

PROCEDURE CALCULATE_FILTER_PARAMETERS;
BEGIN
 R:=1000000;
                   READLN(Ca);
{READLN(STITCHES_PER_INCH);
 READLN(OPTION);}
 STITCHES_PER_INCH:=10.0;
 OPTION:='Y';
 T:=0.0023;
 a:=1;
 b:=(1/(2*R*Ca))+(1/(R*Ca));
 c:=1/(2*sqr(R)*sqr(Ca));
 ROOT1:=-b+sqrt(sqr(b)-(4*a*c));
 ROOT2:=-b-sqrt(sqr(b)-(4*a*c));
```

```
  POWER0:=1/(SQRT(2)*Ca*R);
  POWER1:=-(ROOT1/2);
  POWER2:=-(ROOT2/2);
  writeln(power1:3:3,'  ',power2:3:3);
  TIME_CONSTANT:=R*Ca;
 END;

 PROCEDURE  SIMULATE_CONTROLLER_ACTION;
 BEGIN
  ERRORn:=Fsq-Fnadc;
  IF ERRORn>0 THEN
   CONTROLLER_OUTPUT:=1
  ELSE
   IF ERRORn<0 THEN
    CONTROLLER_OUTPUT:=-1
   ELSE
     CONTROLLER_OUTPUT:=0;
 END;

 PROCEDURE  SIMULATE_MOTOR_ACTION;
 BEGIN
   MOTOR__STEP:=CONTROLLER_OUTPUT;
 END;

 PROCEDURE  CALCULATE_Ln;
 BEGIN
  IF MOTOR_STEP>0 THEN
   Ln:=Lnml-(dX/(1+STRAINnml))
  ELSE
   IF MOTOR_STEP<0 THEN
    Ln:=Lnml+0.399;
 END;


 PROCEDURE  CALCULATE__STRAINn;
 BEGIN
   STRAINn:=(L0/Ln)-1;
 END;

 PROCEDURE  CALCULATE_dFn;
 BEGIN
   dFn:=EA0*((STRAINn*SQR(1-STRAINn*v))-(STRAINnml*SQR(1-STRAINnml*v)));
 END;

 PROCEDURE  CALCULATE_Fn;
 BEGIN
  Fn:=Fnml+dFn;
  IF Fn<0 THEN
   Fn:=0;
 END;

 PROCEDURE  SIMULATE_ELASTIC;
 BEGIN
  CALCULATE_Ln;
  CALCULATE_STRAINn;
  CALCULATE_dFn;
```

```
 CALCULATE_Fn;
END;

PROCEDURE  SIMULATE_FILTER_RESPONSE;
BEGIN
 Fk:=Fkm1+(Fn-Fkm1)*((1-EXP(-POWER0*0.0023))
                       +(POWER2/POWER0)*SIN(FACTOR)*EXP(-FACTOR));
END;

PROCEDURE  SIMULATE_ADC_QUANTISATION;
BEGIN
 IF Fk>0 THEN
  Fnadc:=ROUND(Fk*70.313)
 ELSE
   Fnadc:=0;
END;


PROCEDURE  SIMULATE__SEWING_MACHINE;
BEGIN
 IF LOOP>100 THEN
  BEGIN
   IF (LOOP/10)=ROUND(LOOP/10) THEN
     Ln:=Ln-((1/(STRAINn+1))*(25.4/STITCHES_PER_INCH));
  END;
END;


PROCEDURE  DISPLAY_CURRENT_VALUES;
BEGIN
 WRITELN('Fn ',Fn:2:3,' Fnadc ',Fnadc:2,' STRAINn ',
          STRAINn:2:3,' Fk ',Fk:2:3); END;

PROCEDURE  UPDATE_VARIABLES;
BEGIN
 IF STRAINn>MAXSTRAIN THEN
  MAXSTRAIN:=STRAINn;
 X[LOOP]:=LOOP;
 Y[LOOP]:=STRAINn;
 Fnm1:=Fn;
 Fkm1:=Fk;
 Lnm1:=Ln;
 STRAINnm1:=STRAINn;
 LOOP:=LOOP+1;
END;

PROCEDURE PAPER(CONST N:INTEGER);FORTRAN77;
PROCEDURE  MAP(CONST XM1,XM2,YM1,YM2:REAL);FORTRAN77;
PROCEDURE AXES;FORTRAN77;
PROCEDURE  CURVEO(CONST  X,Y:COORDINATES;CONST M,N:INTEGER);FORTRAN77;
PROCEDURE    PSPACE(CONST    XMIN,XMAX,YMIN,YMAX:SHORTREAL);FORTRAN77;
PROCEDURE CTRMAG(CONST Imag:INTEGER);FORTRAN77;
PROCEDURE  PLOTCS(CONST X,Y:SHORTREAL;CONST PHRASE:LITERAL);FORTRAN77;
PROCEDURE  CTRORI(CONST Quad:SHORTREAL);FORTRAN77;
PROCEDURE  TYPECS(CONST PHRASE:LITERAL);FORTRAN77;
PROCEDURE  TYPENF(CONST R:SHORTREAL;CONST Ndp:INTEGER);FORTRAN77;
```

```
 PROCEDURE    FRAME;FORTRAN77;
 PROCEDURE    GREND;FORTRAN77;
 PROCEDURE   BLUPEN;FORTRAN77;
 PROCEDURE   BLKPEN;FORTRAN77;
 PROCEDURE  REDPEN;FORTRAN77;

 BEGIN
  Fk:=0;
  Fkm1:=0;
  Fkm2:=0;
  Fnm1:=0;
  Fnm2:=0;
  Lnm1:=L0;
  STRAINn:=0;
  Fnadc:=0;
  Ln:=0;
  LOOP:=1;
  STRAINnm1:=0;
  MAXSTRAIN:=0;
  x[1]:=0;
  y[1]:=0;
  Fs:=0.980;
  Fsq:=ROUND(Fs*70.313);
  CALCULATE_FILTER_PARAMETERS;
  FACTOR:=POWER1*T;
  SIMULATE_CONTROLLER_ACTION;
  REPEAT
   BEGIN
    SIMULATE_MOTOR_ACTION;
    SIMULATE_ELASTIC;
    IF OPTION='Y' THEN
      SIMULATE_SEWING_MACHINE;
    SIMULATE_FILTER_RESPONSE;
    SIMULATE_ADC_QUANTISATION;
    SIMULATE_CONTROLLER_ACTION;
    UPDATE_VARIABLES;
   END;
             UNTIL            LOOP>LIMIT;
MAXSTRAIN:=(ROUND((MAXSTRAIN*50)+1))/50;
PAPER(1);
PSPACE(0.1,1.00,0.12,0.95);
REDPEN;
CTRMAG(15);
PLOTCS(0.65,0.25,'Stitches/inch     ');
TYPENF(STITCHES_______PER______INCH,1);
PLOTCS(0.65,0.20,'Filter  T.C. (ms) ');
TYPENF(TIME________________CONSTANT,4);
PLOTCS(0.65,0.15,'Feed/step  (mm)   ');
TYPENF(dX,3);
BLKPEN;
PLOTCS(0.43,-0.08,'Time  t ');
CTRORI(90.0);
PLOTCS(-0.1,0.47,'Strain');
CTRORI(0.0);
BLUPEN;
MAP(0.0,LIMIT,0.0,MAXSTRAIN);
```

```
CURVEO(X,Y,1,LIMIT);
BLKPEN;
AXES;
FRAME;
GREND;
END.
```

APPENDIX C

LISTING OF THE OPEN-LOOP CONTROL SYSTEM PROGRAM

```
PROGRAM  LOOPANAL(INPUT,OUTPUT);
CONST
 dX=0.359;
 L0=254;
 DESIRED_STRAIN=0.10;
 LIMIT=500; LITERAL_SIZE=255;
 STITCHES_PER_INCH=10.0;
 F=1.795;
 MACHINE_SPEED=5200;
 BASE_VALUE=256;

TYPE
 COORDINATES=ARRAY[1..LIMIT] OF SHORTREAL;
  LITERAL=PACKED ARRAY[1..LITERAL_SIZE] OF CHAR;
VAR
 COUNTER,NO_OF_STEPS,P,C,STRAINn,STRAINnml,Ln,Lnml,MAXSTRAIN,
 EA0,NOISE,INCREMENT_CONSTANT:shortreal;

 ERRORn,CONTROLLER_OUTPUT,LOOP,Fsq,
 Fnadc,MOTOR_STEP  :INTEGER;

 SLIPPED:BOOLEAN;

 X,Y:COORDINATES;

 OPTION:CHAR;

 TITLE:LITERAL;

PROCEDURE  RANDOM_NOISE;
BEGIN
   NOISE:=100*RANDOM(0);
END;


PROCEDURE  CALCULATE_Ln;
BEGIN
 IF MOTOR_STEP>0 THEN
  Ln:=Lnml-(dX/(1+STRAINnml))
 ELSE
  IF MOTOR_STEP<0 THEN
   Ln:=Lnml+0.399;
```

```
 END;


 PROCEDURE  CALCULATE__STRAINn;
 BEGIN
   STRAINn:=(L0/Ln)-1;
 END;

 PROCEDURE  PRETENSION_ELASTIC;
 BEGIN
 MOTOR_STEP:=1;
    CALCULATE_Ln;
     CALCULATE_STRAINn;

 END;

 PROCEDURE  SIMULATE_SEWING_MACHINE;
 BEGIN
   IF (LOOP/2)=ROUND(LOOP/2) THEN
       Ln:=Ln-(1/(1+STRAINn))*(1/(1+STRAINn))*(25.4/STITCHES_PER_INCH);
 END;

 PROCEDURE  SIMULATE_SLIPPAGE;
 BEGIN
   Ln:=Ln+10.0;
 END;

 PROCEDURE  CONTROLLER_ACTION;
 BEGIN
  INCREMENT_CONSTANT:=BASE_VALUE*(1-P*P/C);
  COUNTER:=INCREMENT_CONSTANT+COUNTER;
  IF COUNTER>255 THEN
    COUNTER:=COUNTER-256
   ELSE
      Ln:=Ln+F;
 END;

 PROCEDURE  DISPLAY_CURRENT_VALUES;
 BEGIN
   WRITELN('STRAINn ',STRAINn:2:3,' X[LOOP] ',X[LOOP]:2:3,' Y[LOOP] ',Y[LOOP]:2:3)
 ;
 END;

 PROCEDURE  UPDATE__VARIABLES;
 BEGIN
  IF STRAINn>MAXSTRAIN THEN
   MAXSTRAIN:=STRAINn;
  X[LOOP]:=X[LOOP-1]+1/173.4;
  Y[LOOP]:=STRAINn;
  Lnm1:=Ln;
  STRAINnm1:=STRAINn;
  LOOP:=LOOP+1;
 END;

 PROCEDURE PAPER(CONST N:INTEGER);FORTRAN77;
 PROCEDURE  MAP(CONST XM1,XM2,YM1,YM2:REAL);FORTRAN77;
```

```
PROCEDURE AXES;FORTRAN77;
PROCEDURE   AXESSI(CONST  DX,DY:SHORTREAL);FORTRAN77;
PROCEDURE  CURVEO(CONST  X,Y:COORDINATES;CONST M,N:INTEGER);FORTRAN77;
PROCEDURE    PSPACE(CONST    XMIN,XMAX,YMIN,YMAX:SHORTREAL);FORTRAN77;
PROCEDURE CIRMAG(CONST Imag:INTEGER);FORTRAN77;
PROCEDURE  PLOTCS(CONST X,Y:SHORTREAL;CONST PHRASE:LITERAL);FORTRAN77;
PROCEDURE  CIRORI(CONST Quad:SHORTREAL);FORTRAN77;
PROCEDURE  TYPECS(CONST PHRASE:LITERAL);FORTRAN77;
PROCEDURE  TYPENF(CONST R:SHORTREAL;CONST Ndp:INTEGER);FORTRAN77;
PROCEDURE   FRAME;FORTRAN77;
PROCEDURE   GREND;FORTRAN77;
PROCEDURE  BLUPEN;FORTRAN77;
PROCEDURE  BLKPEN;FORTRAN77;
PROCEDURE REDPEN;FORTRAN77;

BEGIN
 TITLE:='Open Loop System Response With Stitching.';
 SLIPPED:=FALSE;
 Lnm1:=L0;
 STRAINn:=0;
 Ln:=0;
 LOOP:=2;
 STRAINnm1:=0;
 MAXSTRAIN:=0;
 x[1]:=0;
 y[1]:=0;
 y[2]:=0;
 COUNTER:=0;
 C:=1/(0.708*10/STITCHES_PER_INCH);
 P:=1/(1+DESIRED_STRAIN);
 NOISE:=RANDOM(ROUND(STITCHES_PER_INCH*10));
 REPEAT
  BEGIN
   PRETENSION_ELASTIC;
   UPDATE_VARIABLES;
   X[LOOP-1]:=(LOOP-1)*0.0023;
  END;
 UNTIL STRAINn>DESIRED_STRAIN;
 REPEAT
  BEGIN
   UPDATE_VARIABLES;
  END;
 UNTIL X[LOOP-1]>0.3;
 REPEAT
  BEGIN
   SIMULATE_SEWING_MACHINE;
   RANDOM_NOISE;
   IF (NOISE>98.2) OR (NOISE<0.9) THEN
    CONTROLLER_ACTION;
   CONTROLLER_ACTION;
   CALCULATE_STRAINn;
   UPDATE_VARIABLES;
   IF (X[LOOP-1]>0.52) AND (SLIPPED=FALSE) THEN
    BEGIN
     SIMULATE_SLIPPAGE;
     SLIPPED:=TRUE;
```

```
      END;
    END;
                    UNTIL                    LOOP>LIMIT;
MAXSTRAIN:=(ROUND((MAXSTRAIN*50)+1))/50;
PAPER(1);
PSPACE(0.15,0.95,0.15,0.5);
Blkpen;
CTRMAG(10);
PLOTCS(0.5,0.25,'Stitches/inch   ');
TYPENF(STITCHES____PER____INCH,1);
PLOTCS(0.5,0.20,'Desired  strain ');
TYPENF(DESIRED_STRAIN,2);

BLKPEN;
PLOTCS(0.43,-0.15,'Time  t (seconds)');
CTRORI(90.0);
PLOTCS(-0.10,0.47,'Strain');
CTRORI(0.0);
MAP(0.0,2.0,0.0,0.12);
CURVEO(X,Y,1,LIMIT);
BLKPEN;
CTRMAG(10);
AXESSI(0.5,0.02);
FRAME;
GREND;
END.
```