

DEVELOPMENT OF AN ADAPTIVE
KALMAN FILTER FOR ESTIMATION
IN CHEMICAL PLANT

A thesis submitted for the degree of
Doctor of Philosophy
by

ROBERT NORMAN WEBB

Department of Chemical Engineering
University of Aston in Birmingham

October 1977

DEVELOPMENT OF AN ADAPTIVE KALMAN FILTER FOR
ESTIMATION IN CHEMICAL PLANT

ROBERT NORMAN WEBB

OCTOBER 1977

A thesis submitted for the degree of Doctor of Philosophy

SUMMARY

The application of the Kalman Filter to the on-line estimation of the state of a chemical process has so far met with a limited amount of success due to the inaccuracies and non-linearities of the mathematical models developed to describe the process being studied. To overcome these problems two theoretical developments are proposed; the first being an adaptive form of the Kalman Filter which compensates for modelling errors by finding the mean and covariance of a number of "fictitious inputs" and the second a numerical technique for determining the state transition matrix of non-linear systems by the use of eigenvector theory.

Following a series of off-line experiments, these modified forms of the Kalman Filter were applied to the on-line estimation of the state of a pilot plant scale double effect evaporator. The work involved can be conveniently divided into the following three sections.

First of all a seventh order mathematical model of the evaporator together with suitable heat transfer correlations and compatible with the Kalman Filter was derived and then tested by comparing the simulated responses with plant data.

Secondly two major software packages were developed,

(i) The Hados Executive Package which was used for interactive data acquisition and,

(ii) The On-line Digital Filtering Package (OLDFP), a real time operating system which controls the execution and data acquisition of filtering programs written in Fortran.

Finally a series of on-line filtering experiments were carried out, the results of which show that the proposed theoretical developments considerably improve the performance of the Kalman Filter by eliminating bias and divergence.

KEY WORDS - KALMAN FILTER, ON-LINE COMPUTATION, MATHEMATICAL MODELLING, EIGENVECTOR THEORY, SIMULATION.

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to the following:-

Professor G.V. Jeffreys, Head of the Department of Chemical Engineering, University of Aston in Birmingham, for providing the facilities for this research to be carried out in his department.

Dr. B. Gay, for his supervision of this project and constructive comments on the preparation of this thesis.

Dr. J.P. Fletcher, for his many useful suggestions and encouragement during this research project.

Messrs M. Lea and D. Bleby, departmental electronics technicians, for their help in maintaining the electronic equipment associated with the project.

The Science Research Council, for their financial support.

Miss J. Morton, for her accurate and patient typing of this thesis.

CONTENTS

	<u>Page</u>
1. <u>INTRODUCTION</u>	
2. <u>LITERATURE REVIEW</u>	
2.1 INTRODUCTION	5
2.2 ESTIMATION AND IDENTIFICATION THEORY	8
2.3 HEAT EXCHANGER AND EVAPORATOR MODELLING	55
2.4 CHAPTER REVIEW	73
3. <u>THEORETICAL DEVELOPMENTS</u>	
3.1 INTRODUCTION	75
3.2 ADAPTIVE FILTERING	77
3.3 EXCESSIVE NON-LINEARITIES	100
3.4 CHAPTER REVIEW	118
4. <u>SIMULATION STUDIES OF THE KALMAN FILTER</u>	
4.1 INTRODUCTION	119
4.2 THEORETICAL BACKGROUND	121
4.3 THE ICL 1904s COMPUTER	133
4.4 DISCUSSION OF COMPUTER PROGRAMS	135
4.5 RESULTS AND DISCUSSION	148
4.6 CHAPTER REVIEW	169
5. <u>THE DOUBLE EFFECT EVAPORATOR/COMPUTER SYSTEM</u>	
5.1 INTRODUCTION	170
5.2 THE DOUBLE EFFECT EVAPORATOR	171
5.3 THE HONEYWELL 316/HADIOS DATA ACQUISITION SYSTEM	179
5.4 CHAPTER REVIEW	187
6. <u>HONEYWELL 316 COMPUTER SOFTWARE</u>	
6.1 INTRODUCTION	188
6.2 STANDARD SOFTWARE	190
6.3 GRAPHICAL SOFTWARE	207
6.4 THE ASTON SIMULATION PACKAGE - ASP	212
6.5 THE HADIOS EXECUTIVE PACKAGE MK.2	216
6.6 THE OLDFP EXECUTIVE	225
6.7 CHAPTER REVIEW	233

	<u>Page</u>
7. <u>STEADY STATE ANALYSIS OF THE DOUBLE EFFECT EVAPORATOR</u>	
7.1 INTRODUCTION	234
7.2 INSTRUMENT CALIBRATION	236
7.3 ON-LINE STEADY STATE EXPERIMENTS	242
7.4 CHAPTER REVIEW	264
8. <u>DYNAMIC ANALYSIS OF THE DOUBLE EFFECT EVAPORATOR</u>	
8.1 INTRODUCTION	266
8.2 DEVELOPMENT OF A DYNAMIC MODEL OF THE DOUBLE EFFECT EVAPORATOR	268
8.3 SIMULATION OF THE DYNAMIC MODEL	292
8.4 RESULTS AND DISCUSSION	294
8.5 CHAPTER REVIEW	301
9. <u>ON-LINE FILTERING EXPERIMENTS</u>	
9.1 INTRODUCTION	302
9.2 DISCUSSION OF THE ON-LINE FILTERING PROGRAMS	303
9.3 EXPERIMENTAL PROCEDURE	318
9.4 RESULTS AND DISCUSSION	326
9.5 CHAPTER REVIEW	342
10. <u>CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK</u>	
10.1 CONCLUSIONS	343
10.2 RECOMMENDATIONS FOR FUTURE WORK	347

APPENDICES

- APPENDIX A - RESULTS OF THE SIMULATION STUDIES OF
THE KALMAN FILTER.
- APPENDIX B - ASSEMBLER LEVEL PROGRAMMING OF THE H316
COMPUTER.
- APPENDIX C - THE HADIOS EXECUTIVE PACKAGE MK.2.
- APPENDIX D - OP-16 REAL TIME OPERATING SYSTEM.
- APPENDIX E - THE OLDFP EXECUTIVE.
- APPENDIX F - LISTINGS OF BASIC COMPUTER PROGRAMS.
- APPENDIX G - THE ON-LINE FILTERING PROGRAMS -
FILTER2, 3 AND 4.
- APPENDIX H - STEADY STATE ANALYSIS OF THE DOUBLE
EFFECT EVAPORATOR.
- APPENDIX J - DYNAMIC ANALYSIS OF THE DOUBLE EFFECT
EVAPORATOR.
- APPENDIX K - RESULTS OF ON-LINE FILTERING EXPERIMENTS.

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

The fundamental objective when designing, constructing and operating physical systems is to convert some readily available raw material into a required commodity with a desired quality. To achieve this objective it is necessary to be able to control the physical system at a state at which the raw materials are efficiently converted into the product. The action of control is essentially a decision making process and as such requires an accurate knowledge of the state of the system under study. The problem of accurately determining the state of a system from noisy measurements is called estimation or filtering and is the main subject of this thesis.

The complexity of the processes found within the chemical industry and the speed at which decisions need to be made has led to the use of high speed digital computers to help the engineer in the estimation and control of the state of the system. Computers are particularly useful when using modern control schemes where two sources of information are available for the estimation of the state of the physical system. The two sources of information available are

measurements of process variables and a mathematical description of the physical system. In chemical engineering systems both are subject to error.

The combination of information from measurements and a mathematical description of the system to provide 'best' estimates of the state of a process is an area of recursive estimation or filtering which has been the subject of much research for many years. In 1960 Kalman (1.1) extended filtering theory to cover the estimation of states described by a set of linear ordinary differential equations and following this development the classical Kalman Filter has been widely applied in many diverse fields, in particular the aerospace industry. Unfortunately, the use of the classical Kalman Filter in the chemical industry has been rather limited due to the following reasons;

(i) A lack of knowledge of the process under consideration.

(ii) Chemical processes usually yield complicated mathematical models which are generally non-linear, of a high order and often consist of partial differential equations.

The objective of this thesis is to develop modifications to the classical Kalman Filter which will overcome some of the problems outlined above. The effectiveness of the proposed modifications will then be demonstrated first of all by a series of simulated experiments and secondly by the real time, on-line estimation of the state of a pilot plant scale double effect evaporator whose dynamic characteristics are poorly understood.

THESIS OUTLINE BY CHAPTER

Chapter 2 reviews the relevant literature concerning the development of estimation theory and in particular the problems associated with real time estimation for chemical processes. Finally, a brief review of heat exchanger dynamics, relevant to double effect evaporators, is presented.

Chapter 3 discusses two theoretical developments which should improve the performance of the classical Kalman Filter. The first of these developments is an adaptive form of the Kalman Filter which compensates for a poor mathematical description of the process under study and the

second is a numerical procedure for overcoming the non-linearities of a process.

The results of a series of simulated experiments using various forms of the Kalman Filter are presented and discussed in chapter 4.

The double effect evaporator, its operation and instrumentation and the link to the H316 computer system are described in chapter 5.

Chapter 6 describes the computer programs available for on-line experiments with the double effect evaporator. This description includes the standard manufacturers software and applications packages developed as part of this research.

The development of a mathematical model to describe the processes occurring in the double effect evaporator is described in chapters 7 and 8; chapter 7 is concerned with a steady state analysis and chapter 8 with the development and testing of a dynamic model.

Chapter 9 describes the on-line implementation of the forms of the Kalman Filter developed in chapter 3. Finally the experimental results obtained are presented and discussed.

Chapter 10 presents the conclusions made as a result of this research and makes recommendations for further study.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

The operation of all types of chemical engineering process plant requires some form of control system to maintain the product at some predetermined quality. The types of control systems now available range from the process operator through to on-line control using high speed digital computers. As the sophistication of the control system used increases so does the required knowledge of the process. For example, a process operator will be aware of simple relationships between operating conditions and measurements and adjust the controls accordingly, whereas, for process control and optimisation using modern multivariable control theory, a detailed mathematical model describing both steady and dynamic states, must be derived in order to satisfy the requirements of this theory for accurate values of all the state variables necessary to describe the system. The derivation of an accurate mathematical model often proves to be a source of difficulty as limitations are usually imposed by the physical information available, the type of mathematical model used and the number of measurable variables. Furthermore, the task is often complicated by two main factors; (i) the measurements made are often corrupted by noise caused either by the components of the measuring device or other nearby electrical systems. (ii) The system itself may be subject to random disturbances.

The problem of determining the state of a system

from noisy measurements is called estimation or filtering and, due to its central importance in engineering, has been the subject of man's attention since the pioneering work of Gauss (2.2) and Legendre (2.1). A wide variety of techniques have been developed during this period and, in general terms, each one depends upon the type of model used, the availability of experimental measurements and the nature of variables of special interest. With the advent of high speed data processing more advanced techniques of data analysis have become available.

The purpose of this review is to survey those techniques applicable to on-line process identification and in particular examine the reported applications and modifications of a type of recursive estimation known as Kalman Filtering. The problem of extending this linear theory to non-linear systems is considered together with reported difficulties of implementing this technique in real time. Finally the mathematical modelling of evaporators and heat transfer equipment is briefly reviewed.

2.2. ESTIMATION AND IDENTIFICATION THEORY

2.2.1. HISTORICAL DEVELOPMENT

The recovery of information from measurements corrupted by uncertainty has long been a struggle endured by many an investigator. Probably the earliest attempts at a solution to this problem were those proposed by Legendre (2.1) in 1806 and Gauss (2.2) in 1809. The fundamental publication by Gauss describes his attempts at determining the orbital elements of a celestial body from available data. The method used has come to be known as the "Method of Least Squares" and since its conception, whenever an investigator has been confronted with data suspected of containing random errors, the "most probable" or "best" estimate of the desired parameter is computed by means of some variation of this method. The method of least squares is a deterministic approach to the problem of minimising errors and, in basic terms, attempts to pass the solution of a physical model as closely as possible through the measurements made. The most serious drawback of this method is that all measurements need to be available before minimisation can take place. Consequently, as the number of measurements increase, the computational requirements involved in obtaining a solution can become prohibitive. Deutsch (2.3) gives a review of the many aspects and applications of this approach to estimation.

Perhaps the first two significant advances in estimation theory were due to Pearson (2.4) and Fisher (2.5).

Pearson developed a technique known as the "Method of Moments" which is no longer widely used as it has been established that the estimates obtained were not the best possible from the point of view of efficiency. Fisher demonstrated that the method of maximum likelihood was usually superior to the method of moments and that estimates derived by this technique could not be essentially improved.

The estimators discussed above are merely smoothing procedures, and in no way make use of any known statistics of the system under consideration in order to provide more accurate estimates of its state. Theoretical considerations show that this could be accomplished if the correct weighting matrix for the data could be computed for the system considered. However, Deutsch (2.3) and Swerling (2.6) both agree that this can be rarely accomplished in practical applications.

The need for some new type of estimator making use of more a priori information was made painfully obvious by a rapid development of communication theory by engineers and physicists. Communication theory, as originally conceived, was applied to the transmission of intelligence by electrical means. At this time the major concern was with the effect of random processes, or noise, on the intelligibility of signals within communication channels. The first attempts at the reduction of the effects of noise were in the proper direction but were severely limited because of the lack of an estimation theory that could be

used to synthesize the required noise separation filters.

A fresh approach to estimation theory came with the use of known system statistics. Researchers in this field have termed this the probabilistic approach to the problem. The foundations of estimation theory using this method are attributed to the original and parallel developments of Wiener (2.7) and Kolmogorov (2.8). These works, although containing complex mathematical treatments, offered for the first time an analytical synthesis technique which could be used for the separation of a desired signal from an environment of undesired noise. Wiener's theory appears to be essentially a least squares estimation process. However, he made elegant use of the fact that he was estimating parameters from input data in the form of a stochastic process. The technique derived is based upon frequency domain analysis and reduces to the solution of a complex integral equation (the Wiener-Hopf equation). Using this method solutions can be obtained for estimation problems relating to a class of linear, stationary systems. An engineer would classify such systems as simple when compared to the more important cases of non-linear and non-stationary problems. These more complex problems remain essentially unsolved despite various modifications and extensions to the Wiener-Kolmogorov theory which have been proposed. The conclusion reached by most authors is that the assumptions required to enable the solution of the Wiener-Hopf integral for complex systems renders the theory inadequate for non-linear estimation.

2.2.2. PROCESS IDENTIFICATION

Since the publication of the Wiener-Kolmogorov theory, an increasing amount of attention has been focused on the problem of determining, from input/output data, useful mathematical descriptions of dynamic systems. The introduction of the high speed digital computer has added extra impetus to this trend by facilitating the implementation of sophisticated data processing operations. The field has subsequently become rather diverse and a precise definition of the term "Identification" is somewhat difficult to come by. Astrom (2.9) defines identification as, "the determination, on the basis of input controls and output measurements of a mathematical model equivalent to the process under consideration", and this being very general is more applicable than most.

The task of subdividing identification into different classes is simplified by the fact that the technique used is usually dictated by the type of mathematical model used to represent the system. In the final analysis this amounts to the nature of the basic structural assumptions made about the system under consideration. General reviews of identification techniques have been published by Nieman et al. (2.10), Seinfeld (2.11) and Cuenod and Sage (2.12). Seinfeld states that the following broad classes of problems require estimation techniques:

(i) The determination of model parameters in non-linear algebraic models from experimental measurements, e.g., the determination of chemical reaction rate

constants from experimental reaction rate measurements.

(ii) The determination of model states and parameters in non-linear dynamic models from laboratory and plant output data, e.g., the modelling and simulation of non-linear dynamic processes from operating data.

(iii) The on-line modelling of dynamic processes where a continuous output signal is used to generate instantaneous estimates of the states and parameters in process models.

The techniques involved in the solution of the third class of problem are classified as sequential or non-sequential estimators. In the sequential approach, estimates of state variables and parameters are generated at each sampling instant. Non-sequential estimators are based on taking a series of samples over a known time interval. This thesis is primarily concerned with the solution of the third class of problem using a sequential technique.

A sequential solution to the estimation problem is commonly referred to as a filter because current state and parameter estimates are calculated as the output measurements become available, hence continuously filtering the system. The filtering problem can thus be 'identified' as the estimation of the current state of a dynamic system using all past and present measurements.

2.2.3. KALMAN FILTERING

The general linear non-stationary filtering problem was essentially solved using the concept of state

variables in the classical method due to Kalman (2.13), Kalman and Bucy (2.14) and Kalman (2.15). The technique used manages to remove the inherent difficulties in the solution of the Wiener-Hopf integral by substitution of an equivalent non-linear differential equation. It is this non-linear differential equation, which on solution, yields the covariance matrix of the minimum filtering error. In turn this matrix contains all of the information necessary for the design of the optimum filter. This approach has the practical advantage that the optimum filter can be synthesised in a sequential manner and thus is often readily implemented in real time without the attendant problems of data storage requirements encountered in earlier least squares approaches.

The Kalman-Bucy method (hereafter referred to as the Kalman Filter) can be summarised by the following five relations, (2.14):

(i) The differential equation governing the optimal filter which is excited by the observed signals and generates the best linear estimate of the message.

(ii) The differential equations governing the error of the best linear estimate.

(iii) The time-varying gains of the optimal filter expressed in terms of the error variances.

(iv) The non-linear differential equations governing the covariance matrix of the errors of the best linear estimate, called the variance equation.

(v) The formula for prediction.

A continuous time linear dynamical system can be described either by a differential equation formulation or by a linear difference equation. For the purpose of this thesis a system characterised by difference equations will be considered.

The time-discrete Kalman Filter is composed of matrix recursion relations, the simplicity of which makes them particularly amenable to implementation on a digital computer. The derivation of these relations is accomplished in a manner that relies more upon physical intuition than upon mathematical sophistication. A number of different derivations are provided by Sorenson (2.16).

The Kalman Filter algorithm provides an optimal estimate of the state of a linear, time varying, dynamic system observed sequentially in the presence of additive white gaussian noise; such systems are referred to as linear stochastic dynamic systems. The estimate obtained at each time is optimal in the sense that it is the maximum likelihood estimate conditioned on all observations up to that time. Such a system can be described by the following vector set of differential equations:

$$\dot{x}(t) = A(t) \cdot x(t) + B(t) \cdot z(t) \quad - (2.1)$$

$$y(t) = M(t) \cdot x(t) + v(t) \quad - (2.2)$$

where, t denotes time,

x is an $n \times 1$ vector of state variables,

y is an $m \times 1$ vector of measurements,

Z is a $p \times 1$ vector of random system disturbances,

represented by a zero mean, white gaussian noise process,
 v is an $m \times 1$ vector of random measurement noise,
represented by a zero mean, white gaussian noise process,
 A is an $n \times n$ matrix of coefficients,
 B is an $n \times p$ matrix of coefficients,
 M is an $m \times n$ measurement matrix.

By integration of equation 2.1 the system is converted to discrete time and can be represented by the vector set of difference equations,

$$x(k+1) = \phi(k+1,k) x(k) + \Gamma(k+1,k) \cdot B(k) \cdot z(k) \quad - (2.3)$$

$$y(k) = M(k) \cdot x(k) + v(k) \quad - (2.4)$$

where, ϕ is the $n \times n$ state transition matrix,

Γ is the $n \times p$ integral state transition matrix,

$$k = k \cdot \Delta t \text{ and } \Delta t = t_{k+1} - t_k.$$

For a stationary linear system, the transition matrices are given by,

$$\phi(k+1,k) = \exp (A \cdot t) \quad - (2.5)$$

$$\Gamma(k+1,k) = \int_{t_k}^{t_{k+1}} \exp (A(t_k + \Delta t - \tau)) \cdot d\tau \quad - (2.6)$$

The respective variances of $u(k)$ and $v(k)$ are given by,

$$Q(k) = E(z(k) \cdot z(k)^T) \quad - (2.7)$$

$$R(k) = E(v(k) \cdot v(k)^T) \quad - (2.8)$$

where, superscript T denotes matrix transposition.

Finally, assuming that the initial state prediction, $x(0,-1)$, together with its error covariance, $P(0,-1)$, are known, then the linear Kalman Filter determines the estimate of state and minimises the quantity J as follows:

$$J = \frac{1}{2} (\bar{x}(k) - x(k,k))^T \cdot P(k,k)^{-1} + \frac{1}{2} \sum_{j=0}^{k-1} ((y(j+1) - M(j+1) \cdot x(j+1,j))^T \cdot R(j+1)^{-1} \cdot (y(j+1) - M(j+1) \cdot x(j+1,j))) - (2.9)$$

where, $\bar{x}(k)$ is the true state of the system,

$x(j,i)$ denotes the estimate of the state x at time j given measurements up to time i ,
 $i \leq j$,

$P(j,i)$ denotes the covariance of the error in this estimate, and,

superscript -1 denotes matrix inversion.

The Kalman Filter algorithm may be written as a set of prediction and estimation equations as follows:-

Prediction:

$$x(k+1,k) = \phi(k+1,k) \cdot x(k,k) - (2.10)$$

$$P(k+1,k) = \phi(k+1,k) \cdot P(k,k) \cdot \phi(k+1,k)^T + \Gamma(k+1,k) \cdot B \cdot Q(k+1) \cdot B^T \Gamma(k+1,k)^T - (2.11)$$

Estimation:

$$K(k+1) = P(k+1,k) \cdot M^T(k+1) \cdot (M(k+1) \cdot P(k+1,k) \cdot M(k+1)^T + R(k+1))^{-1} - (2.12)$$

$$x(k+1,k+1) = x(k+1,k) + K(k+1) (y(k+1) - M(k+1) \cdot x(k+1,k)) - (2.13)$$

$$P(k+1,k+1) = (I - K(k+1) \cdot M(k+1)) \cdot P(k+1,k) - (2.14)$$

$$\text{or, } \quad = (I - K(k+1). M(k+1)) . P(k+1,k). (I - K(k+1). M(k+1))^{T} + K(k+1). R(k+1). K(k+1)^{T} \quad - (2.15)$$

where, $K(j)$ is an $n*m$ weighting or filter gain matrix.

It can be seen by examination of equation 2.13 that the estimate of the state vector, $x(k+1,k+1)$, is the sum of the predicted state and the weighted measurement error, $K(k+1). (y(k+1) - M(k+1). x(k+1,k))$. The original derivation of the algorithm (2.13) points out a number of valuable features of estimates thus calculated. These are that the estimate is uniformly asymptotically stable and that the convergence of the variances of the estimates, as each successive measurement is processed, is insensitive to round off errors, provided the system is observable and controllable.

The dual concepts of observability and controllability were introduced by Kalman (2.13) in his original work as a part of his general filtering theory. Coggan and Noton (2.17) interpret these concepts as follows:

(i) A system is observable, if with perfect measurements and no random disturbances, all the state variables can be determined after a finite number of measurements.

(ii) A system is controllable if all the states are excited by the random disturbances $u(k)$.

The more important concept to the field of filtering theory is that of observability. If one of the state variables of the system can not affect the measurement

vector then the consequent unobservability leads to the estimates produced by the filter being sub-optimal.

The time taken for the filter to converge has been shown by Storey (2.18) to depend on the accuracy of the initial state estimate, $x(0,-1)$, and the initial error covariance, $P(0,-1)$. The larger the initial error in the estimates, the longer the filter takes to converge. This is because until the system has become observable, the filter relies on the initial state estimate. For the case of a large initial error covariance matrix, the filter will display a slower response since it will rely initially upon current noisy measurements.

The statement of the filter algorithm, equations 2.10 to 2.15, gives alternatives, equations 2.14 and 2.15, for determining the error covariance of the state estimations. It has been shown by Aoki (2.18) that equation 2.16 is preferable since the right hand side is the sum of two symmetric positive definite matrices while equation 2.14 is at best the difference of two positive definite matrices. Thus, equation 2.15 is better conditioned for numerical computation and will retain the positive definiteness and symmetry of $P(k,k)$.

Following the original work of Kalman the main contributions to the field of linear filtering theory have been aimed at generalising Kalman's earlier work.

The most significant of these are developments by Cox (2.20), Kushner (2.21) and Friedland and Bernstein (2.22) in extending the theory to allow for correlation between the system and measurement noise. The problem of coloured (time-correlated) noise is partially solved by Kalman (2.15). A good review of these generalisations is given by Jazwinski (2.23).

2.2.4. EXTENSION TO NON-LINEAR SYSTEMS

The theoretical developments discussed in the preceding section have been based upon systems which can be described by a set of first order, linear differential equations, the outputs of the systems being provided by quantities that are linearly related to the state variables. For such systems the theory developed by Kalman provides an optimal filter. Unfortunately, such a model is not immediately applicable to most engineering problems of significance except as a very rough approximation. The dynamical system for most engineering problems is frequently found to be described by a series of n th order non-linear differential equations. The fact that n th order derivatives appear does not offer any theoretical difficulty because any n th order differential equation may be written as a system of n first order differential equations. However, the output of the system commonly appears in terms of quantities that have a non-linear algebraic relation with the state variables and this

poses considerable theoretical and practical problems if an optimal solution is required. The application of the Kalman Filter to such problems may seem strange due to its completely linear basis. However, its computational ease and conceptual simplicity, not to mention its success with large numbers of linearisable problems, has led to a large number of suggested modifications to allow its extension to non-linear systems.

Exact optimal solutions to non-linear estimation problems can be obtained theoretically using general recursion relations describing the evolution of the conditional probability density function in terms of the a priori distributions and the measurement data. This conditional probability density function of the state conditioned on the measurement data contains all the available information that can be used in the development of estimation and control policies for stochastic dynamical systems with noisy measurements. In a comparison of several filters for non-linear estimation problems, Alspach (2.24) states that despite the considerable interest shown to this problem in the literature, it is seldom possible to express the conditional probability density function in a form that can be used to generate specific estimation policies. This is due to the fact that such expressions require a precise knowledge of an infinite set of parameters.

Using finite dimensional approximations two techniques have been proposed to allow the approximate solution of the recursion relations discussed above. The first of these is due to Bucy and Senne (2.25) and is based upon a specific rotating variable density grid, numerical integration technique. The second method due to Alspach and Sorenson (2.26), uses a method involving the approximation of certain densities by a sum of Gaussian like densities with positive weighting coefficients. The main problem encountered in implementing either of these techniques on a practical basis is the excessively expensive amount of computation required when compared to the most advanced modification of the Kalman Filter. Alspach (2.24) and Jazwinski (2.23) are in agreement on this point, and both of them consider that this fact alone gives even greater importance to the work done in applying Kalman's original theory to non-linear systems.

The theoretical development which enables such applications is basically an extension of the linear theory in which a Taylor series expansion, neglecting second and higher order terms, is used to linearise the state and/or the measurement functions about a nominal trajectory. The resulting algorithm has come to be known as the 'Extended Kalman Filter'. The derivations reported by Jazwinski (2.23), Kushner (2.27) and Sorenson (2.16) are representative of many found in the literature. The fundamental assumption made in these derivations is that a nominal solution of the non-linear

differential equations must not only exist, but also provide a 'good' approximation to the actual behaviour of the system. A summary of the derivation is given below.

Consider the non-linear stochastic system described by

$$x(k+1) = f(x(k), u(k)) + z(k) \quad - (2.16)$$

$$y(k) = M(x(k), u(k)) + v(k) \quad - (2.17)$$

where, z and v are as defined for linear systems, and u is a vector of system inputs.

If the system equations are linearised about the current estimate of state, $x(k,k)$, by means of a truncated Taylor series expansion, the state transition matrix becomes,

$$\phi(k+1,k) = I + \Delta t. \left. \frac{\partial f(x(k), u(k))}{\partial x(k)} \right|_{x(k,k)} \quad - (2.18)$$

Similarly, if the measurement equation is expanded about the current predicted state, the measurement matrix becomes,

$$M(k+1) = I + \Delta t. \left. \frac{\partial M(x(k), u(k))}{\partial x(k)} \right|_{x(k+1,k)} \quad - (2.19)$$

Substitution of the above two equations into the linear Kalman Filter, equations 2.10 to 2.15, yields the Extended Kalman Filter. Unfortunately, there is no longer a theoretical proof of the convergence, stability

and optimality of the estimates generated by the filter once this modification has been made. However, it can be stated that if the fundamental assumption and dual restrictions of observability and controllability are obeyed then the filter will provide sub-optimal estimates.

The Extended Kalman Filter is shown by Jazwinski (2.23) to be an extremely efficient estimator when applied to systems which can be categorised as mildly non-linear. In the event that the linearisation previously described became inappropriate in terms of accurately describing the system then the Extended Kalman Filter can cause severe discrepancies between the estimates generated and the true state. Such problems are commonly referred to as divergence (or bias) and the various techniques proposed to deal with such systems can be classified as either second-order filters or iterated filters.

The more complex of these two classes of filter is the second order filter, which takes into account second or higher order terms in the Taylor series expansion. Reported applications of the second order filter disagree as to the effectiveness of this modification. Athans (2.28), using a specific example, concludes that a second-order filter improves the prevention of divergence. Wishner et al. (2.30) reach the same conclusion when comparing three non-linear filters. However, Jazwinski (2.23) is much less

definite and states that although second order filters are more effective at removing bias, they do not solve the problem completely. More recently Alspach (2.24) and Tse (2.29) conclude that no real improvement is obtained and there is a greatly increased computational requirement. It is probably true that the effectiveness of second order filters is determined by the non-linearities present in the system model. In cases where bias is less likely, the increased computational burden may outweigh the improvement in the accuracy of the estimates.

The approach adopted by iterated filters is to relinearise at each stage about the new estimate obtained, thus producing a first iterated estimate. If the residual is not satisfactory after one iteration, another linearisation is performed. Wishner (2.30), Denham and Pines (2.31), Leung and Padmanabhan (2.32) and Jazwinski (2.23) are all in agreement as to the improved performance obtained by iterated filters. In general they have been found to be more effective in preventing divergence than second order or Extended Kalman Filters.

Recently, several attempts have been made to clarify the individual requirements of the various methods proposed for solving the non-linear estimation problem. Particular attention has been focussed on determining when a truly optimal filter is worth while or necessary and under what conditions is some version of the modified Kalman Filter adequate. These attempts

have been made by several authors (2.24, 2.33, 2.34 and 2.35). The conclusions reached are vague in that the only recommendation made is that the choice of filter is very much system dependent. For systems with only slight non-linearities the extra computational requirements of the more complicated methods may be unjustified in terms of the degree of increased accuracy obtained, whereas for complex non-linear systems the simpler methods may break down completely.

2.2.5. APPLICATIONS

The first reported application of the Kalman Filter was in 1962 by the aerospace industry. Smith, Schmidt et al. (2.36, 2.37, 2.38) applied the filtering theory developed by Kalman to the field of celestial navigation. Since then the theory has been applied in many fields as diverse as Agricultural Pest Control (2.39), Traffic Surveillance and Control (2.40) and Stock Market Forecasting (2.41).

The Kalman Filter was not applied to estimation problems in the chemical industry until 1968. The major reason for this delay is that chemical processes yield complicated dynamic models which are usually non-linear, have distributed parameters and are of relatively high order. Thus, it is difficult for researchers to derive accurate mathematical models which can be linearised whilst still obeying the restricting assumptions made in the derivation of the Extended Kalman Filter. To overcome this problem early

research was centered around the use of simulated experiments, in which identical mathematical models were used in both the filter and the simulation procedure used to obtain the "pseudo-measurements". Such experiments, although they do not always present a true picture of the difficulties encountered in on-line applications, provide solutions to many practical and theoretical problems. The success achieved in these off-line applications, together with the fact that the Kalman Filter algorithm can be implemented on a relatively small computer, was enough proof that successful on-line applications could be made.

The applications reviewed can be divided by authors as follows:

Coggan, Noton and co-workers

One of the most notable of the earlier chemical engineering applications of the Kalman Filter was by Noton and Choquette (2.42, 2.43) in the identification of a reactor train for The Polymer Corporation. The initial work was done using an off-line computer applied to open loop control experiments. Following the success of these experiments further work was done during closed loop experiments using a remote time-sharing computer. Although this work cannot be described as an on-line application, the overall exercise was successful in as much as a significant improvement over manual control was reported.

The theoretical problems encountered when dealing with systems containing large numbers of state variables,

unknown parameters, transport lags and strong non-linearities was tackled with considerable success by Coggan and Noton (2.17). In this paper the Extended Kalman Filter is applied to simulated systems exhibiting large random disturbances and having unknown initial values. The possibility of trading model simplification for unnecessary numerical accuracy is also discussed.

An approach to model reduction is suggested by Coggan and Wilson (2.45), who minimise the number of state variables required to describe a system. The Extended Kalman Filter is modified to evaluate measurement error statistics and detect the presence of bias. The same authors also report the on-line implementation of this filter (2.46). The minicomputer used had 4K words of memory and was applied to a tenth order system.

Seinfeld and co-workers

Seinfeld and co-workers have made major practical and theoretical contributions in the field of state and parameter estimation using the Kalman Filter. Attempts have also been made to define theoretically an analysis of the errors of the estimates generated by the filter algorithm.

Seinfeld (2.47) has extended the Kalman Filter to stochastic systems described by non-linear parabolic and hyperbolic partial differential equations. However, the convergence of this type of filter is somewhat suspect and the computational requirements of the

on-line implementation of such a system are prohibitive. Gavalas and Seinfeld (2.48) reduce a plug flow catalytic reactor problem to a lumped-parameter system and successfully estimate state variables. The equivalent distributed-parameter system is filtered by Seinfeld et al. (2.49), but problems of convergence and observability remain unsolved for such systems. A study of the incorporation of these techniques into a distributed-parameter control problem is carried out by Yu and Seinfeld (2.50). This work utilizes a simulated scalar parabolic system. More recently Ajinkya, Ray, Yu and Seinfeld (2.51) report the development of an approximate non-linear filter which is applied to systems described by coupled ordinary and partial differential equations. Results are quoted for the application of this filter to an ingot heating problem.

Sargent and co-workers

Goldman and Sargent (2.52, 2.53) report a feasibility study carried out on the possible on-line use of the Kalman Filter for state and parameter estimation in chemical engineering systems. The extended filter algorithm is derived for disturbance free processes and applied to the estimation of a simulated distillation column and a fixed bed catalytic reactor. The measurements made available by the simulation procedure are corrupted by superimposed Gaussian or rectangular noise. The filter is shown to

be robust in estimating bias and drift. It was also reported that convergence of the filter from initial estimates, $x(0,-1)$, is accelerated by selecting a high initial estimation error covariance matrix, $P(0,-1)$. The general conclusions reached in the choice of the initial value of $P(0,-1)$ have since been verified by other workers, see for example Yoshimura and Soeda (2.55). However, the assumption of a disturbance free process severely limits the possible applications of the filter used in this work and might even be viewed as a factor which would devalue the validity of the results obtained.

An extension of the non-linear catalytic reactor study as part of an optimal control scheme is reported by Joffe and Sargent (2.54). The non-linear distributed parameter stochastic system is decomposed into a non-linear lumped parameter model. Both the process and control scheme are simulated and found to be insensitive to the statistical assumptions, initial estimates and process noise covariance of the filter algorithm.

Recently, Fortescue and Kershenbaum (2.56) report the on-line application of a Kalman Filter to the estimation of a pilot plant scale carbon dioxide absorption-desorption unit. The computer used in this work was a Honeywell series 16 machine. Steady state experiments were carried out to determine the absorption parameters of a distributed parameter model from plant measurements. The results obtained display good

convergence and the problems of non-linearity are essentially overcome.

Wells and co-workers

Wells (2.56) demonstrated the use of the Extended Kalman Filter in estimating state variables and parameters in a highly non-linear simulated chemical process. The study of computation times made, indicated that for a physical system of similar dimension the Kalman Filter could be implemented in real time. An important concept discussed in this paper, is the analogy between the process noise covariance matrix, Q , and process uncertainty. Wells states that the magnitude of the Q matrix can be increased for systems that have dynamics that are not well understood or for state equations that represent a model simplification, thus placing more weight upon the measurements. From this he concludes that an exact description of the process dynamics of a system is not necessary to achieve good estimates. This would appear to be highly beneficial to chemical engineering problems as high numerical accuracy is not required by such systems and so model simplifications and approximations can be made with corresponding adjustments to the Q matrix. Unfortunately, this conclusion has been shown to be without foundation by Jazwinski (2.23) and Coleby (2.58). The reason given by these workers is that if the Q matrix is increased in magnitude it is likely that the filter will rely on the noisy and perhaps

biased measurements made.

Mehra and Wells (2.57) and Wells and Wismer (2.59) report on the dynamic modelling and on-line carbon estimation of a basic oxygen furnace used for steel making. Results are quoted for the on-line implementation of both a linear Kalman Filter and a second order non-linear filter. The authors claim an increase in accuracy of the estimates of between 30 and 50% when compared to other forms of state estimation.

Soliman and co-workers

Soliman, Ray and Szekely (2.60) report on the application of an Extended Kalman Filter to the problem of state estimation in a stainless steel manufacturing process. Good estimates of the molten metal composition were obtained even when only a limited number of temperature measurements were made available. In this work the Extended Kalman Filter was shown to be capable of producing reliable estimates in the presence of realistically high process and measurement noise. The authors conclude that the filter used would be applicable to an on-line situation and state that work is in progress in this direction.

Ramirez and co-worker

Lynch and Ramirez (2.61) present the development and real-time implementation of a time-optimal control algorithm for a continuous stirred-tank reactor. Estimation and filtering are carried out by an

Extended Kalman Filter. The reaction studied is the decomposition of hydrogen peroxide with a homogeneous catalyst of potassium iodide. The experimental work was done in an on-line environment using a General Data Corporation Nova 1210 Minicomputer with 8K words of memory. Lynch and Ramirez conclude that the Kalman Filter works very well in estimating the unmeasured system states and filtering the measurement noise in the system. However, the authors do not implement the complete filter algorithm in an on-line situation. The filter gain matrix, K , is assumed to be constant for a system at or near to steady state and is computed off line from a steady state value of the estimation error covariance matrix, $P(k,k)$. These assumptions are only valid whilst the system under study remains at or near to a steady state.

Fisher, Seborg and co-workers

Perhaps the most complete study of modern control theory, from an experimental point of view, was carried out during the years of 1972 to 1974 by Seborg, Fisher and co-workers in the Department of Chemical Engineering at the University of Alberta, Canada. The objective of this research is to examine promising modern control techniques and with due consideration to the practical constraints of the equipment available, to evaluate these techniques by on-line real time application to pilot plant units. The research carried out was

specifically concerned with the implementation of computer control to a pilot plant size, double effect evaporator and the subsequent evaluation of multi-variable design, analysis and control techniques. During this study the evaporators electronic instrumentation was interfaced to an IBM 1800 computer allowing the computer programmer access to 50 process variables.

The research done can be conveniently divided as follows:

- (1) Modelling and dynamic simulation of the evaporator.
- (2) Conventional control.
- (3) Multivariable feedback control.
- (4) Multivariable servo control.
- (5) Other multivariable control techniques.
- (6) On-line estimation and filtering.

All of the reports made during this study are published collectively by Fisher and Seborg (2.62).

The publications of major concern to this review are those which report on the on-line implementation of the Kalman Filter. The first of these, by Hamilton, Seborg and Fisher (2.63), is a report on the application of the stationary form of the discrete Kalman to state estimation in noisy processes. The effectiveness of the filter was demonstrated by both simulated and experimental tests. The authors state that the incorporation of the filter into a multi-

variable computer control system resulted in control being significantly better than when the Kalman Filter was omitted or replaced by a conventional exponential filter. This study was undertaken in the presence of process and/or measurement noise levels of 10% and it is interesting to note that the authors conclude that although the filter estimates were sensitive to unmeasured process noise, this effect could be reduced by treating the noise covariance matrices R and Q as design parameters rather than statistics. This would appear to indicate that the on-line application of the Kalman Filter involves a certain amount of 'tuning'.

In the later publication by Seborg, Fisher and Hamilton (2.64) several Kalman Filters and Luenberger observers are applied to a computer-controlled evaporator and the results compared. In conclusion it is stated that although the Luenberger Observer performed well under normal conditions, it was quite sensitive to process noise and unmeasured process disturbances. This difficulty was also encountered in the application of the Kalman Filter, but as described earlier, this was overcome by the variation of the R and Q matrices.

Payne and Coleby

The first experimental studies of the on-line implementation of non-stationary forms of the Kalman Filter are reported in publications by Payne (2.65)

and Coleby (2.58). Both of these works studied the on-line, real-time state estimation of an essentially uncontrolled double effect evaporator, whose instrumentation was interfaced to a Honeywell 316 computer with 16K words of memory.

The earlier work of Payne was involved in the application of the extended form of the Kalman Filter. In order to bring about a meaningful real-time implementation, Payne considerably reduced the order of the system to yield a fourth order model and fed measured inputs directly into his transition matrix, $\phi(k+1,k)$. Also included in this publication are the results of an off-line study into the effects of varying the value of the Q matrix when applying the Kalman Filter to state and parameter estimation. The results obtained show that the filter performance is improved by separating the diagonal elements of Q such that confidence in the model is reflected by a small constant on the diagonal elements corresponding to the measured state variables and a large constant on the elements corresponding to parameters and unmeasured variables. The work of Payne highlighted the many difficulties that are encountered during on-line implementation, many of which have been overlooked by workers carrying out research using simulated studies. Two of the most notable points are the cycle time required by the filter algorithm for large systems and the assumptions required in order to make the system operate within the limits imposed by the equipment used.

Following the work of Payne, the problem of a poor system model was tackled by Coleby. A solution to this problem was attempted by two separate approaches. The first of these, which is not reviewed here, is to improve the mathematical model describing the evaporator system. This modelling exercise resulted in a 7th order model of the system. The second approach was to modify the filter algorithm into which the improved model would be included. Coleby states that in cases where there is doubt about the accuracy of one or more of the prediction equations, some method to control filter divergence should be incorporated within the filter algorithm. The results of an off-line survey of the various techniques proposed as solutions to this problem shows that the modification proposed by Coggan and Wilson (2.45) was the most appropriate (a more detailed discussion of such techniques is included in section 2.2.6). Following these off-line studies Coleby implemented the theoretical developments made into an on-line filtering package. The results obtained from this experimentation indicate that although the modifications to the filter algorithm were successful, the mathematical model used was still rather questionable. This led to the net results obtained still falling short of what would be considered as a satisfactory level of accuracy.

2.2.6 DIFFICULTIES ENCOUNTERED IN APPLICATIONS OF THE KALMAN FILTER

A survey of the reported applications of the Kalman Filter reveals that the difficulties encountered fall into two broad categories.

- (i) Computational considerations.
- (ii) Divergence or bias problems.

2.2.6.1. COMPUTATIONAL CONSIDERATIONS

During the early simulation experiments carried out using the Kalman Filter investigators encountered difficulties because of the requirement of high precision in the filter computations and specifically in the computation of the estimation error covariance matrix, $P(k,k)$; see equation 2.14. This matrix is susceptible to round off errors because its elements decrease in magnitude after many sets of data have been processed. These errors can cause the diagonal elements of $P(k,k)$, which represent variances, to become negative and hence meaningless. Sorenson (2.16) proposed an alternative form for the equation used to compute $P(k,k)$, see equation 2.15, and as was stated in section 2.2.3, Aoki (2.18) showed this alternative form to be preferable.

A further problem concerning computational accuracy, as well as computation time and storage, arises when a large number of measurement sources are available. The problem is caused by the need to invert $m*m$ dimensional matrix, where m is the number of

measurements. This inversion is necessary to compute the filter gain matrix, $K(k)$, as shown by equation 2.12. One method proposed, and successfully demonstrated, to overcome this problem, in cases where the measurements are uncorrelated has been presented by Singer and Sea (2.66) and Sorenson (2.16). The method involves the processing of measurements individually and circumvents the requirement for an inversion. In practical applications of the Kalman Filter, Singer and Sea report a 30% reduction in the computer time required for the execution of the filter algorithm.

2.2.6.2. DIVERGENCE PROBLEMS

The most disturbing problem encountered during applications of the Kalman Filter is that of divergence away from the true state of the estimates produced. Three possible causes of this problem have been identified by Alspach (2.24) as;

- (i) Excessive non-linearities within the system.
- (ii) Incorrect system statistics.
- (iii) Construction of the filter on the basis of an erroneous mathematical model.

The possibility of excessive non-linearities within the system has been considered quite extensively in section 2.2.4. However, one further point remains

to be discussed, this being the computation of the predicted values of the state variables. It has been shown by many authors, see for example Coggan and Noton (2.17), Goldman (2.53) and Wells (2.56), that it is not practical to attempt the computation of the predicted state using equation 2.10. This is because the state transition matrix is not accurate enough to give good results unless the prediction interval is very small. Clearly a very small prediction interval is not practical in most cases because of the time required to execute the filter algorithm. To overcome this problem most investigators have used one of the many available numerical integration techniques; the technique used normally being determined by the type of system under consideration.

The problem of incorrect system statistics can be subdivided into two main areas, the incorrect choice of the initial conditions and their error covariances, i.e. $x(0,-1)$ and $P(0,-1)$, and the incorrect choice of the system input and measurement noise covariances, $Q(k)$ and $P(k)$. Research into the effects of using incorrect initial conditions (2.52, 2.53, 2.67, 2.68) has shown that providing the system model is correct, divergence will only occur if there is any inconsistency in the choice of the error covariance of the initial state vector, $P(0,-1)$. That is to say, if the state

vector, $x(0,-1)$ is known to be suspect, this must be reflected in the choice of $P(0,-1)$, i.e. the diagonal elements of $P(0,-1)$ must be larger than when $x(0,-1)$ is known to be correct. The general conclusion of the quoted works is that provided the initial values allocated to $P(0,-1)$ are large enough the filter will converge to the true state. One disadvantage of this approach is that the larger $P(0,-1)$ the longer it takes the filter to converge. However, in such cases some authors, see for example (2.17), have found that where knowledge of the system considered indicates some cross-correlation between initial estimation errors the inclusion of the relevant off-diagonal elements in $P(0,-1)$ helps to promote the convergence of the filter.

The difficulties encountered when an incorrect choice of measurement and process noise statistics is made seems to be rather more complex than the choice of initial conditions. This would seem to be inevitable as neither $Q(k)$ nor $R(k)$ is updated by the filter algorithm. The reported applications of the Kalman Filter, see for example (2.63, 2.65) have stated that filter performance is improved by treating these matrices as design parameters rather than system statistics. Coleby (2.58) agrees with this statement as far as the process noise covariance matrix, $Q(k)$, is concerned, as it is rarely determinable for

on-line chemical engineering applications. However, he differs in his treatment of the measurement noise covariance, $R(k)$, as the statistics should be readily evaluated for most types of instrument used in on-line applications. Thus, the problem would appear to reduce to one of determining $Q(k)$.

Divergence may occur in cases where the values of $Q(k)$ are too small, as the computed value of $P(k/k-1)$ see equation 2.11, will be falsely low thus causing the value of $K(k)$ to be so small that insufficient weight is placed upon new measurement data. The difficulties encountered are made more acute by the fact that if one overcompensates and makes $Q(k)$ too large, the filter is 'loosened' and the estimates produced become noisy as well as it taking longer for the filter to converge. An interesting approach to this problem is made by Mehra (2.69) and Graupe and Krause (2.70) who compute the matrices Q and R from measurement data prior to filtering. Unfortunately the only practical results given are for single output systems and immense problems are envisaged by Graupe and Krause in extending this method to multi-output systems. Other approaches have been suggested but the problem remains unsolved theoretically as far as the practical implementation of the filter is concerned. The method used to determine the value of Q for on-line applications remains one of trial and error.

Accurate modelling of chemical processes, particularly in the dynamic state, is notoriously difficult. This problem is accentuated in on-line applications of the Kalman Filter because of the type of mathematical model required. Thus, it is extremely unlikely that an accurate model of the system to be estimated will be available. The presence within the filter of a poor mathematical representation of the system is probably the most common cause, not to mention the most difficult to deal with, of filter divergence in the estimation of chemical processes. Construction of the filter on the basis of an erroneous model will cause divergence due to the filter "learning the wrong state too well" when it operates over many observations. The problem is particularly acute when the noise inputs to the system and/or the measurement noise are very low, for, as was discussed earlier, this will cause the filter to converge rapidly and take little notice of subsequent measurement data. Perhaps the most ironical factor which has to be taken into account when deciding upon a strategy for the solution of this problem is that in the original derivation of the linear filter Kalman assumes that the system dynamics are completely known and are precisely modelled. This will never be true in practice and therefore the usefulness of more detailed models of the system considered is questionable.

The first attempts at a solution to the problem of filter divergence caused by a poor model were centred around the manipulation of the matrix of input noise covariances, Q . Researchers into this problem, see for example Schlee et al. (2.71) and Jazwinski (2.23), found that inaccuracies in the mathematical model could be compensated for by increasing the value of the Q matrix. This procedure amounts to the addition of constant "fictitious noise" to the system which prevents the convergence of the error covariances of the predictions and thus forces the filter to take into account the measurements made at every sampling time. This adjustment to the Q matrix is justified by the fact that the difference between the true and estimated state could be due to unknown disturbances upon the system inputs if the model used is correct. In a discussion of this technique Jazwinski concludes that this method will be adequate for some inaccurate systems. However, this method has two obvious disadvantages. Firstly, the increase in Q will be constant and as this adjustment is meant to compensate for modelling error, then the error itself must be assumed constant. An assumption which cannot be justified in all cases. The second disadvantage is that as the Q matrix has a larger value than is normal, then the filter will not completely converge and consequently the estimates produced will be somewhat noisy.

An alternative approach to the problems encountered when using poor models is suggested by, amongst others, Anderson (2.72) who proposes the overweighting of the more recent information. The necessary weighting is accomplished by an exponential time factor. This method makes use of the fact that if the model used is inaccurate then the model errors themselves degrade the value of information in the distant past. Anderson (2.72) and Jazwinski (2.23) have shown the effect of such filters during simulated experiments. The results obtained show that this modification enables the filter to reduce divergence but not completely remove it. The main problem encountered in using this method is that of determining the exponential time factor, i.e. the number of past measurements that are to be taken into account by the filter. It is felt by the author that this fact alone will severely limit the possible applications of such filters.

The principle outlined above is also utilised by Lee (2.73) who develops a "moving-average" filter. This method presumes a poor model to be applicable only over a limited time span and furthermore, projections outside of this time span lead to model errors. Unfortunately, the problem of determining the exponential time factor is even more difficult than for the overweighting type of filter.

The obvious disadvantages of the approaches discussed so far led to the development of a completely different method by Schmidt (2.44). This method involves the use of new parameters within the system to account for the uncertainty that exists within the model. These parameters are then estimated along with the remaining state variables describing the system. This approach has serious disadvantages when applied to systems which already have a large number of state variables, because the computational burden imposed by increasing the size of the state vector by the addition of these parameters may be intolerable. Another disadvantage of this approach is that the system may become unobservable as there may not be enough measurements available to carry out the filtering procedure. Improved methods for estimating these parameters, which at the same time reduce the computation load, are proposed by Friedland (2.74) and Lin (2.75). Friedland uses a technique of augmenting the state vector and Lin one of invariant imbedding. These techniques succeed in reducing the computational burden but fail to take into account the need to reduce rather than increase the degree of the state vector for filters applied to complex systems such as chemical processes.

In the earlier discussion on the extension of the Kalman Filter to cope with non-linear systems,

see section 2.2.4, it was stated that when the non-linearities of the system are such that a successful linearisation cannot be achieved, divergence problems may be severe. It is felt by some authors, see for example Coleby (2.58), that such problems are analogous to those of an initial poor model and that the method proposed by Leung and Padmanabhan (2.32) could be applied. This method involves smoothing the trajectory back one measurement and relinearising the model to obtain an improved prediction. If the residuals between the estimates and the measurements are not within the statistical limits of the system the step is repeated, using a recursive relationship, until convergence is obtained. This technique was found to be effective when applied to the estimation of concentration and temperature trajectories for a simulated CSTR system. Possible applications of this iterated filter are limited by the increased computational burden imposed by the iterative nature of the filter and also by the fact that if the non-linear model is itself a poor one, divergence can arise within the recursive algorithm itself.

The most recent and probably the most powerful technique developed to solve the divergence problem is the so-called "Adaptive Filter". This technique examines the residuals between the predictions and

the measurements so as to determine appropriate values for the process noise covariance matrix, $Q(k)$. In 1968 Jazwinski (2.76, 2.77) proposed one of the first Adaptive Filters. The filter developed provides feedback from the residuals in terms of system noise input levels. Onset of divergence causes the residuals to grow and the consequent increase in the value of the matrix $Q(k)$ degrades the prediction error covariance matrix, increases the filter gain, and thus causes the filter to overweight incoming measurement data. Due to the theoretical complexities involved in such an operation Jazwinski makes assumptions about the form of the matrix $Q(k)$, i.e.

$$Q(k) = q(k) \cdot I \quad - (2.16)$$

where $q(k)$ is a scalar adaptively calculated and I is the identity matrix. This assumption somewhat limits the applications of this filter because, by making the diagonal elements equal, Jazwinski was in fact assuming that noise levels are uniform throughout the system.

Sriyananda (2.78) makes greater use of the available system statistics for the control of divergence. He defines an innovation process IN as;

$$IN(k) = y(k) - M(k) \cdot x(k, k-1) \quad - (2.17)$$

which is a non-stationary gaussian white noise process with variance $(M(k). P(k,k-1). M(k)^T + R(k))$. He states that it would be expected that $(IN(k)^T IN(k))$ would be less than three times the trace of this variance $(M(k). P(k,k-1). M(k)^T + R(k))$ with more than 90% probability. If this is not the case then divergence can be suspected and the filter gain, $K(k)$, is frozen at its current value. This has the effect of increasing $P(k,k-1)$ because on updating it is in fact incremented by the factor $Q(k)$ after allowing for the effect of the transition matrix. This process continues until $(IN(k)^T IN(k))$ is less than the trace of $(M(k). P(k,k-1). M(k)^T + R(k))$, when the filter gain, $K(k)$, is again recursively updated. In the absence of process noise, i.e. $Q(k)$ is a null matrix, the filter gain, $K(k)$, would remain frozen and thus keep the error within bounds. This technique only attempts to prevent divergence increasing and does not try to remove the damage done.

A method for the detection and elimination of divergence which uses as a basis a similar method to that proposed by Sriyananda (2.79) is derived by Coggan and Wilson (2.45). The authors state that in the standard Extended Kalman Filter there is no feedback of statistical data from the process and consequently if the model is incorrect or the

assumed, statistical characteristics are wrong the accuracies of the predicted values will be quite different from those indicated by the prediction error covariance matrix, $P(k, k-1)$. To overcome this the matrix $S(k)$, which represents the expected covariances of the differences between the predicted and measured values of the measured variables, is defined as follows:

$$S(k) = (M(k) \cdot P(k, k-1) \cdot M(k)^T + R(k)) \quad - (2.18)$$

The corresponding covariance matrix determined from the data is,

$$Z(k+1) = a \cdot Z(k) + (1-a) \cdot z(k+1) \cdot z(k+1)^T \quad - (2.19)$$

where, $z(k+1) = M(k) \cdot x(k+1, k) - y(k+1) - b(k+1)$ - (2.20)

and, $b(k+1) = a \cdot b(k+1) + (1-a) \cdot (M(k) \cdot x(k+1, k) - y(k+1))$ - (2.21)

An exponential filter ($a = \exp(-h/\tau)$; $0 < h < 1$) is used in preference to an aggregated sample because $Z(k)$ and $b(k)$ are dynamic statistics. If the measurements have errors with zero mean, $b(k+1)$ gives the filtered mean error, and if an element of $b(k+1)$ is persistently positive or negative this implies biased predictions.

This development is incorporated into the Kalman Filter by the following procedure after

the relevant matrix computations have been carried out;

(i) Element Z_{ii} replaces element S_{ii} if
 $Z_{ii} > S_{ii}$

(ii) Compute the matrix $v(k)$ where

$$\begin{aligned} v(k) = & M(k)^T \cdot (S(k) - R(k)) \cdot M(k) \\ & - M(k)^T \cdot M(k) \cdot P(k, k-1) \cdot M(k)^T \cdot M(k) \\ & + P(k, k-1) \end{aligned} \quad - (2.22)$$

(iii) Element V_{ii} replaces element P_{ii} if
 $V_{ii} > P_{ii}$

(iv) Elements of the bias vector $b(k)$ are subtracted from corresponding elements of $x(k, k-1)$ to give bias free predictions of the measured variables. The results obtained by Coggan and Wilson clearly demonstrate the ability of this procedure to prevent and remove filter bias. The procedure does, however, have drawbacks in cases where parameter estimation is required as well as state estimation. In such cases the bias vector, $b(k)$, cannot be used to produce a bias free state as this deviation is the driving force required to drive the parameters to their true values. In addition to this, the method does not take into account deviations of the unmeasured variables, although correction of the measured variables will obviously go some way to limiting the divergence of the estimates of these variables.

A more recent development in the field of adaptive filtering is reported by Jazwinski (2.79) who proposes using a low frequency random forcing function to represent the model error. The filter tracks this random forcing function in addition to the system state and thus adapts to observable model or environmental variations. The model for the random forcing function is a polynomial with random, time-varying, coefficients which allows adaption to almost any model or system variation. Jazwinski models his system thus;

$$x(k+1,k) = \phi(k+1,k) \cdot x(k,k) + G(k) \cdot u(k)$$

- (2.23)

where $u(k)$ represents the low frequency random forcing function and $G(k)$ represents the matrix of noise correlation. The vector $u(k)$ is incorporated into an augmented state vector for estimation within the Kalman Filter.

The filter developed by Jazwinski is somewhat contradictory as, although it is adaptive in nature, a knowledge of the system being considered is required in order to make a prior specification of the matrix $G(k)$. A further disadvantage of this filter is the extra computational burden invoked. The Extended Kalman Filter consists of five recursive relationships whereas Jazwinski's Adaptive

Filter consists of 18 such relationships. Despite the partitioning of some of the relationships to discard the statistics of the forcing functions there is a considerable increase in the computation load.

Coleby (2.58) reports on the results of a simulation study of several of the filters discussed above and recommends that in cases where divergence problems are suspected or likely some method to control them should be incorporated into the filter algorithm. In conclusion Coleby states that Adaptive Filters are generally found to give the best performance and states that the method proposed by Coggan and Wilson is one of the most appropriate in terms of its computational requirements and accuracy of estimates produced.

Reported studies of Adaptive Filters, see for example (2.45, 2.58 and 2.79), have shown them to be more general and powerful in dealing with the problems of divergence due to poor mathematical models. Kilbride-Newman (2.80) emphasizes this but criticizes the filters discussed so far because they over estimate the error covariance matrix, $P(k,k)$. This is because most Adaptive Filters treat the input noise that is supposed to represent the model errors as a zero mean stochastic process and this need not be a true assumption. The Adaptive Filter

of Coggan and Wilson (2.45) attempts to alleviate this problem by removing bias, but, as was discussed earlier, this causes further problems in that only state-variable estimation is possible. To overcome this problem Kilbride-Newman (2.80) develops an Adaptive Filter along similar lines to that of Jazwinski (2.79), but uses an alternative method for the computation of the vector $u(k)$, and modifies the nature of $G(k)$ so that it becomes a noise incidence matrix which is itself adaptively estimated. Thus, no prior knowledge of the system is required except an inaccurate model and the system statistics. The method, which is discussed in greater detail in Chapter 3, uses an exponential filter on the mean and covariances of the residuals to estimate the mean, $u(k)$, and the covariances of the so-called fictitious inputs, $C(k)$. It is the fictitious inputs generated by the filter which compensate for the modelling errors. This is similar to the approach adopted by Coggan and Wilson (2.45) but has a number of advantages. Kilbride-Newman's filter can not only be used for parameter estimation, in addition to the usual state estimation, but also introduces the new concept of a model error compensation strategy. This is due to the fact that the mean of the fictitious disturbances, $u(k)$, is not updated at each time

interval and therefore the mean of the fictitious inputs is only conditioned on the error remaining due to the poor model. The model error compensation strategy effectively updates the system model after a fixed number of measurements and thus improves the performance of the filter. Furthermore the compensation function, although not necessarily representing any real variable, does indicate the source and type of model error that exists. There are, as is always the case, a number of drawbacks. First of all there are a number of parameters required by the filter which need to be specified a priori. Also, the estimates produced may be suspect for the first few samples as it takes some time for the model error compensation strategy to be optimised. Despite these drawbacks and the fact that the filter cycle time is increased by some 10%, the author feels that Kilbride-Newman's Adaptive Filter is the most promising development made so far in the field of recursive filtering as applied to chemical engineering processes.

2.3. HEAT EXCHANGER AND EVAPORATOR MODELLING

2.3.1. INTRODUCTION

The basic principles of operation of an evaporator are easily understood by anyone in the process industry. However, a detailed analysis of the dynamic behaviour of such systems yields a complex set of interacting flow, pressure and temperature variables. Thus, if one were to derive a comprehensive mathematical model for one of the many common forms of evaporator, the number of state variables used would be far too large for the model to be used in an on-line application of the Kalman Filter. It is also likely that the model developed would contain partial differential equations. The above restrictions make it clear that to enable the derivation of a suitable model certain guide lines need to be stated a priori. The criteria adopted in this research are as follows:

(i) The model shall contain only those state variables of dominant or special interest.

(ii) The model will consist of a set of first order ordinary differential equations suitable for use within the Kalman Filter.

(iii) The state variables of special interest are those directly concerned with the heat transfer dynamics of the evaporator studied.

This brief review of the literature appertaining to the modelling of evaporators has been carried out with the above criteria in mind and accordingly is limited to shell and tube type heat exchangers involving an isothermal condensing medium. In particular the dynamics of shell and tube evaporators, similar to the process under study, are considered.

2.3.2. HEAT EXCHANGER MODELLING

Heat transfer equipment, in its many diverse forms, is to be found as an integral part of virtually all chemical and industrial processes. This has led to a vast amount of research being carried out into the dynamic and steady state behaviour of heat exchangers. Due to the great diversity in heat exchange equipment, the numerous process applications and the fact that three modes of heat transfer are involved to varying degrees, the field of research into the behaviour of heat exchangers is both extremely broad and complex. The literature published is reviewed annually by authors such as Eckert et alia., see for example (2.81), who cite substantial numbers of papers dealing with this topic.

2.3.2.1. STEADY STATE ANALYSIS

During the last fifteen years a great deal of

consideration, both experimentally and theoretically, has been given to the problem of two phase, vapour liquid flow and most of the published works in this area have been aimed at providing correlations and predictions of various aspects of heat exchange equipment. An area which has received special attention in the past and in fact still does, is the problem of predicting individual heat transfer coefficients. The results given in the literature for computing heat transfer coefficients are usually rather complex due to the fact that they are normally given as functions of design geometry, flow dynamics, physical properties and position within the heat transfer equipment. Kern (2.82), McAdams (2.83) and Kreith (2.84) give a large number of examples of this type of work.

In 1899 the first of many publications on the study of evaporators appeared in the form of Kestners original patent application (2.85) for climbing film evaporators. The first attempt at providing an empirical formula for predicting heat transfer coefficients is reported by Brooks and Badger (2.86), who produce a correlation from their experimental results for a water/steam system. The results obtained using this correlation proved to be unrealistic, because of the centrally located thermocouples used in the experiments and the assumption that the evaporator can be divided into

a preboiling and a boiling zone.

Stroebe et al. (2.87) show that the assumption of a preboiling zone is not necessarily true and overcome this problem by feeding liquid in a boiling condition into the tubes. The correlation produced for predicting the heat transfer coefficient is of the form:

$$h_b = \frac{7.8 * 10^6 * V^{0.1}}{(N_{pr})^{0.3} * \sigma^2 * (\Delta T_L)^{0.13}} \quad - (2.24)$$

where, h_b is the film heat transfer coefficient for the boiling material in $\text{Btu.hr}^{-1} \text{ft}^{-2} \text{ } ^\circ\text{F}^{-1}$,

V is the specific volume of vapour in $\text{ft}^3 \text{lb}^{-1}$,

N_{pr} is the dimensionless Prandtl Number,

σ is the surface tension in dynes cm^{-1} ,

ΔT_n is the difference between the average tube wall temperature and the average liquid temperature in $^\circ\text{F}$. This correlation has the advantage of taking into account the physical properties of the liquid in the tubes of the evaporator.

The research of Gupta and Holland (2.89) shows that the omission of the liquid feed rate from Stroebe's correlation is highly significant and proposed an alternative form of correlation. The correlation produced by Gupta and Holland for

predicting heat transfer coefficients for water systems is of the form:

$$q/A = \psi \cdot M^{0.6} + 90.3 * (T_b - T_i) * C_p \quad - (2.25)$$

where, q/A equals the heat flux in $\text{Btu. hr}^{-1} \text{ } ^\circ\text{F}^{-1}$,

M is the liquid feed rate in lbs hr^{-1} ,

T_i is the temperature of the feed in $^\circ\text{F}$,

T_b is the boiling temperature in $^\circ\text{F}$,

C_p is the specific heat in $\text{Btu. lb}^{-1} \text{ } ^\circ\text{F}^{-1}$,

and, ψ is a graphically read function of the temperature driving force.

When the feed enters at its boiling point equation 2.25 reduces to:

$$q/A = \psi \cdot M^{0.6} \quad - (2.26)$$

$$\text{and thus, } U_b = \frac{\psi \cdot M^{0.6}}{\Delta T} \quad - (2.27)$$

The authors state that U_b , the overall boiling heat transfer coefficient ($\text{Btu. hr}^{-1} \cdot \text{ft}^{-2} \cdot ^\circ\text{F}^{-1}$), was predicted to an accuracy of 2.85% when using this method. The accuracy quoted is found by calculating the standard deviation. Gupta and Holland, despite the fact that they neglect the physical properties of the feed stream, claim that this correlation has been found to accurately predict heat transfer coefficients for other physical systems.

A more complex approach is adopted by both Dukler et al. (2.90) and Beveridge et al. (2.91) who report on the development of mathematical models of evaporation systems which are used within computer flowsheeting programs. Beveridge et al. divide the evaporator tubes into six distinct regions for heat transfer. Each region requires an accurate prediction of the film side heat transfer coefficient to be computed, and due to the varying nature of these regions, a different correlation is used for each calculation. The reason for using such a complex procedure is that earlier correlations are inadequate due to the following:

(i) The correlations are very much equipment and system dependent.

(ii) The correlations are not sufficiently accurate.

(iii) Some of the required parameters are difficult to precompute for design purposes. The main drawback with this model is that it is very difficult to determine the point of transition from one regime to the next.

In the development of a non-linear, distributed parameter, dynamic model for a simple single-tube heat exchanger, Heidemann (2.92) uses the same type of correlations for predicting the heat transfer coefficients as those determined in earlier experimentation by Gallatig (2.93). These

correlations are assumed by the authors to be linear functions of the arithmetic mean temperature driving force, the shellside vapour flowrate and the tube-side liquid flowrate. The coefficients used in these correlations are determined by multivariable regression analysis.

The main cause of inaccuracy in the majority of these correlations is their system and equipment dependence. As a result it would probably be advantageous to develop correlations based on one of the reported developments but using one's own experimental data to determine any unknown parameters which may be system dependent.

2.3.2.2. DYNAMIC ANALYSIS

Over the last thirty years a considerable amount of attention has been focussed on the study of the dynamic behaviour of heat exchangers. The research carried out falls into two main categories:

(i) The development of transfer functions for the system considered using experimental data obtained by using both frequency response and pulse testing methods.

(ii) An analytical approach based upon a knowledge of the physical laws and system geometry appertaining to the process considered.

The first of these two methods consists of

measuring the response to a known disturbance to the system and then fitting a transfer function to this output response. A comprehensive summary of these methods is presented by Buckley (2.94) who surveys many of the classical frequency-response methods used in the design and testing of heat exchanger control systems. One of the many applications of the frequency response technique is reported by Lees and Hougen (2.95) who use a cosine pulse function to evaluate the dynamics of a shell and tube heat exchanger. At present the successful applications of this technique have been limited to single input/single output systems and is therefore of little value in model development for complex processes. However, research is at present being carried out by Momen (2.105) into the use of a pseudo random binary sequence technique for calculating transfer functions from experimental data for quite complex multiple input/multiple output heat exchange equipment. If such techniques prove to be successful a highly significant advance will have been made which will allow the use of this technique in the development of models for complex heat transfer equipment.

Models developed using the physical laws and system geometry governing a particular heat exchanger have been used within many control systems and

control system design techniques. The majority of the literature shows that such models are similar in approach, differing mainly in the type of disturbance applied to the system. Within the accuracy of the assumptions made, the models developed are applicable for all types of input disturbance and include process non-linearities.

In general the models developed by analytical techniques based upon a knowledge of the system considered, contain a set of partial differential equations, with the outlet temperature as the dependent variable and time and position along the exchanger tubes being the independent variables. Early research in this field was severely limited due to the lack of accurate, high speed digital computers. This meant unrealistic assumptions had to be made, e.g. constant physical properties, constant heat transfer coefficients and the use of lumped parameters, in order to obtain any form of solution in a realistic time period.

A complicated distributed parameter model which includes the effects of the main flow streams and the heat capacities of the exchanger walls and tubes is developed by Stermole and Larson (2.96). The model produced is rather complex and requires the specification of heat transfer correlations. Two different models are obtained by assuming that the

heat transfer coefficients are functions of;

a) the average bulk temperature and flowrate,
and b) the flowrate only.

The resultant models are solved using Laplace transformations. The first of these models proved to be satisfactory but the second, as well as an attempt to simplify the model using linearisation by perturbation techniques, was found to be unsuccessful. Privett and Ferrell (2.97) report on the satisfactory results obtained from a similar model which appears to be an extension of the one produced by Sterrole and Larson. A further development of this type of model is provided by Stainthorp and Axon (2.98), who make allowances within their model for variable tube passes. In this research, the dynamics of a multipass heat exchanger is modelled as a combination of a number of single tube exchangers with each flow reversal chamber assumed to be a first order lag. The mathematical solution is either inverted to give a time-dependent solution or left as a transfer function from which frequency-response data may be obtained.

It has been argued by many researchers that the dynamic behaviour of heat exchangers can only be described accurately by distributed parameter models. Korol'kov (2.99) derives a comprehensive partial differential equation model and uses the results

obtained as a standard reference for comparison with predictions from both lumped parameter models and simplifications of the standard model. Basing his conclusions on the difference between the output responses of the 'simple' models and that of the standard model, Korol'kov states that all of the 'simple' models proved to be inadequate. Unfortunately, no experimental results are presented to give credibility to this work. However, Finlay and Smith (2.100) use a similar technique and verify their standard reference model by comparing it with experimental data. Finlay and Smith also state that in certain conditions the simplified models can be applied and show the results obtained to be of an acceptable accuracy.

In parallel with these attempts to show that lumped parameter models are not very accurate in describing the behaviour of a system, a number of researchers were exploring the possibilities of using some type of model reduction technique in order to convert the extremely complex and cumbersome sets of partial differential equations into a form more acceptable to heat exchanger control systems. These techniques usually involve some form of linearisation of the original model. Koppel (2.101)¹⁰⁰ attempts to remove some of the inaccuracies caused by linearisation of the partial differential equations

by substitution of a normalised variable for the temperature prior to linearisation and solution by Laplace transformation methods. He states that the advantage is that the normalised equation is no longer very dependent upon small variations in the ratio of heat exchange to heat capacity and hence heat transfer coefficient. By comparison with experimental data Koppel shows that this approach provides an improvement on previous techniques when the disturbances are small.

A different approach was adopted by Myers et al. (2.102) who assumed that knowing the initial and final temperature distributions in the axial direction of the heat exchanger, these distributions could be applied during the transient period. This assumption is then used to simplify the distributed parameter model. The justification for this assumption is rather questionable and it is probable that this technique is only applicable to systems with a large capacitance.

Using a set of Hermitian polynomials to approximate the dependence of temperature on axial position, Dorri (2.103) reduces a set of partial differential equations to a set of ordinary differential equations. The resulting equations are shown to be easily soluble using an analogue computer but no verification by comparison with experimental data is provided.

A theoretical and experimental study of a single tube condenser is reported by Schoenberg (2.104) who concludes that predicted responses from a mathematical model of this type of process were very sensitive to the heat transfer coefficients and the temperature drop across the tube and shell. In developing a mathematical model he prefers to approximate the fundamental dynamic phenomena rather than attempting a rigorous solution of the partial differential equations governing the condenser. This approach appears to be quite a sound one in view of the fact that most generalised techniques have been found to fail in one situation or another. In addition to this it is quite likely that experimental observations of a process will lead to sensible simplifications of the model developed. The model developed by Schoenberg (2.104) consists of a set of eight ordinary non-linear differential equations. The model is linearised for small perturbations and solved by using Laplace transformations. The results obtained from this model are shown to be accurate in the early stages of a transient period by comparing them with experimental data. During the later stages the prediction error increases to a greater extent and is attributed by the author to neglecting a time lag in the vapour-liquid interface.

2.3.3. DYNAMICS OF EVAPORATOR SYSTEMS

The majority of the research carried out into the dynamics of evaporator systems has been primarily concerned with the prediction of outlet concentration of a required product. Most researchers in this area have treated the heat transfer dynamics of the system being considered as secondary factors in their work due to the following reasons:

(i) The time constants associated with the dynamics of heat transfer processes are very much smaller than those associated with the dynamics of the concentration processes.

(ii) The pressure and hence the temperature in most evaporation equipment is controlled.

These reasons make valid the assumptions that the dynamics of the heat transfer processes can be ignored and that any change can be considered as a step change.

Mathematical models of evaporator systems reported in the literature have used both empirical and theoretical approaches. Johnson (2.106) presents a variety of empirical models of differing complexity and fits parameters with experimental data from his falling-film evaporator. Nisenfeld and Hoyle (2.107) consider simple empirical models for feed forward control and use two first-order lags and a time delay to dynamically represent a six-effect evaporation process.

The first theoretical derivation presented in the literature is due to Anderson et al. (2.108). This work reports on the derivation of lumped parameter, sixth order model for a single effect evaporator. By neglecting both vapour space and heat transfer dynamics the model is reduced to a third order system. A frequency response comparison between the model and the equipment used proved inconclusive. The original model developed in this work is similar in some respects to the one proposed earlier by Day (2.109).

Zavorka et al. (2.110) report on the development of a general model for a single-effect of a commercial sugar evaporator. This model was extended to a triple-effect system after simplification and includes non-linear relations for the heat transfer coefficients in terms of solution concentrations and liquid levels. Two drawbacks with this work are firstly the omission of a general heat balance and secondly the assumption that vapourisation is proportional to the heat transferred to the liquid. Burdett and Holland (2.111) derive a mathematical model for a seventeen-effect evaporator system used in a desalination process. The model, which contains 380 non-linear equations in 380 unknowns, is shown by a comparison with experimental results to be accurate over large time spans and for large process disturbances.

As was discussed in section 2.2.5., a great deal of work has been done in this area, as well as in the areas of state estimation and computer control, by workers from the Department of Chemical Engineering, University of Alberta, Canada - see (2.62). The initial publication of Ritter and Andre (2.112) presents a direct derivation of a five-equation model for a double effect evaporator. Following this work, Newell and Fisher (2.113) develop a generalised approach for the modelling of multiple-effect evaporators. The approach presented separates the development of dynamic equations from the specification of evaporator configuration and since it is modular, is convenient to use. Using this technique a tenth-order non-linear, dynamic model of a double-effect pilot plant evaporator is derived and then linearised to produce a fifth-order state-space model. The authors report that comparison with experimental open-loop responses shows the results produced by the model to be extremely satisfactory. Wilson et al. (2.114) continue this work with a report on a procedure for reducing the order of the model obtained. The approach used is an intuitive one in which high order models can be approximated by state space models by setting the derivatives of the first-order equations with small time constants equal to zero.

More recently Niemi and Koinstinen (2.115) developed a fifth order mathematical model for a single effect evaporator used for concentrating black liquor in the wood pulping industry. The model is verified by comparison of the results of a computer simulation with experimental data obtained using radioactive tracers. Following this comparison experimental observations are used to linearise and simplify the models where possible. The authors conclude that the results obtained from the final model show it to be capable of producing an acceptable response to solids concentration and could be extended to cover the steam feed input.

Two works which remain to be considered are those due to Payne (2.65) and Coleby (2.58) who consider the same evaporator as the one studied in this research. Payne derives a comprehensive 18th model of a double effect evaporator using an analytical technique based on the physical laws governing the system. This model is then reduced to a fourth order one by assuming that the vapour phase dynamics of the evaporator are controlled. As a consequence, it can be assumed that the derivatives of the equations describing temperatures related to the vapour space are zero. Coleby assumes that the vapour phase dynamics can be described by algebraic equations since the time constants associated

with the vapour phase dynamics are small compared to those relating to the heat transfer dynamics. In order to obtain a differential equation describing the rate of change of the liquid flow on the shell-side of the various evaporator units, Coleby differentiates an algebraic equation and then applies the chain rule. The resulting seventh order model required the introduction of a number of parameters before it gave satisfactory results and even then was liable to cause bias due to the assumption neglecting the vapour phase dynamics.

2.4. CHAPTER REVIEW

The discussion in the preceding sections of this chapter has been concerned with the application of process identification techniques to chemical engineering systems. In particular, a type of recursive state estimator known as the Kalman Filter has been discussed. Also, the mathematical modelling of heat exchange equipment with special reference to the heat transfer dynamics of evaporators has been considered. This discussion has revealed that the following areas require further exploration:

(i) The development, testing and comparison of strategies to ensure and promote the convergence of the estimates generated by the Kalman Filter. This study to be carried out in a simulated environment.

(ii) The application of established techniques to the mathematical modelling of the heat transfer dynamics of a pilot plant scale double effect evaporator. It is evident from the literature survey that the model which will describe the behaviour of the double effect evaporator most accurately will be the one obtained by combining the pertinent features of several of the modelling approaches already discussed and then reducing the degree of this model by the use of experimental observation.

(iii) The on-line application of the modified Kalman Filter to the study of the heat transfer

dynamics of a double effect evaporator. The literature survey has revealed that the on-line applications of the Kalman Filter reported so far have been mainly concerned with the use of simplified filters to track controlled rather than open loop systems.

With the exception of chapters five and six which describe the hardware and software used in on-line experimentation, the remainder of this thesis will follow the structure given above.

CHAPTER 3

THEORETICAL DEVELOPMENTS

3.1. INTRODUCTION

The most serious problem encountered in applying the Kalman Filter to non-linear processes is that of divergence of the estimates away from the true state. This problem, together with the solutions proposed in the literature, has already been discussed in some detail in section 2.2.6.2, where the three possible causes of divergence were identified as;

- (i) Excessive non-linearities.
- (ii) Incorrect system statistics.
- (iii) Use of a poor mathematical model.

When considering the effects of using incorrect system statistics a number of authors, see for example (3.1, 3.2, 3.3), recommend the approach of treating the relevant matrices as design parameters rather than as system statistics. This approach, hereafter referred to as 'Filter Tuning', has been shown by the literature to be a successful one despite the fact that it is no more than a process of trial and error. Indeed, one can go so far as to state that the success achieved when using this technique is such that a more sophisticated method to solve the problem of incorrect system statistics is unnecessary. The main disadvantage of this technique is that the Filter Tuning needs to be carried out with reference to the physical processes considered in order that

the value assigned to each statistic reflects the likely uncertainty of the particular variable(s) referred to.

The main aim of this chapter is to develop techniques to deal with the problems of excessive non-linearities and poor mathematical models. In section 3.2 the problem of dealing with poor mathematical models will be considered. First of all, the Adaptive Filter due to Kilbride-Newman (3.4) is discussed in detail and then modifications to enable its on-line application are proposed. The problem of excessive non-linearities is dealt with in section 3.3. A statement of this problem, as seen by the author, is made and then an entirely new technique for linearisation is developed and discussed.

3.2. ADAPTIVE FILTERING

In cases where divergence problems are suspected or likely some method to control them should be incorporated into the filter algorithm. In section 2.2.6.2. the application of Adaptive Filters to control divergence was discussed and it was found that they are more general and powerful in dealing with bias than other filters.

3.2.1. KILBRIDE-NEWMAN'S ADAPTIVE FILTER

Consider the true system equations to be represented by:

$$\dot{x}_s = f_s (x_s, u_s, z_s) \quad - (3.1)$$

and the model to be represented by,

$$\dot{x}_m = f_m (x_m, u_m, z_m) \quad - (3.2)$$

Equation 3.1 can also be written as,

$$\dot{x}_s = f_m (x_s, u_s, z_s) + f_s (x_s, u_s, z_s) - f_m (x_s, u_s, z_s) \quad - (3.3)$$

Define,

$$F_4 \cdot w = f_s (x_s, u_s, z_s) - f_m (x_s, u_s, z_s) \quad - (3.4)$$

where, F_4 is an $n \times 1$ matrix of the form $F_4 (i,j) = 0$.

except for at most, one element in each row and column which may be unity,

w is an 1×1 vector,

and, l is the dimensionality of the model errors.

It then follows that,

$$\dot{x}_S = f_m(x_S, u_S, z_S) + F_4.w \quad - (3.5)$$

Comparing equations 3.5 and 3.2 it can be seen that $f_m(x, u, z)$ can generate the true state if it is disturbed by the inputs $F_4.w$. These inputs are referred to as 'fictitious inputs' because they have no physical significance yet they provide the necessary corrections to compensate for modelling errors. In general $F_4.w$ will be unknown, but approximating this model compensation term by a Gaussian random process will at least make the errors incurred in using the inaccurate model random and unbiased.

The next step in the development of the Adaptive Filter is to find a way of calculating the mean (E_w), the covariance ($E(w-Ew).(w-Ew)^T$) of the fictitious inputs and the matrix F_4 for the random approximation to the model errors. A realisation of the above factors will be known as a 'Model Error Compensation' strategy. In developing this strategy the primary aim will be to provide a filter in which the estimation error covariance matrix is minimised while at the same time prevented from becoming over optimistic.

Considering $f_m(x, u, z)$ to be linear and

stationary and approximating $F_4.w$ by a random process,

$$\dot{x} = F_1 x + F_2 u + F_3 z + F_4 Ew \quad - (3.6)$$

where, F_1 is an $n*n$ matrix of coefficients,

F_2 is an $n*p$ matrix of coefficients,

F_3 is an $n*r$ matrix of coefficients,

p is the dimensionality of the vector of system inputs,

r is the dimensionality of the vector of process noise,

and, removing the subscripts m and s to simplify the terminology in the following developments.

At this point there are two possible routes which one could take. The first possibility is to estimate Ew , which is an unknown vector of parameters, by forming an augmented state vector. A consideration of the likely increase in dimensionality together with the subsequent increase in computation time and storage requirements leads to this approach being discarded. Indeed the problems associated with increased dimensionality are ones which should always be avoided, particularly when dealing with the complex non-linear systems frequently found in chemical engineering processes. The second possibility, which possesses no disadvantages as regards increased dimensionality, is to write out formally the filter equations for the augmented

system described above, but then only to take estimates of the true state vector, x , leaving the model error vector, Ew , constant. This technique follows that described by Schmidt (3.5) and Jazwinski (3.6) and although it takes no account of the variation in the value of Ew the error committed in not improving the estimate will be modelled so that its effect on the estimates of x will be taken into account.

The following derivation follows that of Kilbride-Newman (3.4) with the exception that the filter developed in this instance is applicable to discrete-time systems as apposed to the continuous-time systems dealt with in the original publication.

Consider a poorly modelled unforced dynamical system represented by the following differential equation,

$$\dot{x}(t) = F_1 \cdot x(t) + F_4 \cdot w(t) \quad - (3.7)$$

On integration this yields,

$$x(k+1) = \phi(k+1, k) \cdot x(k) + \Gamma(k+1, k) \cdot F_4 \cdot w(k) \quad - (3.8)$$

which can be rewritten as,

$$\begin{bmatrix} x(k+1) \\ w(k+1) \end{bmatrix} = \begin{bmatrix} \phi(k+1, k) & \Gamma(k+1, k) F_4 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x(k) \\ w(k) \end{bmatrix} \quad - (3.9)$$

If the formal equations of this augmented system are now considered, the prediction error covariance matrix will be,

$$P(k+1,k) = \begin{bmatrix} P_x(k+1,k) & P_{x,w}(k+1,k) \\ P_{x,w}(k+1,k)^T & P_w(k+1,k) \end{bmatrix} \quad - (3.10)$$

where, $P_x(k+1,k) = E(x(k+1) - x(k+1,k))(x(k+1) - x(k+1,k))^T$

$$P_{x,w}(k+1,k) = E(x(k+1) - x(k+1,k))(w(k+1) - Ew)^T$$

$$P_w(k+1,k) = E(w(k+1) - Ew)(w(k+1) - Ew)^T$$

$x(k+1)$ is the actual state vector at time t_{k+1}

and, $w(k+1)$ is the actual modelling error vector at time t_{k+1} .

Since the a-priori value of $w(k)$, Ew , is not improved by the filter then $P_w(k+1,k)$ remains constant and the relationship for calculating the prediction error covariance matrix becomes,

$$\begin{bmatrix} P_x(k+1,k) & P_{x,w}(k+1,k) \\ P_{x,w}(k+1,k)^T & P_w \end{bmatrix} = \begin{bmatrix} \phi(k+1,k) & \Gamma(k+1,k) \cdot F_4 \\ 0 & I \end{bmatrix}$$

$$\begin{bmatrix} P_x(k,k) & P_{x,w}(k,k) \\ P_{x,w}(k,k) & P_w \end{bmatrix} \begin{bmatrix} \phi(k+1,k) \\ 0 \\ \Gamma(k+1,k) \cdot F_4 \\ I \end{bmatrix}^T$$

Defining,

$$C(k+1,k) = P_{x,w}(k+1,k) \cdot (\Gamma(k+1,k) F_4)^T - (3.12)$$

$$C(k,k) = P_{x,w}(k,k) \cdot (\Gamma(k,k-1) F_4)^T - (3.13)$$

then the modified filter algorithm can be written down as the following set of prediction and estimation equations:

Prediction

$$x(k+1,k) = \phi(k+1,k) \cdot x(k) + \Gamma(k+1,k) \cdot F_4 \cdot E_w - (3.14)$$

$$P_x(k+1,k) = \phi(k+1,k) \cdot P_x(k,k) \cdot \phi(k+1,k)^T + \Gamma(k+1,k) \cdot F_4 \cdot P_w \cdot F_4^T \cdot \Gamma(k+1,k)^T + \phi(k+1,k) \cdot C(k,k) + C(k,k)^T \cdot \phi(k+1,k)^T - (3.15)$$

$$C(k+1,k) = \phi(k+1,k) \cdot C(k,k) + \Gamma(k+1,k) \cdot F_4 \cdot P_w \cdot F_4^T \cdot \Gamma(k+1,k)^T - (3.16)$$

Estimation

$$K(k+1) = P_x(k+1,k) \cdot M(k+1)^T \cdot (M(k+1) \cdot P_x(k+1,k) \cdot M(k+1)^T + R(k+1))^{-1} - (3.17)$$

$$x(k+1,k+1) = x(k+1,k) + K(k+1) \cdot (y(k+1) - M(k+1) \cdot x(k+1,k)) - (3.18)$$

$$P_x(k+1, k+1) = (I - K(k+1) \cdot M(k+1)) \cdot P_x(k+1, k).$$

$$(I - K(k+1) \cdot M(k+1))^T + K(k+1) \cdot R(k+1).$$

$$K(k+1)^T \quad - (3.19)$$

$$C(k+1, k+1) = (I - K(k+1) \cdot M(k+1)) \cdot C(k+1, k).$$

$$(I - K(k+1) \cdot M(k+1))^T + K(k+1) \cdot R(k+1).$$

$$K(k+1)^T \quad - (3.20)$$

The Adaptive Filter shown above, equations 3.14 to 3.20, will only perform satisfactorily if Ew and P_w define consistent stochastic processes. In other words, P_w must accurately represent the covariance matrix of the errors ($w(k) - Ew$). This means that an error in the value of Ew will deteriorate the accuracy of the state estimates by requiring P_w to increase, which in turn results in a larger estimation error covariance matrix, $P(k, k)$. Thus, in order to complete the derivation of the filter, some means of providing a consistency check on the value of P_w must be developed and, in addition, a method for updating the a-priori value of the model error vector, Ew , needs to be developed.

Firstly, a method for providing a consistency check on the value of P_w will be developed. If the values of Ew and P_w define consistent stochastic processes, then the residuals vector, $(y(k) - M(k) \cdot x(k, k-1))$, will be a zero mean white noise process

with,

$$\frac{1}{(t_{k+1}-t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot x(t)) (y(t) - M(t) \cdot x(t))^T \cdot dt$$

$$= M(k+1) \cdot P(k+1, k) \cdot M(k+1)^T + R(k+1)$$

- (3.21)

If it is now assumed that P_w can be represented by a diagonal matrix, $I \cdot c$, where c is a scalar, then defining $\tilde{c} = c - \bar{c}$, where \bar{c} is an a-priori guess for c , we obtain,

$$\frac{1}{(t_{k+1}-t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot x(t)) (y(t) - M(t) \cdot x(t))^T \cdot dt$$

$$= M(k+1) \cdot P(k+1, k) \cdot M(k+1)^T + R(k+1) + (M(k+1) \cdot \Gamma(k+1, k) \cdot F_4) \cdot (M(k+1) \cdot \Gamma(k+1, k) \cdot F_4)^T \cdot \tilde{c}$$

- (3.22)

Therefore,

$$\tilde{c} = \left(\frac{1}{(t_{k+1}-t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot x(t)) (y(t) - M(t) \cdot x(t))^T \cdot (y(t) - M(t) \cdot x(t)) \cdot dt \right.$$

$$\left. - \text{TRACE}(M(k+1) \cdot P(k+1, k) \cdot M(k+1)^T + R(k+1)) \right) / \left(\text{TRACE} \right.$$

$$\left. ((M(k+1) \cdot \Gamma(k+1, k) \cdot F_4) (M(k+1) \cdot \Gamma(k+1, k) \cdot F_4)^T) \right)$$

- (3.23)

where the TRACE of a matrix is defined as the sum of the diagonal elements. Equation 3.23 enables the required consistency check on P_w to be made and as a result \bar{c} is updated to $(\bar{c} + \tilde{c})$ unless this is negative, in which case $\bar{c} = 0$.

Equation 3.23 requires a value for,

$$\frac{1}{(t_{k+1} - t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot x(t))^T (y(t) - M(t) \cdot x(t)) \cdot dt$$

The required value is obtained by representing the above expression by $g(t_{k+1})$ and using an exponential filter as follows,

$$g(t_{k+1}) = g(t_k) + \alpha \left(\frac{1}{t_{k+1} - t_k} \right) \int_{t_k}^{t_{k+1}} (y(t) - M(t) \cdot x(t))^T \cdot (y(t) - M(t) \cdot x(t)) \cdot dt - g(t_k)$$

- (3.24)

where, $0 < \alpha < 1$.

Clearly, the above equation needs to be transformed into a discrete form, as it is unlikely that differential equations will be available to represent the functions $y(t)$ and $M(t)$. This is done by assuming a linear relationship for both $y(t)$ and $M(t)$ between the sampling points t_k and t_{k+1} . Thus, equation 3.22 becomes,

$$g(k+1) = g(k) + \alpha \left(\frac{1}{2} (y(k+1) - M(k+1).$$

$$\begin{aligned} & x(k+1, k))^T (y(k+1) - M(k+1). x(k+1, k)) + \frac{1}{2} ((y(k) - \\ & M(k). x(k, k-1))^T (y(k) - M(k). x(k, k-1))) \\ & - g(k) \end{aligned} \quad - (3.25)$$

Equation 3.25 is a recursive relation for the scalar quantity $g(k)$, which is an estimate of the trace of the covariance matrix of the residuals. This equation together with equation 3.23 can be implemented alongside the filter algorithm defined by equations 3.14 to 3.20. This will ensure that the statistics used in the filter are consistent with the statistics of the residuals.

The derivation of the filter has now reached the stage where the value found for P_w from equations 3.23 and 3.25 will be consistent with whatever value of E_w is used. However, in order to optimise the performance of the Adaptive Filter, some method is required to obtain an accurate estimate of E_w . From its definition, it can be seen that P_w will be minimised when E_w is known precisely and by examination of equation 3.23 it is clear that an accurate E_w will minimise the quantity $g(k)$ defined in equation 3.25. Therefore, the computation of a value for E_w is similar to optimal control problems designed to minimise an

objective function of the form,

$$J = \frac{1}{(t_{k+1} - t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot x(t))^T (y(t) - M(t) \cdot x(t)) \cdot dt$$

- (3.26)

Let \bar{w} be some a-prior prediction of Ew , then

$$\tilde{w} = Ew - \bar{w}.$$

If the prediction due to the mathematical model of the system considered is represented by $\bar{x}(k)$, i.e.

$$\bar{x}(k+1) = \phi(k+1, k) \cdot x(k, k)$$

then,

$$x(k+1, k) = \bar{x}(k+1) + \Gamma(k+1, k) \cdot F_4 \cdot \tilde{w}$$

- (3.27)

Substituting equation 3.27 into equation 3.26 and setting $\partial J / \partial w(k) = 0$ for all k gives,

$$\frac{1}{(t_{k+1} - t_k)} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot \bar{x}(t))^T \cdot dt \cdot M(k+1) \cdot \Gamma(k+1, k) \cdot F_4 = (M(k+1) \cdot \Gamma(k+1, k) \cdot F_4 \cdot w) \cdot M(k+1) \cdot \Gamma(k+1, k) \cdot F_4$$

- (3.28)

Rearranging this equation we obtain,

$$\begin{aligned} \tilde{w} = & \frac{1}{(t_{k+1} - t_k)} \left((M(k+1) \Gamma(k+1, k) \cdot F_4)^T \right. \\ & \left. (M(k+1) \Gamma(k+1, k) \cdot F_4)^{-1} \cdot (M(k+1) \right. \\ & \left. \Gamma(k+1, k) \cdot F_4)^T \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \right. \\ & \left. \bar{x}(t)) \cdot dt \right) \quad - (3.29) \end{aligned}$$

The value of \bar{w} can now be updated to $(\bar{w} + \tilde{w})$. Following a similar procedure to that used for checking the consistency of P_w , the quantity,

$$\frac{1}{t_{k+1} - t_k} \int_{t_k}^{t_{k+1}} E(y(t) - M(t) \cdot \bar{x}(t)) \cdot dt$$

is represented by $\gamma(t_{k+1})$ and using an exponential filter can be calculated using the following equation:

$$\begin{aligned} \gamma(t_{k+1}) = & \gamma(t_k) + \beta \left(\frac{1}{t_{k+1} - t_k} \int_{t_k}^{t_{k+1}} (y(t) - \right. \\ & \left. M(t) \cdot \bar{x}(t)) \cdot dt - \gamma(t_k) \right) \quad - (3.30) \end{aligned}$$

where, $0 < \beta < 1$.

Equation 3.30 can now be transformed into a discrete form by a similar procedure to the one used to obtain equation 3.25 from equation 3.24. The resulting equation is as follows:

$$\begin{aligned} \gamma(k+1) = & \gamma(k) + \beta(\frac{1}{2}(y(k+1)-M(k+1). x(k+1,k)) \\ & + \frac{1}{2}(y(k)-M(k). x(k,k-1)) - \gamma(k)) \end{aligned}$$

- (3.31)

Equation 3.31 is a recursive relationship for estimating the mean of the residuals. This equation together with equation 3.29 can be implemented alongside the filter algorithm, equations 3.14 to 3.20. This will provide a method for calculating an accurate estimate of Ew and so ensure that the filter performance is optimised.

The derivation of the adaptive filter can now be completed as follows:
Consider a noisy, forced dynamical system represented by the following set of differential equations,

$$\dot{x}(t) = F_1. x(t) + F_2. u(t) + F_3. z(t) + F_4. \bar{w}$$

- (3.32)

with measurements,

$$y(t) = M(t). x(t) + v(t).$$

- (3.33)

Integration of equation 3.32 gives,

$$\begin{aligned} x(k+1) = & \phi(k+1,k). x(k) + \Gamma(k+1,k).F_2.u(k) \\ & + \Gamma(k+1,k).F_3. z(k) + F_4. \bar{w} \end{aligned}$$

- (3.34)

If the matrices $Q(k)$ and $R(k)$ are as defined in section 2.2.3, the complete adaptive filter will consist of the following prediction/estimation relationships.

Prediction

$$\begin{aligned} x(k+1,k) = & \phi(k+1,k) \cdot x(k,k) + \Gamma(k+1,k) \cdot F_2 \cdot u(k) \\ & + \Gamma(k+1,k) \cdot F_3 \cdot z(k) + F_4 \cdot \bar{w} \end{aligned} \quad - (3.35)$$

$$\begin{aligned} P(k+1,k) = & \phi(k+1,k) \cdot P(k,k) \cdot \phi(k+1,k)^T \\ & + \Gamma(k+1,k) \cdot F_3 \cdot Q(k+1) \cdot F_3^T \cdot \Gamma(k+1,k)^T \\ & + \phi(k+1,k) \cdot C(k,k) + C(k,k)^T \cdot \phi(k+1,k)^T \\ & + (\Gamma(k+1,k) \cdot F_4) (\Gamma(k+1,k) \cdot F_4)^T \cdot \bar{c} \end{aligned} \quad - (3.36)$$

$$\begin{aligned} C(k+1,k) = & \phi(k+1,k) \cdot C(k,k) + \Gamma(k+1,k) \cdot F_4 \cdot \\ & \cdot P_w \cdot F_4^T \cdot \Gamma(k+1,k)^T \end{aligned} \quad - (3.37)$$

Take measurements,

$$y(k) = M(k) \cdot x(k) + v(k)$$

$$\begin{aligned} \gamma(k+1) = & \gamma(k) + \beta(\frac{1}{2}(y(k+1) - M(k+1) \cdot x(k+1,k)) + \frac{1}{2}(y(k) - \\ & M(k) \cdot x(k,k-1)) - \gamma(k)) \end{aligned} \quad - (3.38)$$

$$\begin{aligned} \tilde{w} &= ((M(k+1). \Gamma(k+1, k). F_4)^T. (M(k+1). \\ &\Gamma(k+1, k). F_4))^{-1}. (M(k+1). \Gamma(k+1, k). F_4)^T. \\ &\gamma(k+1) \end{aligned} \quad - (3.39)$$

$$\bar{w} = \bar{w} + \tilde{w} \quad - (3.40)$$

$$\begin{aligned} g(k+1) &= g(k) + \alpha(\frac{1}{2}((y(k+1) - M(k+1). x(k+1, k))^T. \\ &(y(k+1) - M(k+1). x(k+1, k))) + \frac{1}{2} ((y(k) - \\ &M(k). x(k, k-1))^T. (y(k) - M(k). \\ &x(k, k-1))) \end{aligned} \quad - (3.41)$$

$$\begin{aligned} \tilde{c}_{k+1} &= \frac{g(k+1) - \text{TRACE} (M(k+1). P(k+1, k). M(k+1))^T}{\text{TRACE} ((M(k+1). \Gamma(k+1, k). F_4)(M(k+1). \\ &+ R(k+1)))} \\ &\Gamma(k+1, k). F_4)^T) \end{aligned} \quad - (3.42)$$

$$\begin{aligned} \bar{c}_{k+1} &= \bar{c}_k + \tilde{c}_{k+1} & \bar{c}_{k+1} &> 0 \\ &= 0 & \bar{c}_{k+1} &< 0 \end{aligned} \quad - (3.43)$$

$$P_w = I. \bar{c}_{k+1} \quad - (3.44)$$

Estimation

$$\begin{aligned} K(k+1) &= P(k+1, k). M(k+1)^T. (M(k+1). P(k+1, k). \\ &M(k+1)^T + R(k+1))^{-1} \end{aligned} \quad - (3.45)$$

$$\begin{aligned}
 x(k+1, k+1) &= x(k+1, k) + K(k+1) \cdot (y(k+1) - \\
 &M(k+1) \cdot x(k+1, k)) \quad - (3.46)
 \end{aligned}$$

$$\begin{aligned}
 P(k+1, k+1) &= (I - K(k+1) \cdot M(k+1)) \cdot P(k+1, k) \cdot \\
 &(I - K(k+1) \cdot M(k+1))^T + K(k+1) \cdot R(k+1) \cdot \\
 &K(k+1)^T \quad - (3.47)
 \end{aligned}$$

$$\begin{aligned}
 C(k+1, k+1) &= (I - K(k+1) \cdot M(k+1)) \cdot C(k+1, k) \cdot \\
 &(I - K(k+1) \cdot M(k+1))^T + K(k+1) \cdot R(k+1) \cdot \\
 &K(k+1)^T \quad - (3.48)
 \end{aligned}$$

The Adaptive Filter developed above, equations 3.35 to 3.48, can now be used in exactly the same recursive manner as the Extended Kalman Filter.

3.2.2. IMPLEMENTATION OF THE ADAPTIVE FILTER

The Adaptive Filter derived in the previous section is not as yet ready to be applied in either an on-line or simulated environment because the parameters α and β and the matrix F_4 are unknown. The problem of parameter estimation also requires consideration and, as will be seen later, the associated parameter, θ , needs to be determined.

3.2.2.1. THE F_4 MATRIX

The F_4 Matrix will, in general, be unknown, although in situations where deliberate modelling errors have been committed some information will be available about the nature of F_4 . Such a situation would occur if the true system equations are known but considered too complicated or involve too many state variables for on-line filtering. The use of a simplified model in such cases will mean there exists a knowledge of the modelling error committed.

In the majority of situations all that is known about F_4 is that $F_4(i,j) = 0$ except for, at most, one element in each row and column, which may be unity. Having stated this, it should be pointed out that there is little theoretical evidence that such a structure is correct. However, the constraints placed upon the structure of F_4 does ease its computation, and as will be discussed in chapter 4, it turns out that such a structure is likely to be near to the truth. In order that the locations of these non-zero elements may be determined, the following procedure is adopted. The dimension of the model error compensation term, l , is taken to be unity

and thus F_4 is reduced to an n -dimensional vector. Each unit vector of this form is tried in the filter and the vector that minimises $g(k)$ is selected as the first column of F_4 . The dimension of F_4 is then increased to 2 and the same process used to determine the second column of F_4 . This process is continued until $l = M$, the dimension of the measurement vector. This is the largest number of independent variables that can be computed via equation 3.39 and therefore restricts this method to finding model error compensation vectors with dimension less than or equal to M . Whilst the dimensionality of F_4 is being varied care is needed to ensure that the correct dimensions are used when inverting the following matrix,

$$(M(k+1) \quad \Gamma(k+1,k) \cdot F_4)^T (M(k+1) \quad \Gamma(k+1,k) \cdot F_4),$$

see equation 3.39. When F_4 is a $n \times 1$ vector this matrix will be a 1×1 ; when F_4 is an $n \times m$ matrix the dimensions will be $m \times m$ and etc.

When considering a possible application of the Adaptive Filter, particularly in cases where the dimensionalities of the state and measurement vectors are large, it soon becomes clear that the

determination of the F_4 matrix is the most serious drawback of this filter. For example, if $n = 8$ and $m = 5$, then a total of 30 trial runs would need to be carried out in order to determine F_4 . Clearly this would not be possible in an on-line situation and a way of reducing the number of trials needs to be found. No theoretical technique is available to do this but in most applications an examination of the mathematical model will lead to a considerable reduction in the number of trials required. Since F_4 relates the state vector to the model error vector, which is in turn determined by a calculation involving the vector of residuals, see equation 3.39, then errors can only be determined for state variables which occur in those differential equations that describe the behaviour of measured state variables, e.g. if the differential equation describing x_1 is,

$$\dot{x}_1 = (x_7 - x_1) / \tau_{\eta 1}$$

where $\tau_{\eta 1}$ is a time constant,

then, if x_1 is measured, the possible non-zero elements of the first column are (1,1) and (7,1).

3.2.2.2. THE PARAMETERS α AND β

The values chosen for both of these parameters will clearly affect the optimality of the performance of the adaptive filter. At present, no theoretical treatment exists which will enable the calculation of either α or β and so it is necessary to determine them by trial and error experiments using a simulated filter. Kilbride-Newman (3.4) carried out extensive experimentation in order to find suitable values for these parameters and obtained the following results,

$$\alpha = 1/k \quad , \quad 1/k > 0.2$$

$$= 0.2 \quad , \quad 1/k < 0.2$$

$$0.2 < \beta < 0.4.$$

In the case of β the best performance was obtained when a value of 0.3 was used but for different systems this may vary within the limits given above.

3.2.2.3. PARAMETER ESTIMATION AND θ

In the discussion of adaptive filters in section 2.2.6.2. it was stated that the Adaptive Filter due to Kilbride-Newman (3.4) was capable of parameter estimation in addition to state

estimation. This is true so long as equations 3.39 and 3.40 are only included in the filter algorithm at every θ th time increment. This restriction allows the filter to respond as much as it can to changing conditions thus ensuring that the model error compensation term is only conditional on errors caused by a poor mathematical model. This procedure does, however, introduce an unknown parameter, θ , which again has to be determined by trial and error. Kilbride-Newman (3.4) found that the optimal value of θ was 4 but that good results were obtained by the following range of values,

$$3 < \theta < 6$$

3.2.3. DISCUSSION OF THE ADAPTIVE FILTER

Once the parameters α , β and θ and the matrix F_4 have been determined from initial on-line experimentation using the filter, it only remains to specify the following quantities,

$$x(0,0), C(0,0), P(0,0), g(0), \gamma(0), \bar{c}, \\ \bar{w}, R(k), Q(k),$$

before the Adaptive filter derived in section 3.2.1., equation 3.35 to 3.48, can be applied in an on-line environment. The two most powerful

features of this adaptive filter are, (i) the filter's ability to eliminate divergence, and (ii) the model error compensation strategy. The importance of eliminating divergence in chemical engineering applications was stressed in section 2.2.6.2. However, the ability of the adaptive filter to indicate possible areas of modelling error must be an extremely significant asset. Indeed it should eventually be possible to improve poor mathematical models by examination of the values of both F_4 and \bar{w} computed by the filter.

The adaptive filter is not, however, without a number of disadvantages. The earlier discussion on the determination of F_4 highlighted the fact that this may be a lengthy process. Other problem areas which have been indicated are those associated with the choice of α , β and θ . Although the adaptive filter is reasonably robust with respect to these parameters, it was found by Kilbride-Newman (3.4) that in cases where α is too small or β is either too small or too large the estimates become noisy and tend to oscillate about the true values.

The restriction placed on the number of fictitious inputs, \bar{w} , when determining F_4 was

done primarily to combat the problem of unobservability. If one tries to determine too many fictitious inputs the performance of the adaptive filter is likely to display the type of instability obtained when applying any form of the Extended Kalman Filter to processes offering too few measurements.

The increased cycle time of the adaptive filter is not a serious problem in the majority of chemical engineering applications as the time constants associated with such processes are large enough to allow ample time for the execution of the filter algorithm. However, the possibility that convergence will be delayed due to the time required to optimise \bar{w} is a problem which needs consideration. The slower convergence is due to the fact that \bar{w} is only updated every θ time increments. However, this problem will only occur in systems where parameter estimation is being carried out.

3.3. EXCESSIVE NON-LINEARITIES

The original developments made by Kalman (3.7) in the field of sequential estimation are, strictly speaking, only applicable to systems described by sets of linear, first order difference equations. If this theory is to be extended to processes described by sets of non-linear differential equations we need to make the following fundamental assumption (see Sorenson (3.8)):

"A nominal solution of the non-linear differential equations must not only exist but also provide a 'good' approximation to the actual behaviour of the system. The approximation is 'good' if the difference between the nominal and actual solutions can be described by a system of linear differential equations known as 'linear perturbation equations'".

Once this assumption has been made the derivation of the Extended Kalman Filter can proceed. However, as was discussed in section 2.2.4, the reported applications of the Extended Kalman Filter state that there are a number of instances when divergence will occur due to the above assumption being untrue. The proposed solutions to this problem include:

(i) Using numerical integration to compute the predictions.

(ii) Second order filters.

(iii) Iterated filters.

The results obtained using these techniques have, however, not been completely satisfactory. In order to try and identify the problem areas, let us first consider the derivation of the Extended Kalman Filter in greater detail.

The derivation of the Extended Kalman Filter essentially consists of developing an expression to compute a state transition matrix relating the state of the system at time t_k to the state at t_{k+1} . Consider an unforced dynamical system represented by the following set of differential equations.

$$\dot{x} = f(x) \quad - (3.49)$$

where, $f(x)$ is a non linear function with respect to one or more state variables; usually a high proportion in chemical engineering systems.

Now consider the calculation of the state of the system at time t_{k+1} by using simple Euler numerical integration and knowing the state at time t_k ,

$$\dot{\bar{x}}(k) = f(x(k)) \quad - (3.50)$$

and, $x(k+1) = h(x(k)) \doteq x(k) + f(x(k)) \cdot \Delta t$

$$\quad - (3.51)$$

where, $h(x(k))$ represents the integral of $f(x)$, evaluated between the limits t_k and t_{k+1} .

The function $h(x(k))$ is now expanded about a nominal trajectory, $\bar{x}(k)$, using a Taylor series expansion

$$h(x(k)) = h(x(k)) \Big|_{\bar{x}(k)} + (x(k) - \bar{x}(k)) \cdot$$

$$\frac{\partial h(x(k))}{\partial x(k)} \Big|_{\bar{x}(k)} + \frac{(x(k) - \bar{x}(k))^2}{2^1} \cdot$$

$$\frac{\partial^2 h(x(k))}{\partial x(k)^2} \Big|_{\bar{x}(k)} + \dots$$

$$\quad - (3.52)$$

Replacing $h(x(k))$ by $x(k+1)$ and truncating the series to the first derivative we obtain,

$$x(k+1) \doteq \bar{x}(k+1) + (x(k) - \bar{x}(k)) \cdot \frac{\partial h(x(k))}{\partial x(k)} \Big|_{\bar{x}(k)}$$

$$\quad - (3.53)$$

If perturbation variables are now introduced,

$$\delta x = x - \bar{x}$$

equation 3.53 can be simplified to,

$$\delta x(k+1) = \left. \frac{\partial h(x(k))}{\partial x(k)} \right|_{\bar{x}(k)} \delta x(k) \quad - (3.54)$$

Thus, by comparison with linear systems, we can see that the matrix,

$$\left. \frac{\partial h(x(k))}{\partial x(k)} \right|_{\bar{x}(k)} \quad - (3.55)$$

is an approximation to the non-linear state transition matrix, $\Phi(k+1,k)$.

Differentiating equation 3.51 with respect to $x(k)$ gives,

$$\left. \frac{\partial h(x(k))}{\partial x(k)} \right|_{\bar{x}(k)} = I + \left. \frac{\partial f(x(k))}{\partial x(k)} \right|_{\bar{x}(k)} \cdot \Delta t \quad - (3.56)$$

Therefore, our approximate non-linear state transition matrix can be computed by use of the following expression,

$$\Phi(k+1,k) = I + \left. \frac{\partial f(x(k))}{\partial x(k)} \right|_{\bar{x}(k)} \cdot \Delta t \quad - (3.57)$$

where, $\partial f(x(k))/\partial x(k)$ is known as the Jacobian of partial derivatives and is defined as,

$$\frac{\partial f(x(k))}{\partial x(k)} = \begin{bmatrix} \partial f_1 / \partial x_1 & \text{-----} & \partial f_1 / \partial x_n \\ \vdots & & \vdots \\ \partial f_n / \partial x_1 & \text{-----} & \partial f_n / \partial x_n \end{bmatrix}$$

- (3.58)

Thus, it can be clearly seen that the value of $\Phi(k+1,k)$ calculated by the procedure given above is at best only approximately correct. From this statement it follows that as the non-linearity of the system considered increases then the error in the value of $\Phi(k+1,k)$ will also increase. Coggan and Noton (3.9) confirm the existence of the above errors when they state that sufficiently accurate state predictions can only be obtained using such a state transition matrix if the sampling interval, $t_{k+1}-t_k$, is sufficiently small. To overcome this problem most of the reported applications of the Kalman Filter have used numerical integration to compute the predicted state. Thus, although one can be confident that the state predictions are sufficiently accurate, the related statistics may be grossly inaccurate due to the use of an

inaccurate state transition matrix in the calculation of $P(k,k+1)$, see equation 2.11. Clearly this shows that a new and more accurate method for calculating $\phi(k+1,k)$ is required.

3.3.1. ALTERNATIVE PROCEDURES FOR CALCULATING THE STATE TRANSITION MATRIX.

There are two possible approaches to the problem of how to calculate a more accurate state transition matrix:

(i) Post computation - that is, having calculated the predicted state by some type of numerical integration, compute a matrix which will accurately describe the transition from the estimated state at time t_k to the predicted state at time t_{k+1} .

(ii) Pre-computation - that is, develop a method for calculating an accurate state transition matrix knowing only the estimated state at time t_k . This matrix can then be used to compute both the predicted state at time t_{k+1} and the related statistics.

The second of these two approaches would appear to be the most consistent with the requirements of the Kalman Filter. However, a

literature survey of the relevant techniques reveals only one possibility for calculating $\phi(k+1,k)$ in this way. This technique involves the calculation of a functional known as the Matrizant (3.10). Unfortunately no algorithm has yet been developed for the rapid calculation of a numerical value for the Matrizant.

Two techniques which involve the first type of approach are ones that use either multiple linear regression or the Canonical Transformation. The technique utilising multiple linear regression, although feasible, was not adopted because p samples, where p is the number of unknowns in $\phi(k+1,k)$, would have to be taken before evaluation was possible and due to this, the resulting state transition matrix would contain too high a proportion of over weighted historical data. Thus, the most suitable approach is one involving Canonical Transformations.

3.3.2. THE CANONICAL TRANSFORMATION

For the sake of clarity, before considering the calculation of $\phi(k+1,k)$ for non-linear systems, the theory of Canonical Transformations applicable to linear systems will be considered.

Consider the following set of linear differential equations,

$$\dot{x}(t) = F_1 \cdot x(t) \quad - (3.59)$$

with the initial condition,

$$x_0 = x(0) \quad - (3.60)$$

The eigenvalues of this system, λ , are defined by the following equation,

$$| F_1 - \lambda \cdot I | = 0 \quad - (3.61)$$

Corresponding to each of these eigenvalues there is at least one non-trivial solution, v , to equation 3.62.

$$F_1 \cdot v_i = \lambda_i \cdot v_i \quad , \quad i = 1, n \quad - (3.62)$$

The solution corresponding to λ_i is called the i th eigenvector, v_i .

Defining the eigenvector matrix, V , as,

$$V = v_1, v_2, v_3, \dots, v_n \quad - (3.63)$$

it can be shown, see Bellman (3.11), that if F_1 is pre-multiplied by V^{-1} and post-multiplied by V , the resulting matrix, Λ , takes the form $I \cdot \lambda$, i.e.,

$$\Lambda = V^{-1} \cdot F_1 \cdot V = I \lambda \quad - (3.64)$$

Transforming x and \dot{x} as follows,

$$x^* = V^{-1} \cdot x \quad - (3.65)$$

$$\dot{x}^* = V^{-1} \cdot \dot{x} \quad - (3.66)$$

we obtain our transformed system,

$$\dot{x}^* = \Lambda \cdot x^* \quad - (3.67)$$

Since Λ is a diagonal matrix the transformed equations have now been decoupled, that is \dot{x}_i^* depends only on x_i^* . Hence, each individual equation can now be written as,

$$\dot{x}_i^* = \lambda_i \cdot x_i^* \quad - (3.68)$$

the solutions of which are,

$$x_i^*(t) = \exp(\lambda_i \cdot t) \cdot x_i(0)^* \quad - (3.69)$$

Thus, the solution of equation 3.67 is,

$$x^*(t) = I \cdot \begin{bmatrix} \exp(\lambda_1 t) \\ \exp(\lambda_2 t) \\ \vdots \\ \exp(\lambda_n t) \end{bmatrix} \cdot x_0^* \quad - (3.70)$$

writing, the vector,

$$\begin{bmatrix} \exp(\lambda_1 t) \\ \exp(\lambda_2 t) \\ \vdots \\ \exp(\lambda_n t) \end{bmatrix}$$

as $\exp(\lambda t)$, equation 3.70 becomes,

$$x^*(t) = I. \exp(\lambda t). x_0^* \quad - (3.71)$$

If the above equation is written in discrete form,

$$x^*(k+1) = I. \exp(\lambda \Delta t). x^*(k) \quad - (3.72)$$

where, $\Delta t = t_{k+1} - t_k$, and then compared with the equivalent equation for the untransformed state,

$$x(k+1) = \phi(k+1, k). x(k) \quad - (3.73)$$

it is clear that the expression $I. \exp(\lambda \Delta t)$ represents the state transition matrix for the transformed state, $\phi^*(k+1, k)$.

At this point it is worth noting two of the properties of Canonically Transformed systems. These are, (i) The eigenvalues of both the original and the transformed systems are identical.

(ii) The transformed state transition matrix, $\phi^*(k+1, k)$ is a diagonal matrix.

3.3.3. THE APPLICATION OF CANONICAL TRANSFORMATIONS TO NON-LINEAR SYSTEMS

Consider an unforced dynamical process described by the following set of non-linear

differential equations,

$$\dot{x} = f(x) \quad - (3.74)$$

Rearrangement of the above set of equations often makes it possible to rewrite 3.74 in the form,

$$\dot{x} = F_1(x) \cdot x \quad - (3.75)$$

where, $F_1(x)$ is an $n \times n$ matrix of coefficients. This matrix is evaluated by substituting the value of the state vector, x , into the expressions obtained by the rearrangement of equation 3.74. Writing equation 3.75 in discrete form,

$$\dot{x}(k) = F_1(k) \cdot x(k) \quad - (3.76)$$

it can be seen that the matrix $F_1(k)$ is in effect a linearisation of the system equations about the nominal state trajectory at time t_k , i.e.,

$$F_1(k) = F_1(x) \Big|_{x(k)} \quad - (3.76a)$$

Clearly this linearisation procedure will produce a solution which accurately describes the state of the system in the immediate vicinity of the sampling point t_k . The eigenvalues, $\lambda(k)$, and the eigenvector matrix, $v(k)$, corresponding to the solution given by equation 3.76 can now be

determined thus allowing the computation of the transformed state vector, $x^*(k)$, as follows:

$$x^*(k) = V^{-1}(k) \cdot x(k) \quad - (3.77)$$

The next step in the procedure is to calculate a value for the transformed state vector at time t_{k+1} . This can be done by using an appropriate numerical integration technique and then following the same procedure as that given above for time t_k .

Having obtained values for both $x^*(k)$ and $x^*(k+1)$ and assuming the existence of a transformed state transition matrix for non-linear systems, $\phi^*(k+1,k)$, then the equation relating these quantities can be obtained by assuming that such an equation will be of the same form as in the linear case, equation 3.72, i.e.,

$$x^*(k+1) = \phi^*(k+1,k) \cdot x^*(k) \quad - (3.78)$$

Assuming that $\phi^*(k+1,k)$ is a diagonal matrix (c.f. linear case), then it is simply calculated as follows:

$$\phi_{i,i}^*(k+1,k) = x_i^*(k+1) / x_i^*(k) ; i = 1, n \quad - (3.79)$$

It now remains to develop a method for retransforming

$\phi^*(k+1,k)$. This is done by substituting equation 3.77 and the corresponding equation for time t_{k+1} into equation 3.78 and re-arranging,

$$x(k+1) = (V(k+1) \cdot \phi^*(k+1,k) \cdot V^{-1}(k)) \cdot x(k)$$

- (3.80)

Comparing equations 3.80 and 3.73 it can be deduced that,

$$\phi(k+1,k) = V(k+1) \cdot \phi^*(k+1,k) \cdot V^{-1}(k)$$

- (3.81)

3.3.3.1. THE GENERAL NON-LINEAR CASE

The majority of mathematical models describing the transient behaviour of chemical engineering processes consist of sets of equations which are not only non-linear but also include forcing functions or inputs, $u(t)$, and some vector describing process noise, $z(t)$. Thus the general mathematical model will be,

$$\dot{x} = f(x,u,z)$$

- (3.82)

As before, this equation can usually be rearranged (c.f. equations 3.74 and 3.75) to give,

$$\dot{x} = F_1(x,u,z) \cdot x + F_2(x,u,z) \cdot u + F_3(x,u,z) \cdot z$$

- (3.83)

Considering $F_2(x,u,z)$ and $F_3(x,u,z)$ to be linear and stationary, then the discrete form of equation 3.83 is,

$$x(k+1) = \phi(k+1,k). x(k) + \Gamma(k+1,k) \\ (F_2.u(k) + F_3.z(k)) \quad - (3.84)$$

Transformation of equation 3.84 gives,

$$x^*(k+1) = \phi^*(k+1,k). x^*(k) + \Gamma^*(k+1,k). \\ (F_2^*.u(k) + F_3^*.z(k)) \quad - (3.85)$$

$$\text{where, } F_2^* = V(k)^{-1}.F_2 \quad - (3.86)$$

$$\text{and } F_3^* = V^{-1}(k).F_3 \quad - (3.87)$$

Substitution of equations 3.77, 3.86 and 3.87 into 3.85 and comparing the resulting equation with 3.84 yields the following relationships between transformed and untransformed transition matrices,

$$\phi(k+1,k) = V(k+1). \phi^*(k+1,k). V^{-1}(k) \quad - (3.88)$$

$$\Gamma(k+1,k) = V(k+1). \Gamma^*(k+1,k). V^{-1}(k) \quad - (3.89)$$

In order to calculate values for both $\phi^*(k+1,k)$ and $\Gamma^*(k+1,k)$ it is necessary to make the following assumptions,

- (i) $\Gamma^*(k+1,k)$ is a diagonal matrix
(ii) The relationship between $\phi^*(k+1,k)$
and $\Gamma^*(k+1,k)$ is as follows,

$$\Gamma^*(k+1,k) \doteq \phi^*(k+1,k) \cdot \Delta t \quad - (3.90)$$

The first assumption can be justified by considering the derivation of this matrix for the linear case where it can be proved that $\Gamma^*(k+1,k)$ is diagonal. The second assumption is more difficult to justify. However, since the system is discrete the value of $\phi^*(k+1,k)$ will be constant between t_k and t_{k+1} and thus, equation 3.90 which is in fact the correct relationship for linear systems, is a reasonable approximation to other more complex relationships which could be derived.

Combining equations 3.90 and 3.85 and considering the i th state variable gives the following equation for the calculation of $\phi^*(k+1,k)$,

$$\phi_{i,i}^*(k+1,k) = x_i^*(k+1) / (x_i^*(k) + \Delta t \cdot s_i(k))$$

$$; i = 1, n \quad - (3.91)$$

where, $s(k) = F_2^* u(k) + F_3^* z(k)$. Once the transformed state transition matrix, $\phi^*(k+1,k)$, has been calculated by the above method, the

transformed integral state transition matrix can be determined using equation 3.90.

3.3.3.2. SUMMARY OF THE CALCULATION PROCEDURE

The complete calculation procedure for the general non-linear case can now be summarised as follows:

- (i) Compute $F_1(k)$ - equation 3.76a.
- (ii) Determine the eigenvalues, $\lambda(k)$, and the eigenvectors, $V(k)$.
- (iii) Calculate the transformed state vector, $x^*(k)$ - equation 3.77, and $s(k)$ - equation 3.85.
- (iv) By an appropriate type of numerical integration determine the solution of equation 3.82, $x(k+1)$.
- (v) Compute $F_1(k+1)$ - equation 3.76a.
- (vi) Determine the eigenvalues, $\lambda(k+1)$, and the eigenvectors, $V(k+1)$.
- (vii) Calculate the transformed state vector, $x^*(k+1)$ - equation 3.77, and $\bar{s}(k+1)$ - equation 3.85.
- (viii) Compute $\phi^*(k+1,k)$ - equation 3.91.
- (ix) Compute $\Gamma^*(k+1,k)$ - equation 3.90.
- (x) Retransform both $\phi^*(k+1,k)$ and $\Gamma^*(k+1,k)$ - equations 3.88 and 3.89.

The above procedure is presented schematically in Figure 3.1.

3.3.4. DISCUSSION OF CANONISATION PROCEDURE

The procedure summarised in the preceding section produces a considerably more accurate value for the state transition matrix, $\phi(k+1,k)$, than was achieved by previous methods because of the way it 'fits' the matrix to a known change in state. Initial tests show that when both types of $\phi(k+1,k)$ (i.e. the one derived at the beginning of section 3.3 and the one summarised in section 3.3.3.2) are used to calculate the value of $x(k+1)$ from $x(k)$ the error is reduced from 10% to less than 1%. From this result it can be concluded that the value of $\phi(k+1,k)$ found by the new procedure 'fits' the change from $x(k)$ to $x(k+1)$ very accurately whereas the value of $\phi(k+1,k)$ found by the previous method is a poor approximation to the correct value.

The introduction of this canonisation procedure into the Kalman Filter will essentially produce a new type of filter with, as yet, unknown characteristics. The cycle time of this new

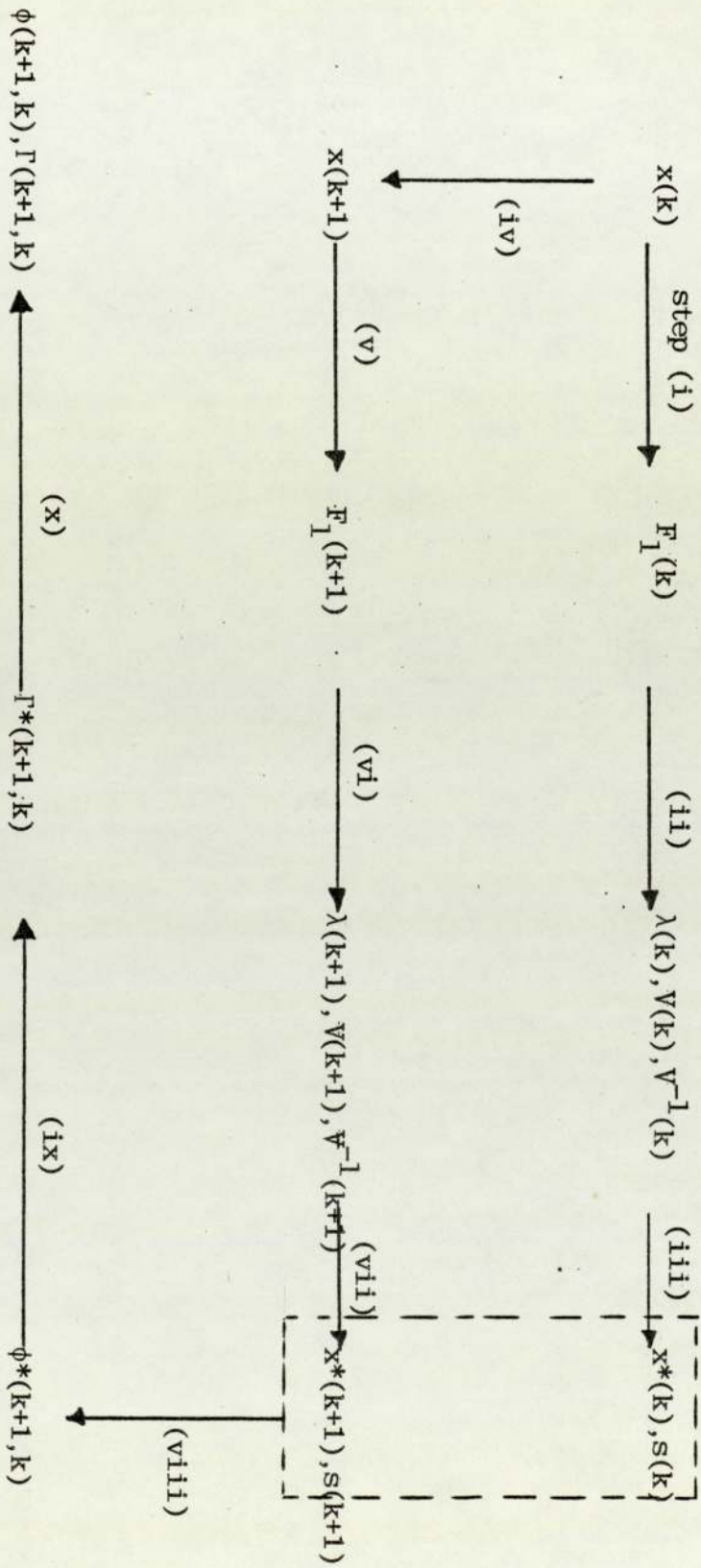


FIGURE 3.1 - CALCULATION PROCEDURE FOR EVALUATING $\phi(k+1, k)$ AND $\Gamma(k+1, k)$ BY A CANONISATION METHOD

filter will undoubtedly be greater than that for a conventional Extended Kalman Filter, but in cases of high non-linearity where the system time constants are large (i.e. chemical engineering processes) it will enable greater confidence to be placed in the statistics generated by the filter. One further advantage which has not yet been discussed is the absence of unknown parameters from the filter. This must surely be a significant advantage over many filters proposed in the literature.

Prior to the on-line implementation of the new filter a number of points needed to be investigated in a simulated situation. These points are as follows:

(i) The characteristics of the new filter need to be determined.

(ii) An efficient routine for calculating the eigenvalues and eigenvectors needs to be found.

(iii) The absence of undesirable numerical/roundoff errors needs to be established.

3.4. CHAPTER REVIEW

Two modifications to the Kalman Filter have been developed in order to eliminate divergence and bias problems due to,

- (i) A poor mathematical model.
- (ii) Excessive non-linearities.

Discussion of these developments has revealed a number of areas requiring further investigation prior to the on-line implementation of the modified filters. This investigation will be discussed in chapter four.

CHAPTER 4

SIMULATION STUDIES OF THE KALMAN FILTER

4.1. INTRODUCTION

In section 2.2.5 the discussion on the application of Kalman Filters to chemical engineering processes reported that the majority of the research carried out in this area has been concerned with the performance of simulated filters. It was noted that although the results given in the literature were encouraging, a number of problems associated with the on-line implementation of Kalman Filters had been overlooked. This does not mean that these simulation studies are devalued in any way, merely that they should be regarded as the first and indeed an essential step in the application of Kalman Filters to chemical engineering processes. In the author's own experience, the most important features of such simulation studies are that they enable the elimination of any numerical problems associated with the proposed filter and allow the filter performance to be assessed in a situation which is accurately known. Thus, the results obtained in simulation studies give a good indication of what to expect when the filter is applied in an on-line context, and this is the main objective of the investigation reported in this chapter.

The theoretical developments proposed in chapter three are aimed at ensuring and promoting the convergence of the Kalman Filter when applied to chemical engineering processes. In order to assess the performance of these new filters, the accuracy of the estimates produced will be compared with the accuracy of those generated by standard forms of the Kalman Filter. This investigation takes the form of firstly producing noisy measurements from a computer program written to simulate the behaviour of a three tank blending system. These results are then used as measurement data to test each filter being studied. Finally the estimates generated are output in both graphical and digital forms.

4.2. THEORETICAL BACKGROUND

The experiments done in this study of the performance of the Kalman Filter were carried out by using three computer programs. The separation of the experimental procedure into these sections was done primarily to simplify the task of writing the programs. The first of these is a program to simulate the transient and steady state behaviour of a three tank blending system. This program, known as SIMULPROG, uses a mathematical model of a blending system similar to the one used by Coggan and Noton (4.1) in earlier simulation studies of the Extended Kalman Filter. In order to produce realistic measurement data the results generated by this program are corrupted at the sampling times by the addition of Gaussian noise. The second computer program includes five different forms of the Kalman Filter and is known as FILTER. This program was written in modular form and is similar in structure to on-line filtering programs except that the results output are more comprehensive than would normally be required and, due to its off-line nature, it uses a simpler method for acquiring measurement data. Although this program does not take place in real time it gives a good indication of the likely cycle time of each

filter because the program is run separately for each of the filters. In this program calculation of the errors in the estimates produced by each filter is straightforward since the correct values are available. These error calculations are carried out after each cycle of the filter. Two sets of results are produced by this program; the first set may be output to a line printer, the other being stored inside the computer for use by the third program. The third program is called GRAPHPLOT and its function is to display graphically the estimates produced by FILTER.

4.2.1. THE THREE TANK BLENDING SYSTEM

The process chosen to test the filters is the three tank blending system shown schematically in Figure 4.1. This system is not intended to represent a real plant but is thought to exhibit features which are not only typical of chemical processes but will also impose on the filters the kind of rigorous tests required. These features include non-linearities within the mathematical description of the process, an unknown variable transport lag, measurements which are few and inaccurate and of course unmeasured random disturbances. One further point which influenced the choice of this process is that it contains a

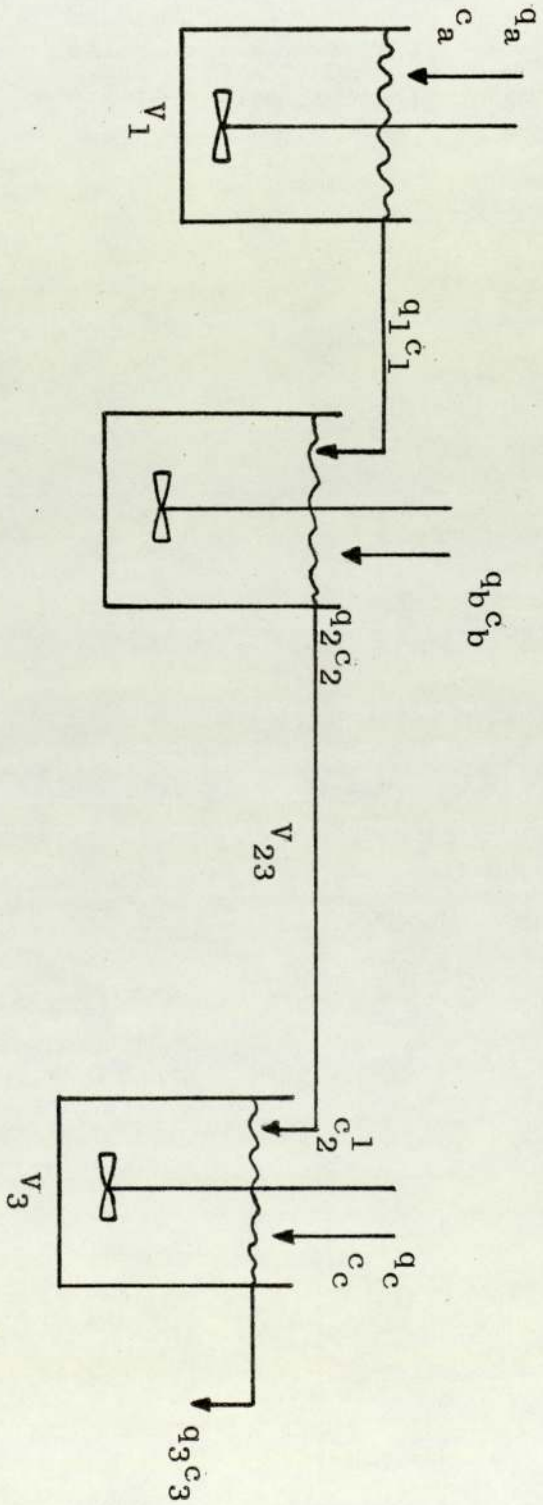


FIGURE 4.1 THREE TANK BLENDING SYSTEM

relatively large number of state variables and thus in this sense is not only representative of real plants but also enables a realistic study of the problems associated with the important concept of observability.

With reference to Figure 4.1, q is a volumetric flowrate, C is a solute concentration, and V is the known volumetric hold-up of a vessel. Perfect mixing is assumed in each vessel but in the dynamic state the concentration C_2' entering vessel 3 differs from C_2 the concentration in vessel 2 due to a transport lag in the long pipe connecting the two vessels. Each vessel has a hydraulic time constant, τ_n , but the effect of variation in hold-up with throughput upon the concentration dynamics is assumed to be swamped by other effects and is neglected. The state variables and known parameters of this process are defined in Table 4.1.

Random disturbances enter the process only through the feed streams, a , b and c and because of this the inputs are estimated along with the other process variables. To enable this estimation the inputs are described by the following type of differential equation,

$$\dot{x}_j = a_j (x_j - \bar{x}_j) \quad - (4.1)$$

TABLE 4.1 - BLENDING SYSTEM VARIABLES

STATE VARIABLE	PROCESS VARIABLE	NOMINAL VALUE	STEADY STATE VALUE	AUTO-CORRELATION COEFFICIENT	* INDICATES MEASURED VARIABLE
x ₁	q ₁	5.00	-	-	*
x ₂	C ₁	0.50	-	-	*
x ₃	q ₂	7.00	-	-	
x ₄	C ₂	0.40	-	-	
x ₅	q ₃	10.00	-	-	*
x ₆	C ₃	0.38	-	-	*
x ₇	q _a	5.00	4.00	0.50	
x ₈	C _a	0.50	0.20	0.85	
x ₉	q _b	2.00	1.50	0.80	
x ₁₀	C _b	0.35	0.50	0.50	*
x ₁₁	q _c	2.90	3.50	0.90	*
x ₁₂	C _c	0.30	0.70	0.50	
x ₁₃	C ₂	0.40	-	-	

- BLENDING SYSTEM PARAMETERS

TANK 1 $\tau_{n1} = 0.22$ hr $V_1 = 10.0$ M³ TANK 2 $\tau_{n2} = 0.25$ hr $V_2 = 16.0$ M³

TANK 3 $\tau_{n3} = 0.32$ hr $V_3 = 26.0$ M³

VOLUMETRIC HOLD-UP, $V_{23} = 5.0$ M³

SAMPLING INTERVAL = 0.25 hr

where, $a_j = \log_e (\alpha_j)/S$

α_j is an auto-correlation coefficient

S is the sampling interval (hrs)

x_j is an input to the system

and, \bar{x}_j is the steady state value of x_j

It is via equation 4.1 that both random and deterministic changes are introduced into the system. Random disturbances are simulated by the addition of Gaussian noise to x_j prior to the calculation of the derivative of this state variable. Since the nominal values of the inputs differ appreciably from their steady state values the expected behaviour of these state variables will be an exponential approach towards the steady state value; the slope of this approach depending upon the value selected for the auto-correlation coefficient, α_j . When x_j has converged on \bar{x}_j , the value of this input will display random deviations from the steady state value due to the random noise introduced into the system. Once the system has reached steady state it is possible to introduce further transients by varying the values of \bar{x}_j . Clearly, such variations will cause the system to move towards a new steady state.

The state equations describing the blending system are:

$$\dot{x}_1 = (x_7 - x_1)/\tau_{n1} \quad - (4.2)$$

$$\dot{x}_2 = (x_7 \cdot x_8 - x_1 \cdot x_2)/V_1 \quad - (4.3)$$

$$\dot{x}_3 = (x_1 - x_3 + x_9)/\tau_{n2} \quad - (4.4)$$

$$\dot{x}_4 = (x_1 \cdot x_2 - x_3 \cdot x_4 + x_9 \cdot x_{10})/V_2 \quad - (4.5)$$

$$\dot{x}_5 = (x_3 - x_5 + x_{11})/\tau_{n3} \quad - (4.6)$$

$$\dot{x}_6 = (x_3 \cdot x_{13} - x_5 \cdot x_6 + x_{11} \cdot x_{12})/V_3 \quad - (4.7)$$

$$\dot{x}_7 = a_1 \cdot (x_7 - \bar{x}_7) \quad - (4.8)$$

$$\dot{x}_8 = a_2 \cdot (x_8 - \bar{x}_8) \quad - (4.9)$$

$$\dot{x}_9 = a_3 \cdot (x_9 - \bar{x}_9) \quad - (4.10)$$

$$\dot{x}_{10} = a_4 \cdot (x_{10} - \bar{x}_{10}) \quad - (4.11)$$

$$\dot{x}_{11} = a_5 \cdot (x_{11} - \bar{x}_{11}) \quad - (4.12)$$

$$\dot{x}_{12} = a_6 \cdot (x_{12} - \bar{x}_{12}) \quad - (4.13)$$

$$\dot{x}_{13}(t) = x_4(t-T) \quad - (4.15)$$

$$T = V_{23}/x_3$$

where, V_{23} is the volumetric hold-up in the pipe between vessels 2 and 3.

For the purpose of simulation the concentration entering vessel 3, C_2' , is treated as the state variable x_{13} . It is calculated by storing all of the previous values of x_4 in an array and then when a value of x_{13} is required, this is found by calculating $t-T$ and then linearly interpolating between the two consecutive values of x_4 which lie either side of $t-T$. During the initial period when the value of $t-T$ is negative, the value of x_{13} is assumed to remain constant at its nominal value.

4.2.2. DESCRIPTION OF THE FILTERS TESTED

The performance of five different forms of the Kalman Filter are assessed in this investigation and for the sake of brevity each of these filters has been assigned a reference number which will be used as a means of identification throughout the remainder of this thesis. The task of classifying these filters is achieved by dividing the process of filtering into the following three steps:

(I) PREDICTION - The calculation of the predicted state, $x(k+1,k)$.

(II) STATE TRANSITION - The calculation of the state transition matrix, $\phi(k+1,k)$

(III) ESTIMATION - The calculation of the estimated state, $x(k,k)$.

Each of these steps can in turn be carried out by two different methods:

Methods of PREDICTION:- (i) $x(k+1,k)$ is calculated by first determining $\phi(k+1,k)$ from the estimated state $x(k,k)$ and then using the following equation,

$$x(k+1,k) = \phi(k+1,k) \cdot x(k,k) \quad - (4.16)$$

(c.f. equation 2.10) This method will be referred to as 'Prediction via the State Transition Matrix'.

(ii) $x(k+1,k)$ is calculated by the use of the Runge-Kutta 4 type of numerical

integration; for a detailed account of this form of numerical integration see Lapidus (4.2). From now on this method will be referred to as 'Prediction via RK4 integration'. Methods of STATE TRANSITION:- (i) The state transition matrix, $\phi(k+1,k)$, is calculated by using equations 3.57 and 3.58. This method will be known as 'The calculation of the state transition matrix by a truncated Taylor series'.

(ii) The state transition matrix, $\phi(k+1,k)$, is calculated by the procedure given in section 3.3.3.2. This method will be referred to as 'The calculation of the state transition matrix by canonisation'.

Methods of ESTIMATION:- (i) With the exception of the prediction step, equation 2.10, estimation proceeds as shown in section 2.2.3., equations 2.11 to 2.15. This method will be known as the 'Kalman Filter'.

(ii) Estimation proceeds as outlined in section 3.2, equations 3.35 to 3.48. This method will be known as the 'Adaptive Filter'. N.B. When using this method and using Prediction via RK4 it is necessary to take into account the model error vector \bar{w} when calculating the derivatives of the state variables - see equation 3.32.

The filters tested in this investigation can

now be classified as follows.

TYPE 1 :- Prediction via state transition.

State transition matrix calculated by
a truncated Taylor Series.

Estimation by the Kalman Filter.

TYPE 2 :- Prediction via RK4.

State transition matrix calculated by
a truncated Taylor Series.

Estimation by the Kalman Filter.

TYPE 3 :- Prediction via RK4.

State transition matrix calculated by
canonisation.

Estimation by the Kalman Filter.

TYPE 4 :- Prediction via RK4.

State transition matrix calculated by
a truncated Taylor Series.

Estimation by the Adaptive Filter.

TYPE 5 :- Prediction via RK4.

State transition matrix by canonisation.

Estimation by the Adaptive Filter.

Of these five filters the first two are standard forms of the Extended Kalman Filter and the last three are new filters which are to be tested.

4.2.3. CALCULATION OF EIGENVALUES AND EIGENVECTORS

In order to determine the state transition

matrix by the method given in section 3.3.3.2 it is necessary to be able to compute rapidly the eigenvalues and eigenvectors of a known matrix A.

The fundamental algebraic eigenproblem is the determination of those values of λ for which the set of n simultaneous linear equations in n knowns,

$$A \cdot x = \lambda \cdot x \quad - (4.17)$$

has a non-trivial solution. Equation 4.17 may be written in the form,

$$(A - \lambda \cdot I) \cdot v = 0 \quad - (4.18)$$

and for arbitrary λ this set of equations has only the trivial solution $v = 0$. The general theory of simultaneous linear algebraic equations shows that there is a non-trivial solution if, and only if, the matrix $(A - \lambda I)$ is singular, i.e.

$$\det (A - \lambda I) = 0 \quad - (4.19)$$

Equation 4.19 is called the characteristic equation of the matrix A and its roots are called the eigenvalues, λ . Corresponding to each eigenvalue, λ , the set of equations 4.18 has at least one non-trivial solution, v , known as an eigenvector. For the case of distinct eigenvalues it can be shown that there exists a set of n eigenvectors each of which is unique and linearly independent apart from an arbitrary multiplier. In most cases it is convenient to choose this multiplier so that

the eigenvector has some desirable numerical property, such vectors are called normalised vectors.

In situations where the dimension of the matrix is large or when the eigenvalues and eigenvectors need to be calculated often the most suitable techniques are normally numerical methods. Wilkinson (4.3) gives excellent explanations of some of these methods. Many of these numerical techniques for finding the eigensystem of a matrix consist essentially of the determination of a similarity transformation which reduces a matrix A of general form to a matrix B of special form, for which the eigenproblem may be more simply solved. One of the many algorithms for determining eigenvalues is the LR transformation due to Rutishauser (4.4). The development of this method is regarded by Wilkinson (4.3) as the most significant advance made in connection with the eigenvalue problem since the advent of automatic computers. Rutishauser's algorithm gives a reduction of a general matrix to triangular form by means of non-unitary transformations. Rutishauser (4.4) writes

$$A = L.R \quad - (4.20)$$

where L is the unit lower-triangular matrix and

R is the upper-triangular matrix. Suppose we now form the similarity transform $L^{-1}.A.L.$ of the matrix A, this gives,

$$L^{-1}.A.L = L^{-1} (L.R). L = R.L. \quad - (4.21)$$

Hence, if we decompose A and then multiply the factors in the reverse order, a matrix similar to A is obtained. In the LR algorithm this process is repeated indefinitely. If we rename the original matrix A_1 then the algorithm is defined by the equations

$$A_{s-1} = L_{s-1} R_{s-1} \quad - (4.22)$$

$$R_{s-1} \cdot L_{s-1} = A_s$$

Clearly A_s is similar to A_{s-1} and hence by induction to A_1 . Rutishauser (4.4) showed that under certain restrictions,

$$L_s \rightarrow I \text{ and } R_s \rightarrow A_s \rightarrow \begin{bmatrix} \lambda_1 & & X \\ & \lambda_2 & \\ & 0 & \\ & & \lambda_n \end{bmatrix} \quad - (4.23)$$

as $s \rightarrow \infty$

Reference (4.5) gives an example of the application of the LR algorithm to computer based eigenproblems. The computer program given is written in the Fortran IV language (for details of Fortran IV

programming see McCracken (4.6)), and so is easily utilised in this study as the same language is used. This example elaborates considerably on the basic LR method and the resulting program is long and involved. However, because certain extra features are included, it is computationally quite efficient. The program incorporates the following features:

- (i) Economy of storage,
- (ii) Special handling of tridiagonal matrices to take advantage of their high proportion of zeros, and,
- (iii) Acceleration of convergence, when appropriate.

4.3. THE ICL 1904S COMPUTER

The computer programs written for these simulation studies incorporate a number of features not usually included in on-line programs. These features are as follows:

- (i) Increased error checking
- (ii) Error calculation
- (iii) Comprehensive printout.

Thus, the programs require a large amount of computer time and core space. It was therefore decided to use the university's central computer in this investigation.

The present configuration is:-

- (1) An ICL 1904S Central Processing Unit with hardware floating point arithmetic and 192K words of 500 nanosecond memory (word = 24 bits).
- (2) An Operators Console.
- (3) Input: 2101 card reader, 2000 cards/minute
1916 paper tape reader, 1000 characters/second.
- (4) Output: 1933 lineprinter, 1350 lines/minute,
120 characters/line.
1925 paper tape punch, 110 characters/sec.
1934 graph plotter, 30 inches wide,
300 steps/sec, step size 0.005 ins.

(5) Magnetic Tapes:

4 magnetic tape decks using 0.5 ins magnetic tape. Information is stored across 9 tracks at a density of 1600 characters/inch. The maximum transfer rate is 160,000 characters/sec.

(6) Magnetic Discs:

4 exchangeable disc drives, each drive holding 8,192,000 characters stored on 203 tracks. The transfer rate is 208,000 characters/sec.

4 exchangeable disc drives, each drive holding 60,000,000 stored on 203 tracks. The transfer rate is 417,000 characters/sec.

(7) Magnetic Drums:

1 1964 Slow speed drum with storage capacity of 512K words. The maximum transfer rate is 100K characters/sec.

1 2851 High speed drum with storage capacity of 512K. The maximum transfer rate is 1300K characters/sec.

(8) Front End Processor:

A Digico Micro 16v acts as a front end processor to the 1904S mainframe. At present there are 32 terminals. Transfer rates are 2400 bauds for V.D.U.'s and 110 bands for teletypes.

4.4. DISCUSSION OF COMPUTER PROGRAMS

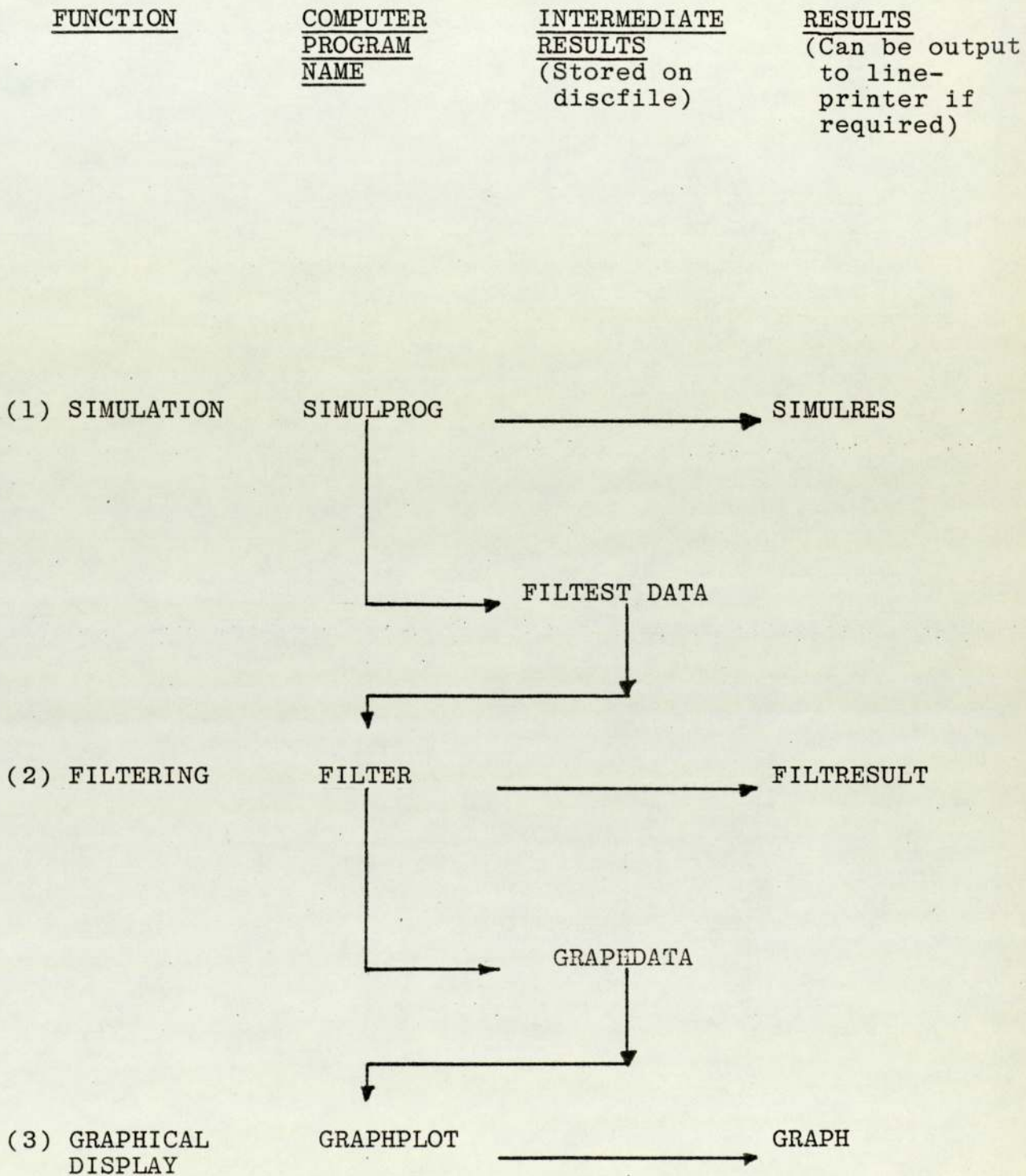
As was stated in section 4.2 this investigation into the behaviour of various forms of the Kalman Filter was carried out using three computer programs. The relationship between these programs and the order in which they are executed is shown schematically in Figure 4.2. The results produced by both the simulation program, SIMULPROG, and the filtering program, FILTER, were initially stored on disc files so that they could be examined using a visual display unit (V.D.U.) before either a permanent copy was produced on the line printer or a graph was plotted using the graph plotting program, GRAPHPLOT.

All of the programs used in this investigation are written in the FORTRAN IV computer language according to the specifications given by McCracken (4.6).

4.4.1. THE SIMULATION PROGRAM - SIMULPROG

Simulation of the behaviour of the three tank blending system shown in Figure 4.1 is achieved by the numerical integration of the mathematical model of the process given by equations 4.2 to 4.13. The writing of the computer program to perform this task was considerably simplified by the use of a set of subroutines to perform the integration. These

FIGURE 4.2 - RELATIONSHIP OF COMPUTER PROGRAMS



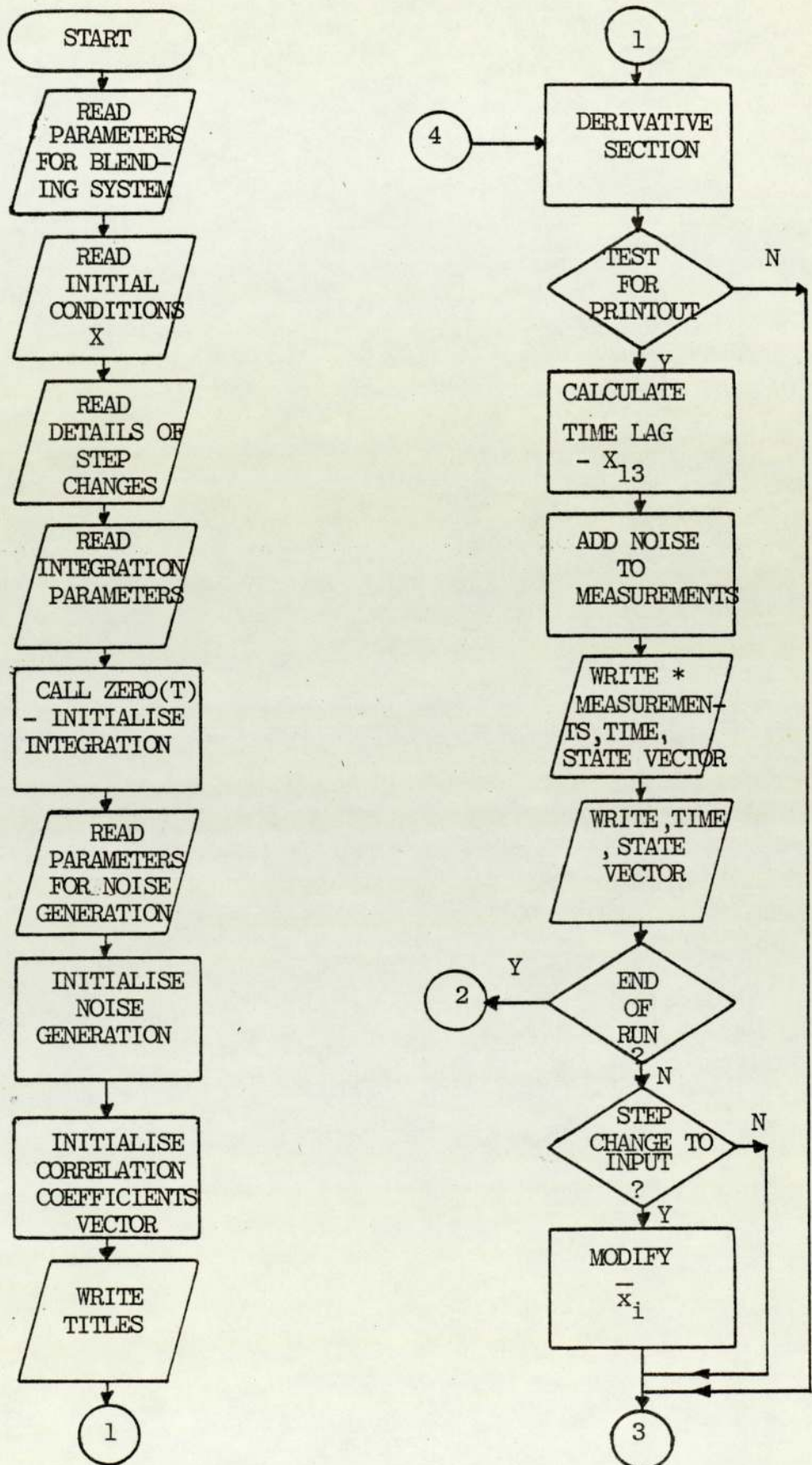
subroutines are in fact modified forms of those first published by Gay and Payne (4.7). In this earlier work Gay and Payne interfaced a set of four Fortran subroutines to a BASIC-16 interpreter to produce an interactive digital simulation package known as the Aston Simulation Package (ASP): a fuller description of this package will be given in section 6.4. These subroutines, after modifications to give them greater compatibility with a main program written in Fortran, were used to form the program SIMULPROG. The structure of the main segment of this program is the same as that given in Figure 6.2 and is shown schematically by the flowchart given in Figure 4.3. In this flowchart all write instructions, with the exception of those indicating otherwise, direct output to disc file SIMULRES.

Using the ASP subroutines it is possible to perform numerical integration using either the modified Euler method or the Runge-Kutta fourth order method. In this application the Runge-Kutta method is used.

4.4.2. THE FILTERING PROGRAM - FILTER.

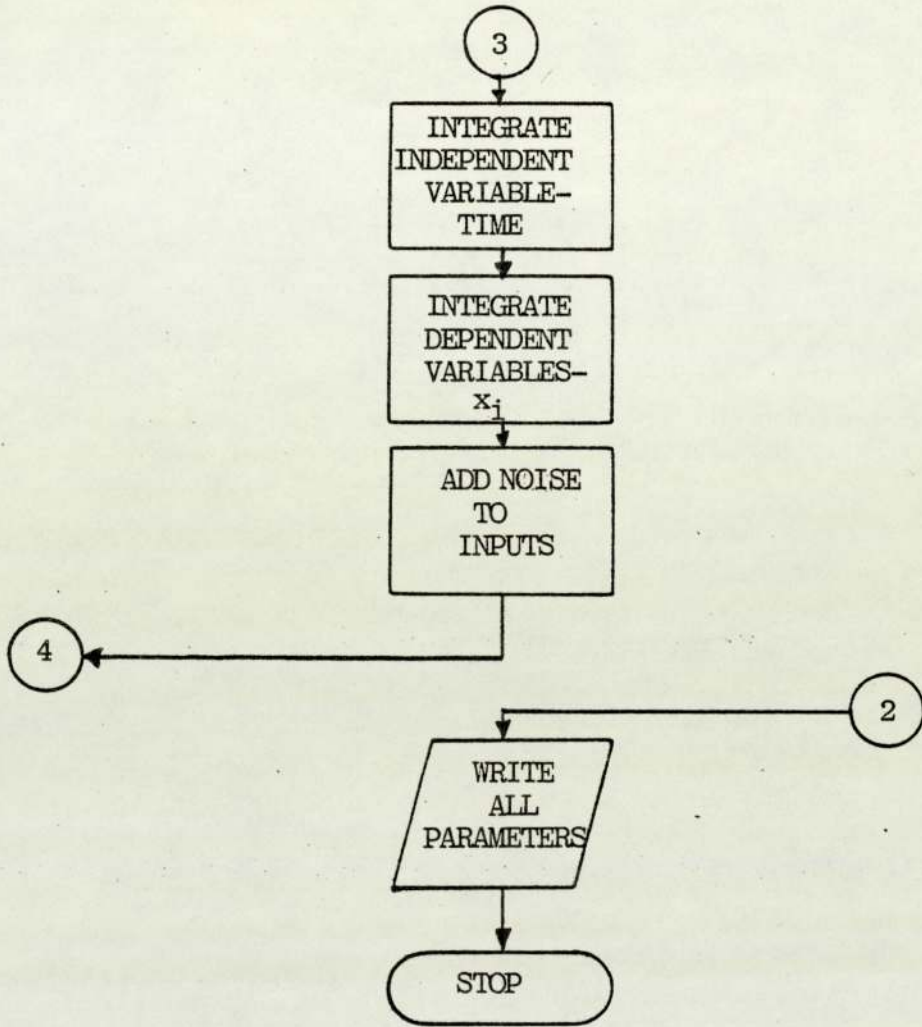
The process of filtering the data produced by the program SIMULPROG is carried out by the program

FIGURE 4.3 - FLOWCHART OF SIMULPROG



*This output is directed to FILESTDATA

FIGURE 4.3 - continued



FILTER which includes the five different forms of the Kalman Filter described in section 4.2.2. Due to the variety of filters to be included it was decided to write this program in modular form, i.e. as a set of subroutines. The subroutines can then be called, when required, by a main-program named FILTEST. Thus, the complete program consists of the following segments:-

MAIN SEGMENT :- FILTEST

PREDICTION SEGMENTS:- Subroutine PRED1 - prediction via the State Transition Matrix. Subroutine PRED2 - prediction via RK4 numerical integration.

SEGMENTS TO CALCULATE

THE STATE TRANSITION

MATRICES:- Subroutine TRANS1 - Calculation using a truncated Taylor series. Subroutine TRANS2 - Calculation by canonisation.

ESTIMATION SEGMENTS:- Subroutine KALMAN - Kalman Filter. Subroutine ADAPT - Adaptive Filter.

In addition to the above segments, the following subroutines are necessary in order to complete the program:-

- Subroutine MODEL - Calculates the coefficients matrix $A(x)$ as defined by,
- $$\dot{x} = A(x) \cdot x$$
- Subroutine RUTIS - Calculates the eigenvalues and eigenvectors of a Matrix using Rutishausers LR transformation ; see reference (4.5).
- Subroutine MATMUL - performs matrix multiplication.
- Subroutine MATADD - performs matrix addition.
- Subroutine MATRAP - performs matrix transposition.
- Subroutine MATINV - performs matrix inversion by the method of Gauss-Jordan.
- Subroutines ZERO, PRNTE, INTI, INTX - perform numerical integration by the fourth order Runge-Kutta method. These routines are again modifications of those used by Gay and Payne (4.7)

Throughout the program communication between segments is by argument transfer. The complete program occupies 29K words of computer core space and takes between 70 and 170 seconds to run depending on the type of filter used.

The more important segments of FILTER will now be discussed individually.

4.4.2.1. THE MAIN SEGMENT - FILTEST

FILTEST is the main segment of FILTER and essentially its functions are to handle input and output, call the subroutines necessary to perform the task of filtering, calculate the errors in the estimated state vector and test for the end of the run. The type of filter used in a particular run is determined by the value of the variable ITYPE and, as shown by Figure 4.4., the five different types of filtering are achieved by the following combinations of subroutines:-

Type 1 Filter	- (1) KALMAN
	(2) TRANS1
	(3) PRED1
Type 2 Filter	- (1) KALMAN
	(2) PRED2
	(3) TRANS1
Type 3 Filter	(1) KALMAN
	(2) PRED2
	(3) TRANS2
Type 4 Filter	- (1) ADAPT
	(2) PRED2
	(3) TRANS1
Type 5 Filter	- (1) ADAPT
	(2) PRED2
	(3) TRANS2

FIGURE 4.4 - FLOWCHART OF FILTEST

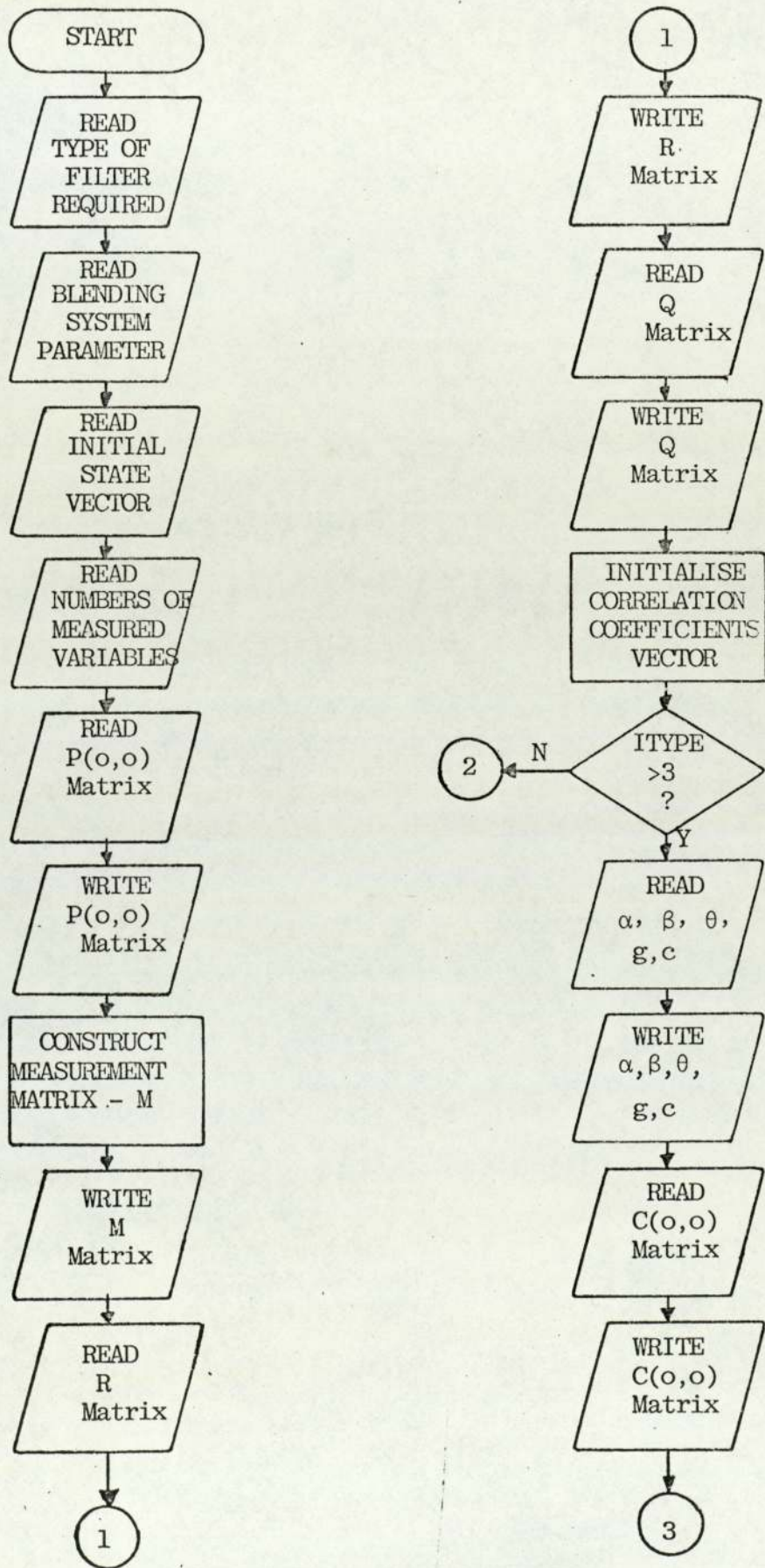
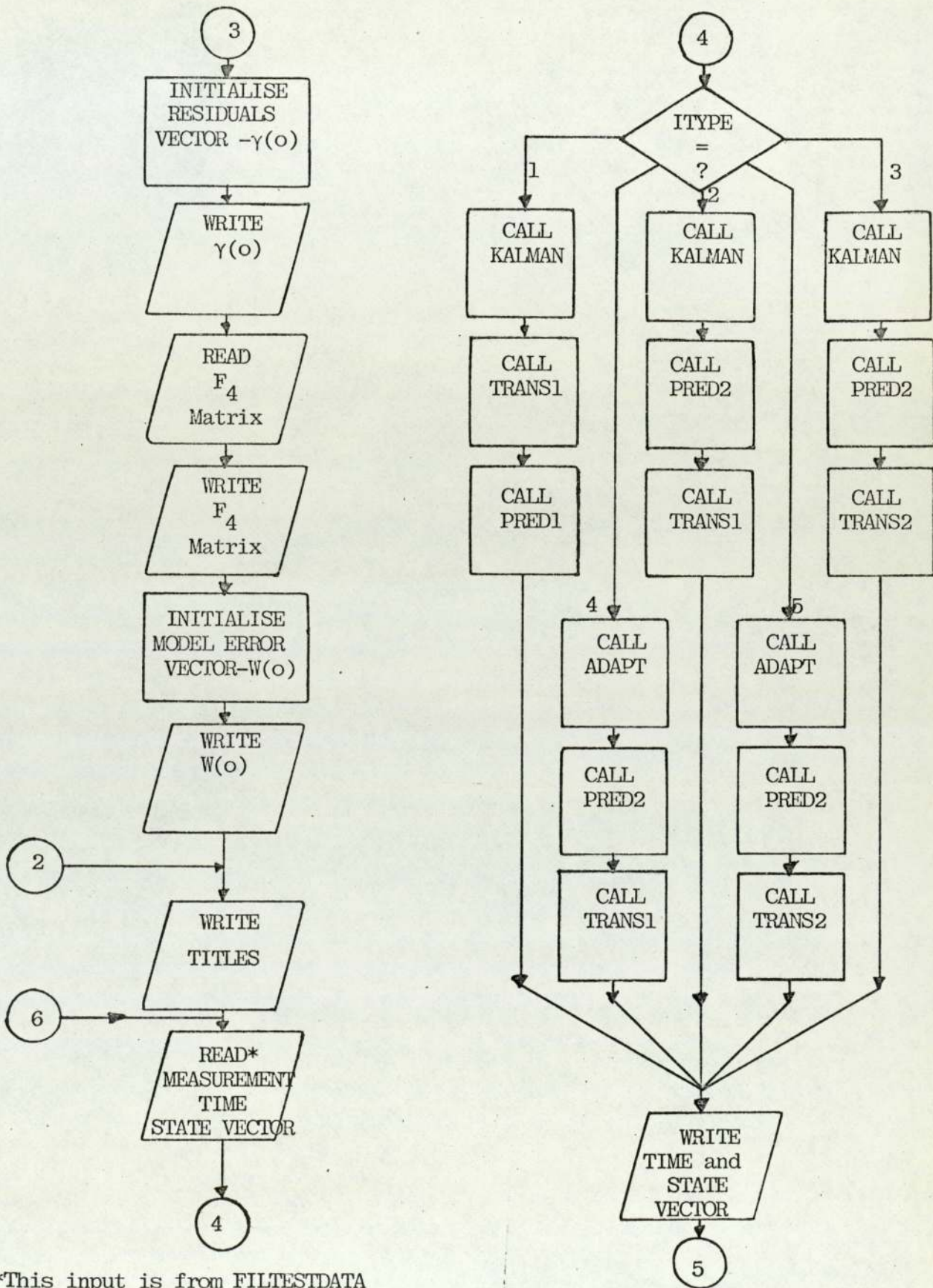
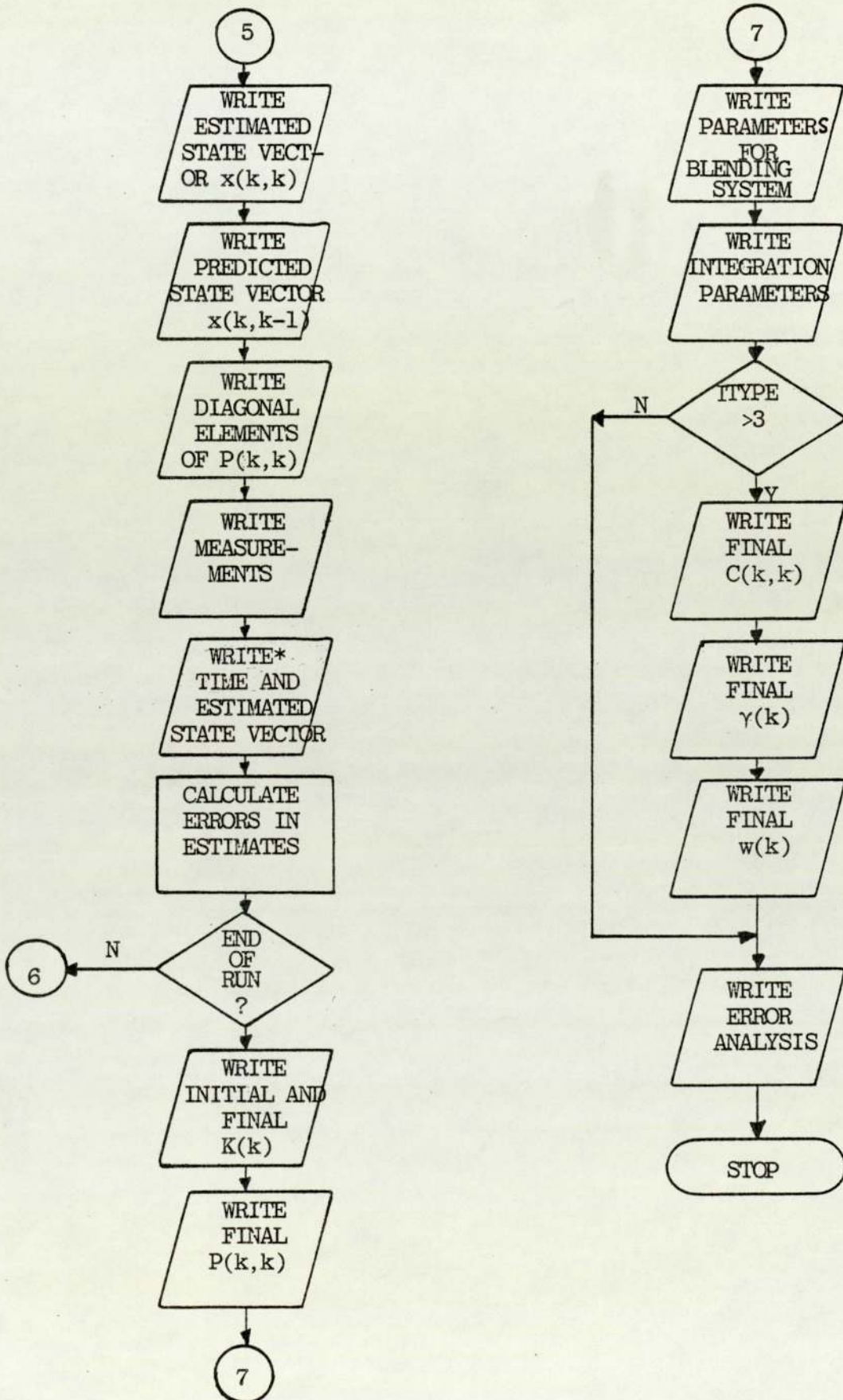


FIGURE 4.4 - continued



*This input is from FILTESTDATA

FIGURE 4.4 - continued



*This output is directed to GRAPHDATA

4.4.2.2. THE ESTIMATION SEGMENTS - KALMAN AND ADAPT

One of the most attractive features of the Kalman Filter is its ease of implementation on a digital computer and as a result both of the estimation segments consist simply of CALLS on the matrix manipulation routines. The flowchart of subroutine KALMAN is given in Figure 4.5 and as can be seen it is simply an equation by equation translation of the algorithm defined by equations 2.11 to 2.15.

Figure 4.6 shows the flowchart of subroutine ADAPT which, despite its additional complexity and storage requirements, is also easily translated from the algorithm defined by equations 3.35 to 3.48.

4.4.2.3. THE PREDICTION SEGMENTS - PRED1 and PRED2

The simplest of the segments in FILTER is subroutine PRED1. Since all that is required of PRED1 is to multiply the estimated state vector, $x(k,k)$, by the state transition matrix, $\phi(k+1,k)$, calculated by subroutine TRANS1, this subroutine consists of just two lines of program. However, as shown by Figure 4.7, subroutine PRED2 is rather more complex. In this segment of the program the

FIGURE 4.5 - FLOWCHART OF SUBROUTINE KALMAN

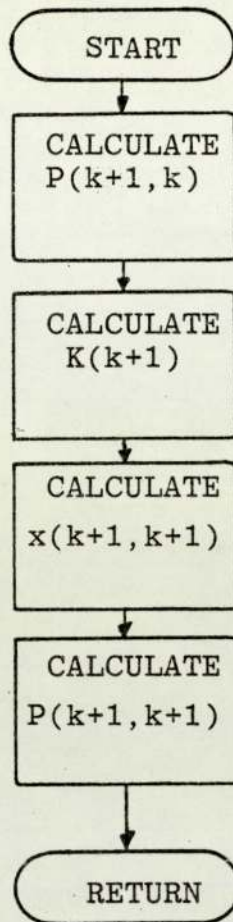


FIGURE 4.6 - FLOWCHART OF SUBROUTINE ADAPT

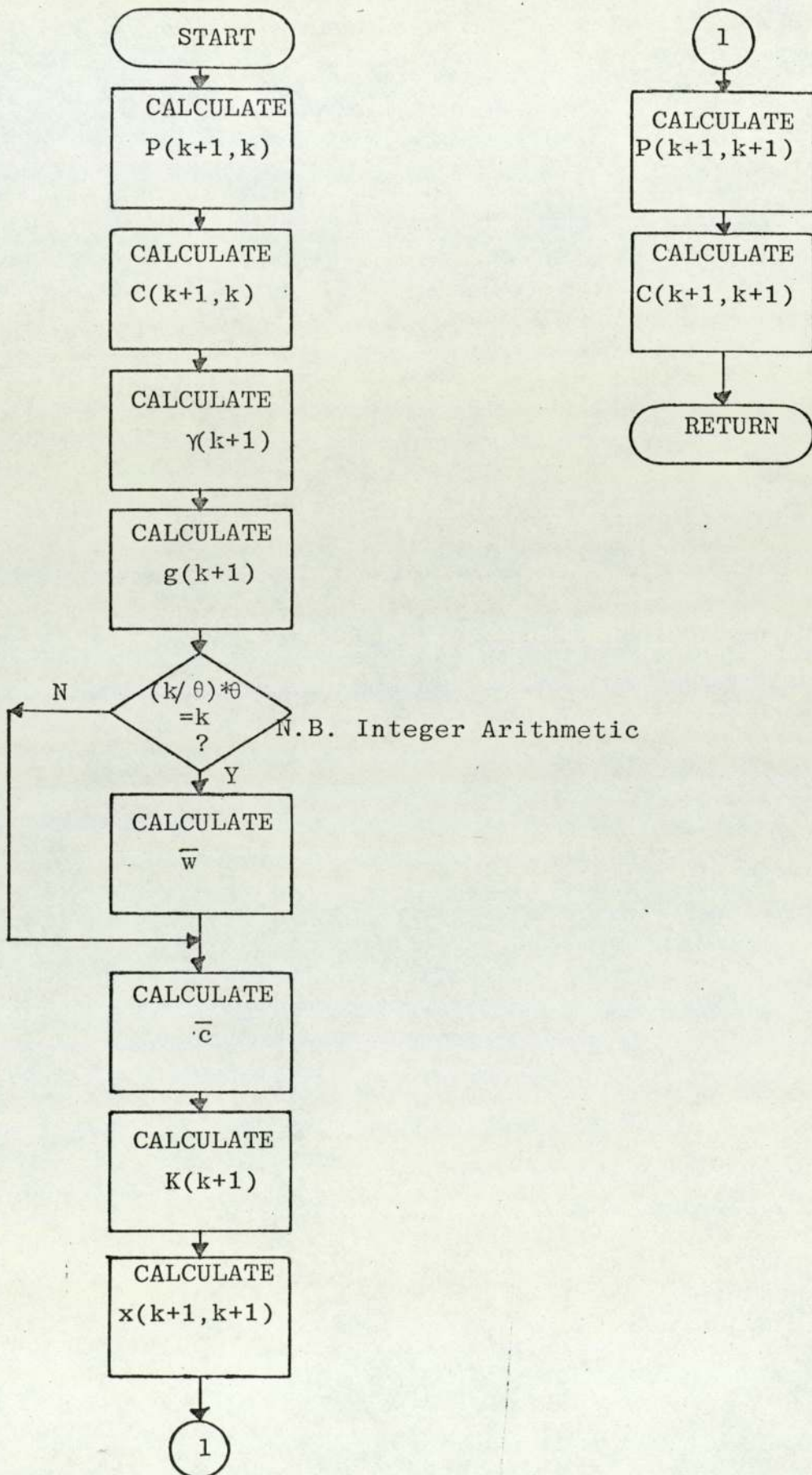
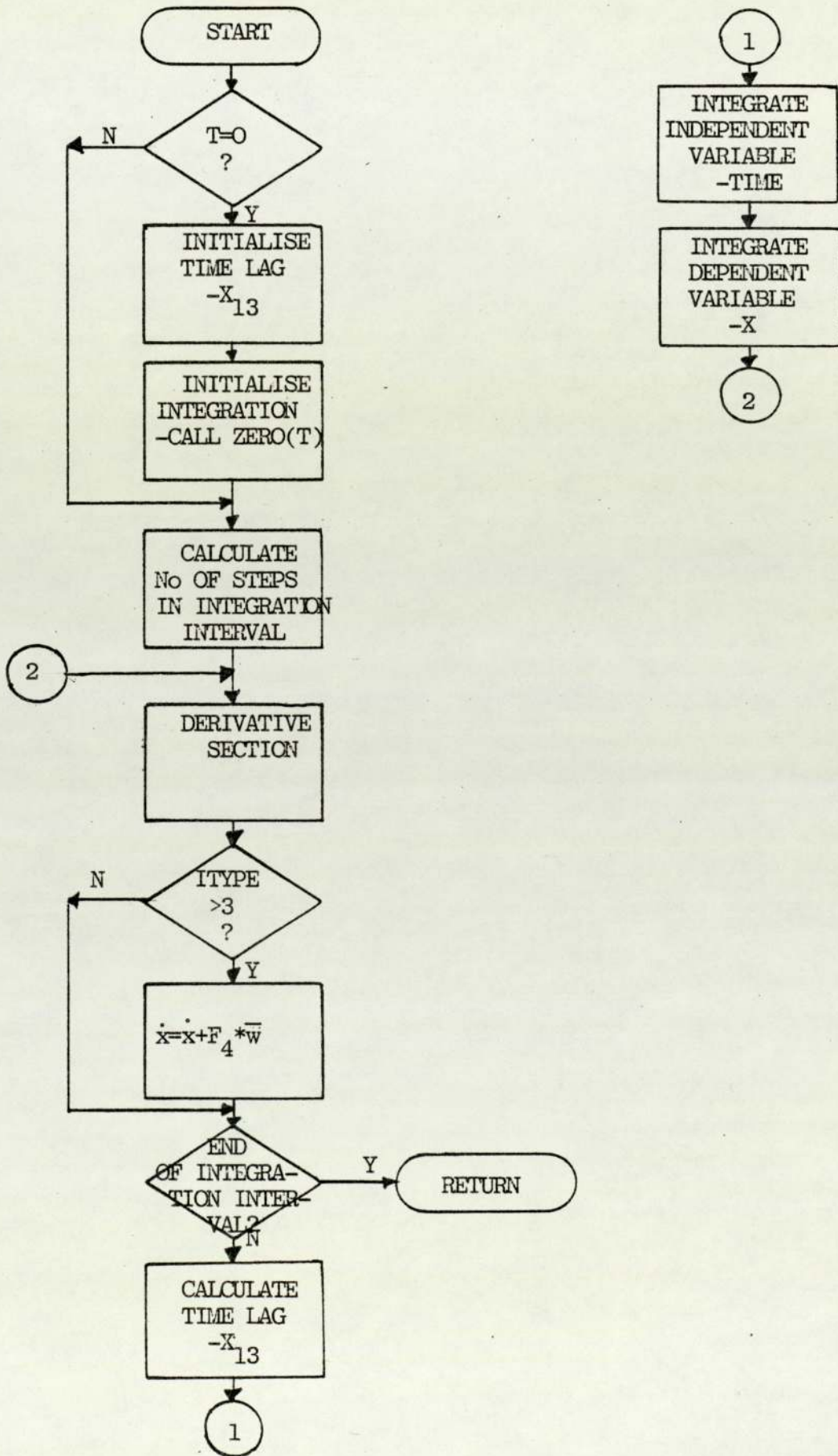


FIGURE 4.7 - FLOWCHART OF SUBROUTINE PRED2



predicted state vector, $x(k+1,k)$, is calculated from the estimated state, $x(k,k)$, by the Runge-Kutta fourth order method of numerical integration. As shown by the flowchart, the integration is performed by CALLS on the subroutines ZERO, PRNPF, INTI and INTX.

4.4.2.4. THE STATE TRANSITION SEGMENTS - TRANS1 and TRANS2

Subroutine TRANS1 calculates the state transition matrix, $\phi(k+1,k)$, using a truncated Taylor series. This method, which was described in detail in section 3.3, requires the calculation of the Jacobian of partial derivations for the mathematical model being used. For the system described by the model

$$\dot{x} = f(x) \quad - (4.24)$$

the Jacobian of partial derivations is defined as,

$$\frac{\partial f(x)}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad - (4.25)$$

The state transition matrix can now be calculated as follows,

$$\phi(k+1,k) = I + \frac{\partial f(x)}{\partial x} \Big|_{x(k,k)} \cdot \Delta t \quad - (4.26)$$

Thus, in this example $\phi(k+1,k)$ can be calculated at each cycle of the filter by setting all elements of the matrix THY to zero except for those defined below:

$$\text{THY } (1,1) = \frac{-1}{\tau_{n1}} * \Delta t + 1 \quad - (4.27)$$

$$\text{THY } (1,7) = \frac{1}{\tau_{n1}} * \Delta t \quad - (4.28)$$

$$\text{THY } (2,1) = \frac{-x_2}{V_1} * \Delta t \quad - (4.29)$$

$$\text{THY } (2,2) = \frac{-x_1}{V_1} * \Delta t + 1 \quad - (4.30)$$

$$\text{THY } (2,7) = \frac{x_8}{V_1} * \Delta t \quad - (4.31)$$

$$\text{THY } (2,8) = \frac{x_7}{V_1} * \Delta t \quad - (4.32)$$

$$\text{THY } (3,1) = \frac{1}{\tau_{n2}} * \Delta t \quad - (4.33)$$

$$\text{THY } (3,3) = \frac{-1}{\tau_{n2}} * \Delta t + 1 \quad - (4.34)$$

$$\text{THY } (3,9) = \frac{1}{\tau_{n2}} * \Delta t \quad - (4.35)$$

$$\text{THY } (4,1) = \frac{x_2}{V_2} * \Delta t \quad - (4.36)$$

$$\text{THY } (4,2) = \frac{x_1}{V_2} * \Delta t \quad - (4.37)$$

$$\text{THY } (4,3) = \frac{-x_4}{V_2} * \Delta t \quad - (4.38)$$

$$\text{THY } (4,4) = \frac{-x_3}{V_2} * \Delta t + 1 \quad - (4.39)$$

$$\text{THY } (4,9) = \frac{x_{10}}{V_2} * \Delta t \quad - (4.40)$$

$$\text{THY (4,10)} = \frac{x_9}{v_2} * \Delta t \quad - (4.41)$$

$$\text{THY (5,3)} = \frac{1}{\tau_{n3}} * \Delta t + 1 \quad - (4.42)$$

$$\text{THY (5,5)} = \frac{-1}{\tau_{n3}} * \Delta t \quad - (4.43)$$

$$\text{THY (5,11)} = \frac{1}{\tau_{n3}} * \Delta t \quad - (4.44)$$

$$\text{THY (6,3)} = \frac{x_{4(t-T)}}{v_3} * \Delta t \quad - (4.45)$$

$$\text{THY (6,4)} = \frac{x_3}{v_3} * \Delta t \quad - (4.46)$$

$$\text{THY (6,5)} = \frac{-x_6}{v_3} * \Delta t \quad - (4.47)$$

$$\text{THY (6,6)} = \frac{-x_5}{v_3} * \Delta t + 1 \quad - (4.48)$$

$$\text{THY (6,11)} = \frac{x_{12}}{v_3} * \Delta t \quad - (4.49)$$

$$\text{THY (6,12)} = \frac{x_{11}}{v_3} * \Delta t \quad - (4.50)$$

$$\text{THY (7,7)} = A_1 * \Delta t + 1 \quad - (4.51)$$

$$\text{THY (8,8)} = A_2 * \Delta t + 1 \quad - (4.52)$$

$$\text{THY (9,9)} = A_3 * \Delta t + 1 \quad - (4.53)$$

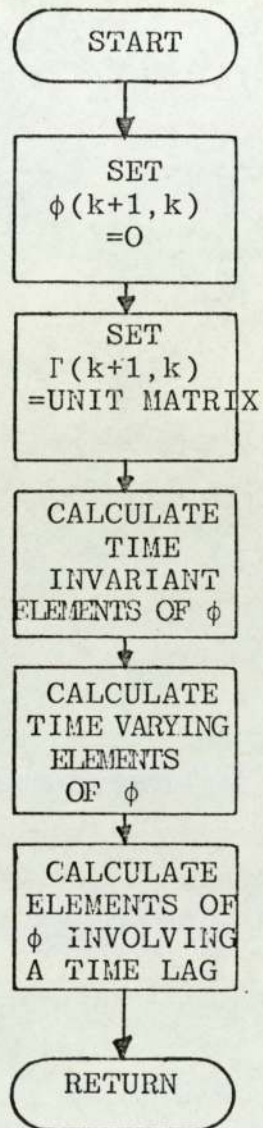
$$\text{THY (10,10)} = A_4 * \Delta t + 1 \quad - (4.54)$$

$$\text{THY (11,11)} = A_5 * \Delta t + 1 \quad - (4.55)$$

$$\text{THY (12,12)} = A_6 * \Delta t + 1 \quad - (4.56)$$

The flowchart for subroutine TRANS1 is shown in Figure 4.8.

FIGURE 4.8 - FLOWCHART OF SUBROUTINE TRANS1

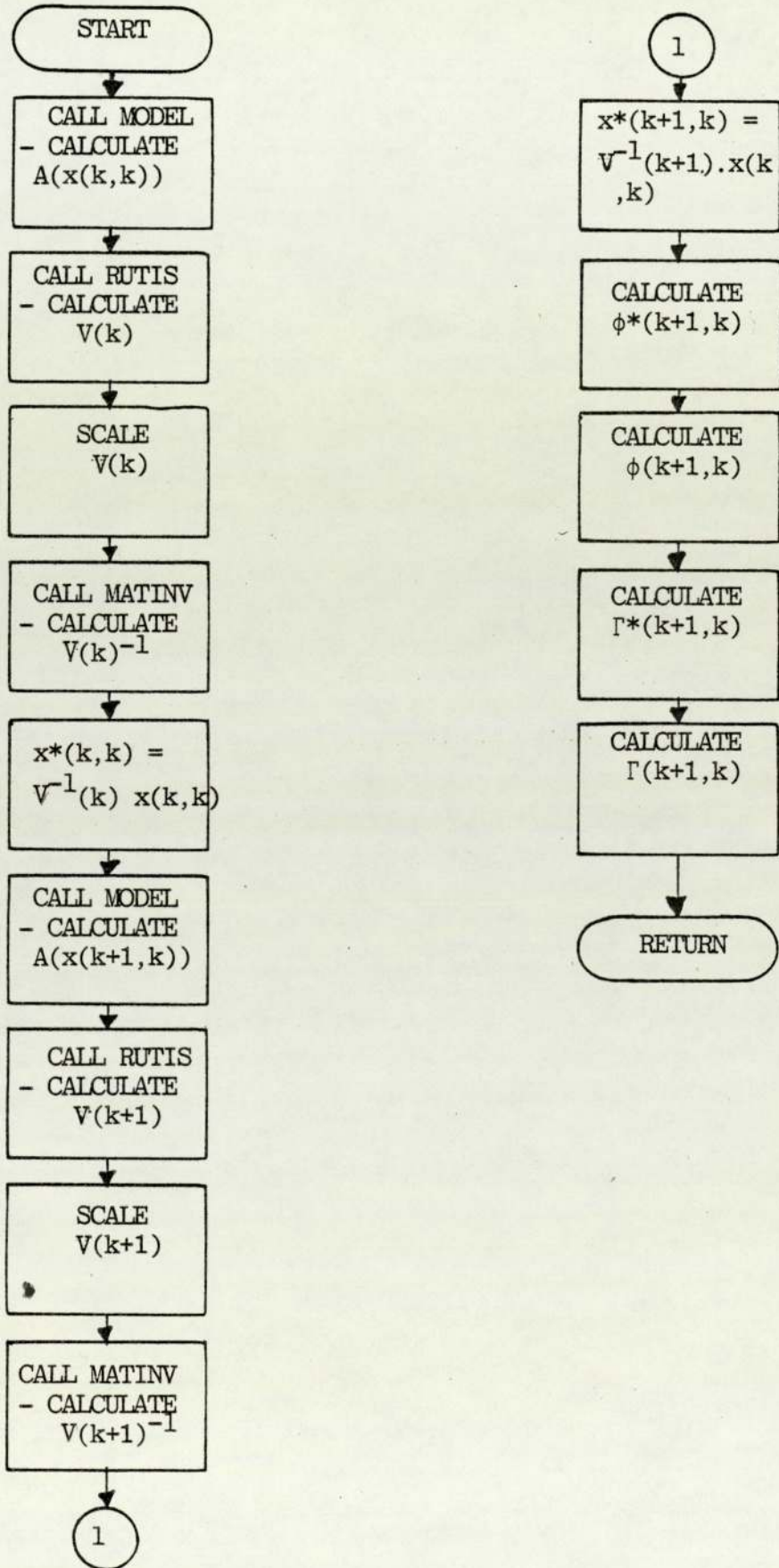


Calculation of the state transition matrix by the method of canonisation is rather more sophisticated than the method described above and consequently subroutine TRANS2 is not only more complicated but also requires more computer time for its execution. The subroutine is probably best understood by comparing the flowchart given in Figure 4.9 and the calculation procedure shown schematically in Figure 3.1. As can be seen from the flowchart TRANS2 calls subroutine MODEL to calculate the coefficient matrix $A(x)$ and then calls subroutine RUTIS to find the eigenvalues and eigenvectors. One of the advantages of using this method to calculate $\phi(k+1,k)$ is that once the transformed state transition matrix, $\phi^*(k+1,k)$, has been calculated, the integral state transition matrix, $\Gamma(k+1,k)$, can be computed with very little extra effort.

4.4.2.5. SUBROUTINE MODEL

In the earlier discussion on the calculation of the state transition matrix by the method of canonisation it was stated that in many cases it is possible to rearrange non-linear models into the form,

FIGURE 4.9 - FLOWCHART OF SUBROUTINE TRANS2



$$\dot{x}(k) = A(k) \cdot x(k) \quad - (4.57)$$

In this application of the Kalman Filter such a rearrangement is possible and consequently when required, A(k) is calculated by subroutine MODEL. The method used is to set all elements of the matrix A to zero except those defined below.

$$A(1,1) = \frac{-1}{\tau_{n1}} \quad - (4.58)$$

$$A(1,7) = \frac{1}{\tau_{n1}} \quad - (4.59)$$

$$A(2,2) = \frac{x_1}{V_1} \quad - (4.60)$$

$$A(2,8) = \frac{x_7}{V_1} \quad - (4.61)$$

$$A(3,1) = \frac{1}{\tau_{n2}} \quad - (4.62)$$

$$A(3,3) = \frac{-1}{\tau_{n2}} \quad - (4.63)$$

$$A(3,9) = \frac{1}{\tau_{n2}} \quad - (4.64)$$

$$A(4,2) = \frac{x_1}{V_2} \quad - (4.65)$$

$$A(4,4) = \frac{-x_3}{V_2} \quad - (4.66)$$

$$A(4,10) = \frac{x_9}{V_2} \quad - (4.67)$$

$$A(5,3) = \frac{1}{\tau_{n3}} \quad - (4.68)$$

$$A(5,5) = \frac{-1}{\tau_{n3}} \quad - (4.69)$$

$$A(5,11) = \frac{1}{\tau_{n3}} \quad - (4.70)$$

$$A(6,4) = \frac{x_4(t-T)}{x_4(t)} * \frac{x_3}{V_3} \quad - (4.71)$$

$$A(6,6) = \frac{-x_5}{V_3} \quad - (4.72)$$

$$A(6,12) = \frac{x_{11}}{V_3} \quad - (4.73)$$

$$A(7,7) = A_1 * (x_7 - \bar{x}_7) * \left(\frac{1}{x_7}\right) \quad - (4.74)$$

$$A(8,8) = A_2 * (x_8 - \bar{x}_8) * \left(\frac{1}{x_8}\right) \quad - (4.75)$$

$$A(9,9) = A_3 * (x_9 - \bar{x}_9) * \left(\frac{1}{x_9}\right) \quad - (4.76)$$

$$A(10,10) = A_4 * (x_{10} - \bar{x}_{10}) * \left(\frac{1}{x_{10}}\right) \quad - (4.77)$$

$$A(11,11) = A_5 * (x_{11} - \bar{x}_{11}) * \left(\frac{1}{x_{11}}\right) \quad - (4.78)$$

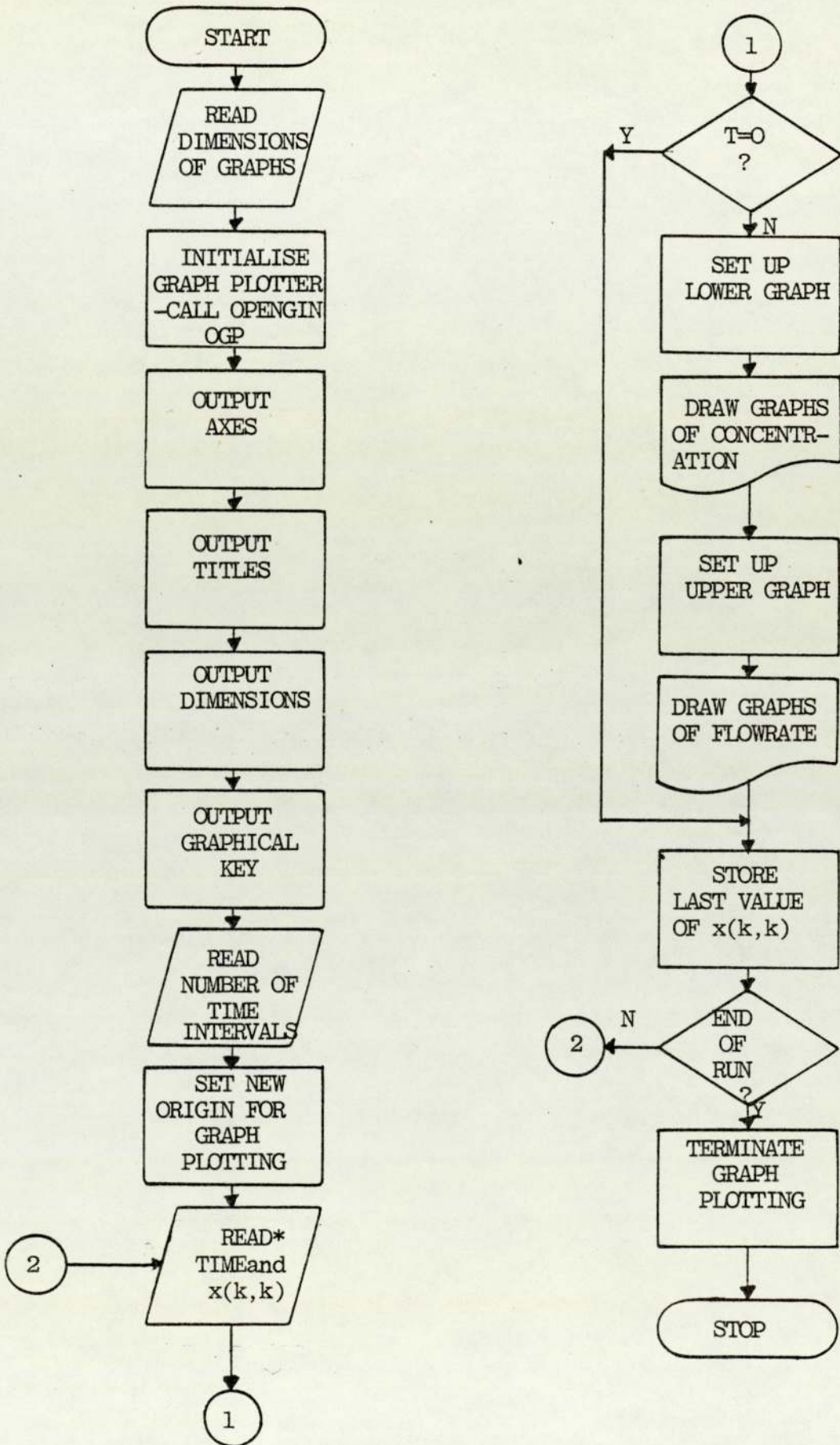
$$A(12,12) = A_6 * (x_{12} - \bar{x}_{12}) * \left(\frac{1}{x_{12}}\right) \quad - (4.79)$$

4.4.5. THE GRAPHPLOTTING PROGRAM - GRAPHPLOT

The problem of how to display the results obtained to the best advantage is one which is common to many forms of technical research. It is now widely accepted that in many cases the best method to use is some kind of graphical display. However, as is usually the case the best method is not always the easiest one to implement and in the case of graph plotting this has certainly

been the situation for some time. Fortunately the advent of both V.D.U.'s with graphical capabilities and high speed graph plotters, either of which can be linked to a digital computer, has considerably alleviated the problem. Therefore, it was decided to write a program which could plot all of the results obtained on the ICL 1934 graph plotter. Figure 4.10 shows the flowchart of the GRAPHPLOT program which was written using the recently developed GINO-F software (4.8).

FIGURE 4.10 - FLOWCHART OF GRAPHPLOT



*This input is from GRAPHDATA

4.5. RESULTS AND DISCUSSION

Once the computer programs discussed in the previous section had been developed and debugged, it was necessary to decide upon a suitable series of experiments to test the optimality and convergence of the five types of filter. The strategy adopted was to first of all generate a set of noisy measurements using SIMULPROG and then use this data in a series of experiments to tune all of the filters. In these initial experiments the correct mathematical model was used in order to simplify the optimisation of the various parameters and starting conditions. The best performance of each filter was determined by examination of the errors in the state estimates which are calculated as follows:

(i) The error in each state estimate at each cycle of the filter is calculated using,

$$\xi(k) = \left| 100 * (x(k,k) - x(k)) * \left(\frac{1}{x(k)} \right) \right| - (4.80)$$

where, $x(k)$ is the correct value of the state vector at time t_k .

(ii) The cumulative error, $\sum_{i=1}^k \xi(i)$, for the run is calculated,

(iii) At the end of the run the average error for each state variable, $\left(\frac{1}{NC} \right) * \sum_{i=1}^{NC} \xi(i)$, is calculated: NC is the total number of filter cycles.

(iv) The mean error for the experiment is calculated by summing the average errors for each state variable and dividing by the degree of the state vector.

When the filter tuning experiments were complete there followed three sets of experiments to determine the performance of each type of filter in the presence of noisy measurements, incorrect starting conditions and finally a poor mathematical model.

4.5.1. SIMULATION RESULTS

In order that suitable measurement data could be provided for the filtering experiments two runs of SIMULPROG were carried out. In the first run the three tank blending system is initially in a period of transition towards a steady state, which is almost reached after 10 hours. The second run begins in the same way but after 5 hours the mean values of both of the inputs to the first tank, \bar{x}_7 and \bar{x}_8 , are modified to produce additional transient behaviour.

The modifications made are,

\bar{x}_7 - decreased by 1.0

\bar{x}_8 - increased by 0.09

One of the features of both of these runs is that whereas the flowrates of the system reach steady values after approximately two hours, the concentrations of the system do not begin to approach steady values until the end of the simulation experiments.

The measurements produced in both of these runs of SIMULPROG were corrupted by the addition of Gaussian noise with the following statistics:

Flowrates - $\mu = 0, \sigma = 0.1$

Concentrations - $\mu = 0, \sigma = 0.01$

Thus, the measurements were very noisy.

4.5.2. FILTER TUNING EXPERIMENTS

Using the results produced by the first run of SIMULPROG a series of experiments were carried out to optimise the performance of each filter. Since the experiments for each filter were carried out separately and in view of the fact that the features to be considered are different in each case, it is convenient to discuss the filters individually.

TYPE 1

It was never expected that the results produced by this filter would be at all acceptable and this fear was confirmed by the early experiments. The results obtained can only be regarded as the

worst examples of many of the problems associated with the Extended Kalman Filter. However, the performance of this type of filter can be regarded as one which all of the others should improve upon and is thus useful in determining which filter, if any, provides the best estimates. It was found that the errors in the estimates were minimised when,

$$R(k) = 1.0 * I , Q(k) = 1.5 * I.$$

TYPE 2

The use of numerical integration to compute the estimates considerably improves the optimality and convergence of the Extended Kalman Filter. The tuning experiments carried out with this filter showed that the most accurate estimates were obtained when,

$$R(k) = 1.0 * I , Q(k) = 1.5 * I.$$

TYPE 3

The optimisation of this filter took longer than the others due to a number of numerical problems associated with the computation of the inverse eigenvector matrix, V^{-1} . It soon became apparent that the eigenvector matrix was becoming singular as the three tank blending system

approached a steady state. This is due to the way in which the inputs to the system are modelled. Examination of equation 4.1 reveals the fact that as x_j approaches \bar{x}_j the related coefficient in the $A(x)$ matrix, see equations 4.74 to 4.79, tends towards zero. As a result of this the eigenvector matrix will contain some small diagonal elements and singularity is to be expected. The problem was overcome by multiplying the eigenvector matrix by a suitable scalar prior to its inversion. This action in no way affects the computation of $\phi(k+1,k)$ as the scaling factor eventually cancels out.

Following the solution of these numerical problems it was found that the errors in the estimates were minimised when,

$$R(k) = 1.0 * I \quad , \quad Q(k) = 0.5 * I.$$

TYPE 4

The procedure for tuning the adaptive filters is quite different to that followed for the previous filters as more parameters and matrices need to be determined. To begin with the matrices $R(k)$ and $Q(k)$ were set to the same values as those used for the Extended Kalman Filter (type 2) and the parameters α, β and θ were given the following values,

$\alpha = \frac{1}{k}$, where k = the number of filter cycles,
until $\alpha = 0.2$ when it remains constant.

$$\beta = 0.3$$

$$\theta = 4$$

The next step in the tuning process was to determine the matrix F_4 . The procedure followed was as discussed in section 3.2.2.1., e.g. in determining the first column of F_4 the following values were tried,

(i) All elements zero except $F_4(1,1)$ which equals 1,

(ii) All elements zero except $F_4(7,1)$ which equals 1.

The reason for choosing these particular elements to have non-zero elements was that since the first measurement is of x_1 then it can be seen from equation 4.2 that the only state variables involved are x_1 and x_7 . In all twelve different values of F_4 were tried before deciding on the following matrix,

$$F_4(i,j) = 0 \quad , \quad i = 1,n \\ j = 1,m$$

except for,

$$F_4(7,1), F_4(8,2), F_4(5,3), F_4(6,4), F_4(10,5), \\ F_4(11,6),$$

all of which were assigned the value 1.

One of the features of the Adaptive Filter which cannot be backed up by theoretical considerations is the way in which the matrix F_4 is determined when there is no information available about the likely accuracy of the mathematical model being used. However, the results obtained in the experiments discussed above do lend weight to the procedure used. For example, in determining column 2 of F_4 the non-zero elements tried were $F_4(1,2)$, $F_4(2,2)$, and $F_4(8,2)$ and the values of $g(k)$ obtained were,

0.8058, 0.8361 and 0.1754

respectively. Thus the obvious choice for this column of F_4 is the one where element 8 contains one. This behaviour was observed throughout the entire procedure for finding F_4 and it seems to indicate that normally a local minimum does exist. One further observation made whilst finding F_4 was that it may not necessarily have the maximum number of columns, i.e. the matrix is $n \times l$ and l may be less than or equal to m . This observation was made whilst determining the sixth column of F_4 when it was found that the improvement in both the accuracy of the estimates and the minimisation of $g(k)$ were minimal when this column was added to the matrix.

Attention was now focussed on the parameters α , β and θ . Each of these parameters was varied within the ranges given in sections 3.2.2.2. and 3.2.2.3. and because only a very small difference in filter performance was observed, these parameters were left at their initial values for the remaining experiments.

Finally the matrices $R(k)$ and $Q(k)$ were varied and it was noticed that far more weight is placed on the measurements by the Adaptive Filter. Thus, it was found that the following values of $R(k)$ and $Q(k)$ minimised the error in the estimates,

$$R(k) = 2.25 * I \quad , \quad Q(k) = 0.1 * I .$$

TYPE 5

This filter is essentially a hybrid version of types 3 and 4 and as such is somewhat of an unknown quantity. The parameters α , β and θ were assigned the same values as those used previously and although F_4 was redetermined, it was found to have exactly the same value as that found for the fourth type of filter. However, the effects of a different kind of state transition matrix did display themselves in the values chosen for $R(k)$ and $Q(k)$, which were,

$$R(k) = 2.0 * I \quad , \quad Q(k) = 0.5 * I .$$

4.5.3. FILTERING RESULTS

Applying the principle that one learns to walk before attempting to run, the first set of experiments, RUN 1, is merely a test of the smoothing capabilities of each type of filter, i.e. one is testing the efficiency of each filter in removing measurement noise. The next set of experiments, RUN 2, is a test of how rapidly the estimates generated by each type of filter will converge on the actual state when the initial value of the state vector, $x(0,0)$, is incorrect, the measurements used in this set of experiments are the same as those used in RUN 1. The most significant test of filter performance was carried out in the final set of experiments, RUN 3. In RUN 3, not only are the measurements noisy and the value of $x(0,0)$ incorrect, but also the mathematical model incorporated in each filter is a poor one. There are many instances in the literature when reference is made to a poor mathematical model without ever quantifying what is meant by such a statement. This raises the question, what makes a poor model? In referring to RUN 3 this statement should be taken to mean that not only are some of the parameters incorrect, but also a change in state takes place which can

in no way be predicted by the mathematical model incorporated in the filters. Thus, this change, details of which are given in section 4.5.1, can only be detected through the measurements supplied to the filter at each sampling time. RUNS 1, 2 and 3 are summarised in table 4.2. The graphs of the estimates generated in RUN 3 display all of the features of all three sets of experiments and are included in Appendix A.

4.5.4. DISCUSSION OF FILTERING RESULTS

The results given in Table 4.3 and Appendix A show quite clearly that the inclusion of a strategy to promote and ensure the convergence of the Extended Kalman Filter does produce a considerable improvement in the accuracy of the estimates generated. It is however, necessary at this point to consider the advantages and disadvantages of the various filters tested.

4.5.4.1. FILTERING OF MEASUREMENT NOISE

When both the mathematical models of a process and the initial state estimates are known accurately then the task of state estimation reduces to one of filtering off measurement and process noise. Thus, the results of RUN 1 show the capability

TABLE 4.2 - SUMMARY OF RUNS 1, 2 AND 3

	R U N 1	R U N 2	R U N 3
MEASUREMENTS	Noisy - $\mu = 0$ $\sigma = 0.1, 0.01$ Simulprog run 1	Noisy - $\mu = 0$ $\sigma = 0.1, 0.01$ Simulprog run 1	Noisy - $\mu = 0$ $\sigma = 0.1, 0.01$ Simulprog run 2
INITIAL CONDITIONS - $x(0,0)$	Correct see table 4.1	Incorrect $x_1=4.00$ $x_7=5.1$ $x_2=0.40$ $x_8=0.48$ $x_3=8.00$ $x_9=1.85$ $x_4=0.49$ $x_{10}=0.33$ $x_5=8.00$ $x_{11}=3.20$ $x_6=0.30$ $x_{12}=0.28$	Incorrect see run 2
MATHEMATICAL MODEL	Correct see table 4.1	Correct see table 4.1	Incorrect $\tau_{n3} = 0.34$ $v_1 = 9.00$ $v_3 = 28.0$ $x_2 = 0.90$ $x_5 = 0.90$

TABLE 4.3 - AVERAGE ESTIMATION ERRORS

RUN NUMBER TYPE OF FILTER	1	2	3
1	9.070	9.699	10.269
2	1.815	2.985	6.098
3	0.471	1.418	2.567
4	0.612	1.875	2.429
5	0.673	1.593	3.197

TABLE 4.4 - COMPUTATIONAL REQUIREMENTS

TYPE OF FILTER	EXECUTION TIME (MIL)	COMPUTER MEMORY (relative to type 2)
1	76	-10%
2	82	0
3	117(86)	+25% (+5%)
4	135	+15%
5	162	+40%

of each filter to smooth the state estimates and as shown in Table 4.3, with the exception of Type 1, all of the filters perform reasonably well. The first type of filter exhibits divergence not because of the noisy measurements but because of the inaccurate predictions computed. This fact clearly demonstrates that the use of a truncated Taylor series expansion is not a particularly accurate way of calculating $\phi(k+1,k)$. As a result of this, it is not surprising that the best performance is given by the third type of filter. However, this result is far from conclusive as the Adaptive filters (types 4 and 5) produce comparable performances and indeed the second type of filter produces estimates which are quite acceptable.

One general point which emerges from this set of experiments is that the filters have been tuned correctly since the estimates computed are affected by both the predictions and the measurements. This might seem to be an unnecessary observation but there have in the past been instances reported in the literature, see for example Goldman (4.9), where the matrices $Q(k)$ and $R(k)$ are selected such that the measurement vector, $y(k)$ has little affect on the estimates.

4.5.4.2. CONVERGENCE FROM INCORRECT INITIAL ESTIMATES

The different characteristics of the filters began to emerge in RUN 2 of this investigation. The first type of filter initially shows some signs of convergence, but the estimates very quickly become biased as in RUN 1. The main difference between the second and third types of filter in this run is the rate at which the two estimates of x_4 converge on the actual state. Since the concentration x_4 is not only represented by a non-linear differential equation but also unmeasured it is significant that the third type of filter converges more rapidly as this confirms the greater accuracy of the canonisation method for calculating the state transition matrix. In general all of the filters converge rapidly on flowrates but the third and fifth types of filter are more effective in their convergence on concentrations and in particular on x_4 .

It was in this set of experiments that doubts first began to arise about the optimality and convergence of the fifth type of filter. One of the features of this filter which became apparent in this run was that although the estimates converged, the diagonal elements of the error covariance matrix, $P(k,k)$, associated with

unmeasured variables became unusually large. As will be discussed later there are no general theoretical results which can be used to predict the observability of non-linear systems but a fairly reliable indication that a system is unobservable is that the estimates oscillate about the actual state and the related diagonal elements of $P(k,k)$ become rather large. Thus, the fifth type of filter appears to converge but at the same time displays one of the features of unobservability. In an attempt to overcome this problem the matrices $P(0,0)$, $R(k)$, $Q(k)$ and the parameters α , β and θ were all varied but this approach met with little success. It was therefore decided that as none of the other filters displayed this behaviour it must be due to the design of the filter itself.

4.5.4.3. POOR MATHEMATICAL MODELS

The most significant results obtained in this investigation were those produced by the third set of experiments. The first and perhaps the most important observation to be made is that both the first and the second types of filter fail to detect the changes caused by the modifications made to \bar{x}_7 and \bar{x}_8 after 5 hours. This is caused

by the filter gain matrix $K(k)$ becoming too small for the measurements to have a significant affect on the estimates. All of the filters initially have quite large filter gains but as the filters converge the matrix $P(k+1,k)$ becomes smaller and as a result $K(k)$ becomes small : see equation 2.12. Thus, when changes are made which are not predicted by the mathematical model the first two filters have learned the current state too well and the estimates generated diverge from the actual state of the process. One technique which could be used to combat this problem is to increase the value of $Q(k)$ and decrease the value of $R(k)$. However, as was discussed earlier, this has the effect of making the estimates noisier and causing the value of $P(k,k)$ to become unduly large.

The third, fourth and fifth types of filter do detect this change in state but the methods of detection are somewhat different. The fourth type of filter monitors the change by making use of its model error compensation strategy to correct the values of the derivatives produced by the mathematical model. To illustrate this point let us compare the value of \bar{w} at the end of RUN 2 with that at the end of RUN 3.

RUN 2 - 0.184, 0.067, 0.560, 0.019, -0.007, -0.194

RUN 3 - -1.469, 0.276, -0.680, 0.023, -0.008, -0.198

As can be seen the first two elements of \bar{w} are much larger for RUN 3 and due to the structure of F_4 this means that differential equations 4.8 and 4.9 are being corrected. Thus, the modelling error committed by changing \bar{x}_7 and \bar{x}_8 is being compensated for. In addition to this as the model error vector increases in size so does P_w and thus $P(k+1,k)$ becomes larger : see equation 3.36. In turn this causes the filter gain to increase (see equation 3.45) and so more attention is given to new measurements.

The reason the third type of filter is able to detect the change is that the filter gain matrix has a different structure due to the way in which $\phi(k+1,k)$ is calculated. The different structure means that flowrates can only be affected by other measured flowrates and similarly for concentrations. For filters with state transition matrices calculated using a truncated Taylor series this is not the case and each state variable can be affected by measurements of both concentration and flowrate. For the third type of filter this different structure together with the fact that the elements of $K(k)$ are in general larger means that it is able to react quickly to

new measurement data.

One other interesting observation which was made regarding the different way in which the filters react to the change in \bar{x}_7 and \bar{x}_8 is that the third type of filter compensates by changing x_8 and x_9 whereas the fourth filter is more correct in changing x_7 and x_8 .

In general the results of RUN 3 tend to indicate that all of the filters except the first can cope with the inaccuracies caused by incorrect model parameters but only the third, fourth and fifth types can detect changes which can not be predicted by the mathematical model.

4.5.4.4. COMPUTATIONAL REQUIREMENTS

Having discussed the superior performance of the filters which include the theoretical developments proposed in chapter three, it is now time to review the disadvantages of these modifications in terms of their increased computational requirements. As can be seen from Table 4.4 the implementation of the fifth type of filter requires an excessive increase both in the necessary computer memory and the execution time. This fact alone constitutes a sufficient reason that this filter should not be

considered for an on-line application and when the doubts about its properties of convergence and optimality had been reviewed it was decided that this filter had too many disadvantages to be regarded as an improvement over the Extended Kalman Filter. It is rather ironic that this should be so in the case of what initially was regarded as the most sophisticated of all of the five types of filter.

The increase in computing power required by the third type of filter led to an investigation into ways of decreasing both the execution time and the computer memory required. The simplifications considered were, (i) assume the eigenvector matrix to be constant throughout the run, (ii) assume the eigenvalues are the same as the diagonal elements of the coefficients matrix, $A(x)$. The second simplification was one which resulted from the observation that the eigenvalues rarely differed from the diagonal elements of $A(x)$ except in the fourth significant digit. However, this assumption only produced a marginal improvement in the computational requirements of this type of filter. The first simplification did, however, lead to a significant decrease in both execution time and required computer memory and, as can be seen by the figures in parenthesis in Table 4.4.,

the third type of filter now requires only 5% more computing power than the second type of filter. The eigenvector matrix chosen for this simplified filter was the one obtained halfway through the first set of experiments and in RUN 3 this produced an increase in estimation error of 0.056

(i.e. from 2.367 to 2.423)

when compared with the complete version of the filter. Clearly the choice of eigenvector matrix will influence the performance of this simplified filter quite considerably and although it is fairly easy to find the matrix which produces the best performance in off-line experiments, it may be more difficult in an on-line situation.

In general both the third and fourth types of filter require greater computing power but it is felt that this disadvantage is outweighed by the superior performance obtained.

4.5.4.5. OBSERVABILITY

The concept of observability in non-linear systems is one which has been considered by many researchers in this field and although a number of them, see for example Coggan and Noton(4.1),

have attempted to quantify observability, there are as yet no general theoretical results which can be used to predict the observability of such systems. It is felt, however, that when a system is unobservable this fact will be displayed by the results obtained in filtering experiments. The behaviour characteristic of unobservability is that some of the estimates oscillate about the true states and the related diagonal elements of the error covariance matrix, $P(k,k)$, become unusually large. During this investigation two experiments were carried out to try and establish the observability of this process. In the first experiment the state vector was extended to include C'_2 , x_{13} , and as a result both x_{13} and x_6 appear to be unobservable (see Appendix A-15). This is quite interesting since it is not clear whether or not C'_2 can be considered as a state variable. Jacobs (4.10) states that the degree of the state vector, n , is determined by the number of state variables necessary to describe the dynamic behaviour of the system and that it is normally helpful to regard the n state variables as coordinate axes defining an n dimensional state space. From this definition it is clear that since C'_2 is in fact

C_2 delayed, then both x_4 and x_{13} relate to the same coordinate axis in state space. As a result of this x_{13} is redundant and it is therefore not surprising that it causes unobservability.

The second experiment carried out (Appendix A-16) was to see what happened if the degree of $y(k)$ was decreased to five by omitting the measurement of x_5 . The results show that x_5 and to a certain extent x_3 are now both unobservable.

A great number of additional experiments could have been carried out to see what other situations lead to the estimates becoming unobservable, but it was felt that the results already obtained demonstrate the system's observability. However, it is clear that great care is necessary when choosing the state variables and measurements of a process to be investigated.

4.5.5. CONCLUSIONS

The results obtained in this investigation have led to the following general conclusions being made about the filters tested:

(i) When the Extended Kalman Filter incorporates a poor model some strategy to ensure and promote convergence is required.

(ii) The strategies proposed in chapter three, i.e. types 3 and 4 filter, are strategies capable of improving the performance of the Extended Kalman Filter when it incorporates a poor model.

(iii) It is not as yet possible to combine both of these theoretical developments within the same filter, i.e. type 5.

(iv) The Adaptive filter is more correct in the way it compensates for a poor model.

(v) The matrix F_4 can be determined by a realistic number of trials using the Adaptive filter.

(vi) When the mathematical model of a process includes non-linearities, calculation of the state transition matrix, $\phi(k+1,k)$, by canonisation, i.e. type 3 filter, improves the performance of the Extended Kalman Filter.

(vii) The third type of filter can be considerably simplified without any great loss of accuracy.

(viii) Observability is an important concept to be borne in mind when deciding on which state variables to estimate and what measurements are necessary.

4.6. CHAPTER REVIEW

The simulation studies of the Kalman Filter reported in this chapter have shown that the theoretical developments proposed in chapter three improve the performance of the filter when the mathematical model used to represent the process is either a poor one or includes nonlinearities. The importance of the concept of observability has also been discussed.

The results of this investigation have shown that it is feasible to apply the second, third and fourth filters in an on-line situation.

CHAPTER 5

THE DOUBLE EFFECT EVAPORATOR/COMPUTER SYSTEM

5.1. INTRODUCTION

State estimation in a real time, on-line environment requires the linking of the chemical engineering process under study to a computer so that measurement data can be obtained and processed as required. Subsequently, information can be returned to the process either by additional hardware, i.e. on-line control, or by tele-printer messages to a process operator.

The chemical engineering process used in this research was a double effect evaporator using a water/steam system. The evaporator was linked by a Honeywell Analogue Digital Input Output System (HADIOS) to a Honeywell 316 Computer. The measurements taken from the evaporator were 'conditioned' insitu prior to transmission along cabling to the computer.

Figure 5.1 shows a schematic representation of the system.

THE DOUBLE EFFECT EVAPORATOR/COMPUTER SYSTEM

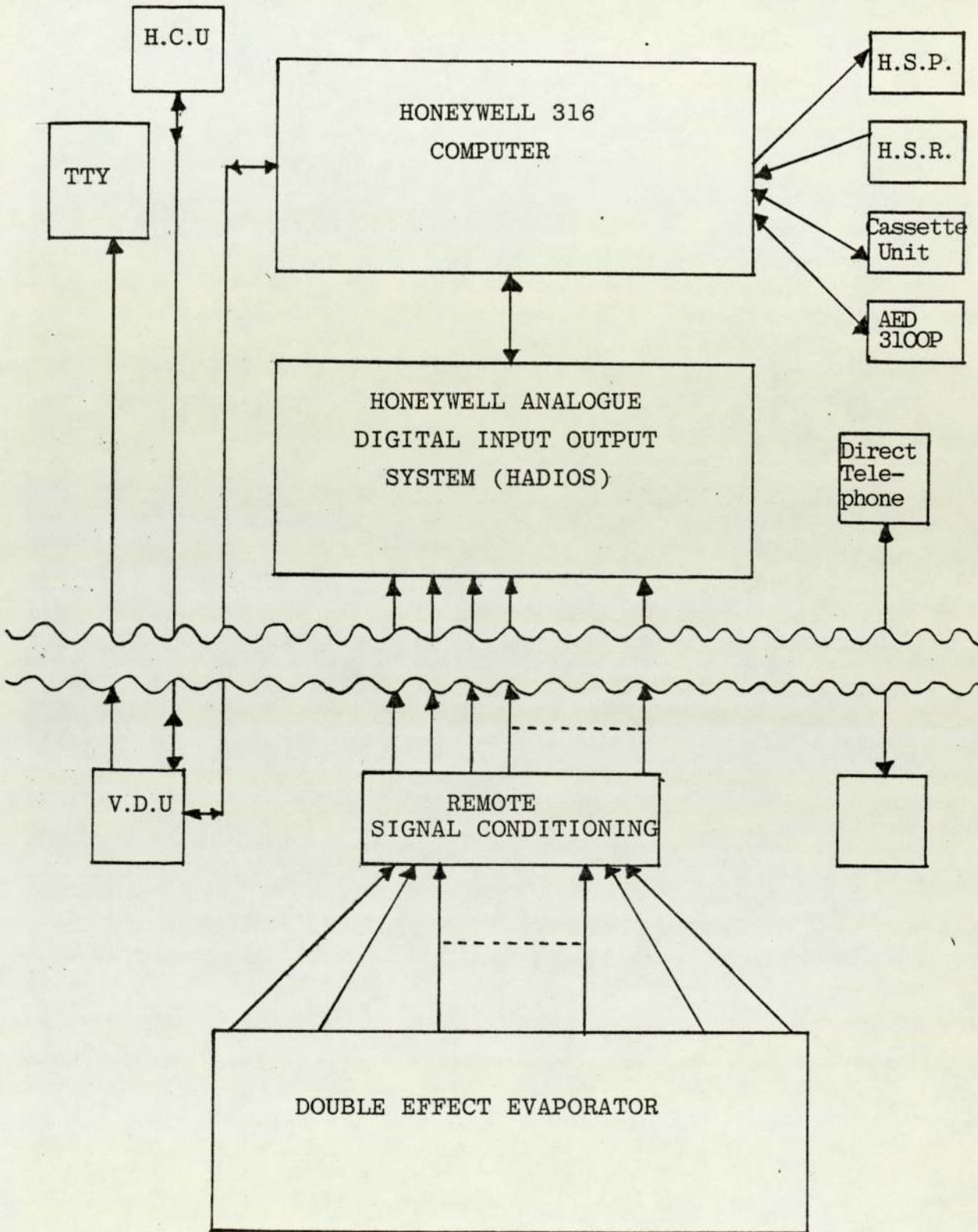


FIGURE 5.1

5.2. THE DOUBLE EFFECT EVAPORATOR

The double effect evaporator was manufactured and commissioned by the Kestner Evaporator and Engineering Company Limited. The plant was designed for operation either as a single or a double effect evaporator, with the additional option of running under vacuum conditions. For the purpose of this research the evaporator was operated as a double effect system under vacuum conditions. A detailed description of the engineering construction is provided in Table 5.1 and the original operating conditions suggested by the manufacturers are given in (3.1).

5.2.1. PROCESS DESCRIPTION

The evaporation process consists of two stages, a climbing film first effect and a forced circulation second effect. A schematic diagram of the evaporator is shown in Figure 5.2, with Plates 1 and 2 showing front and rear views of the plant.

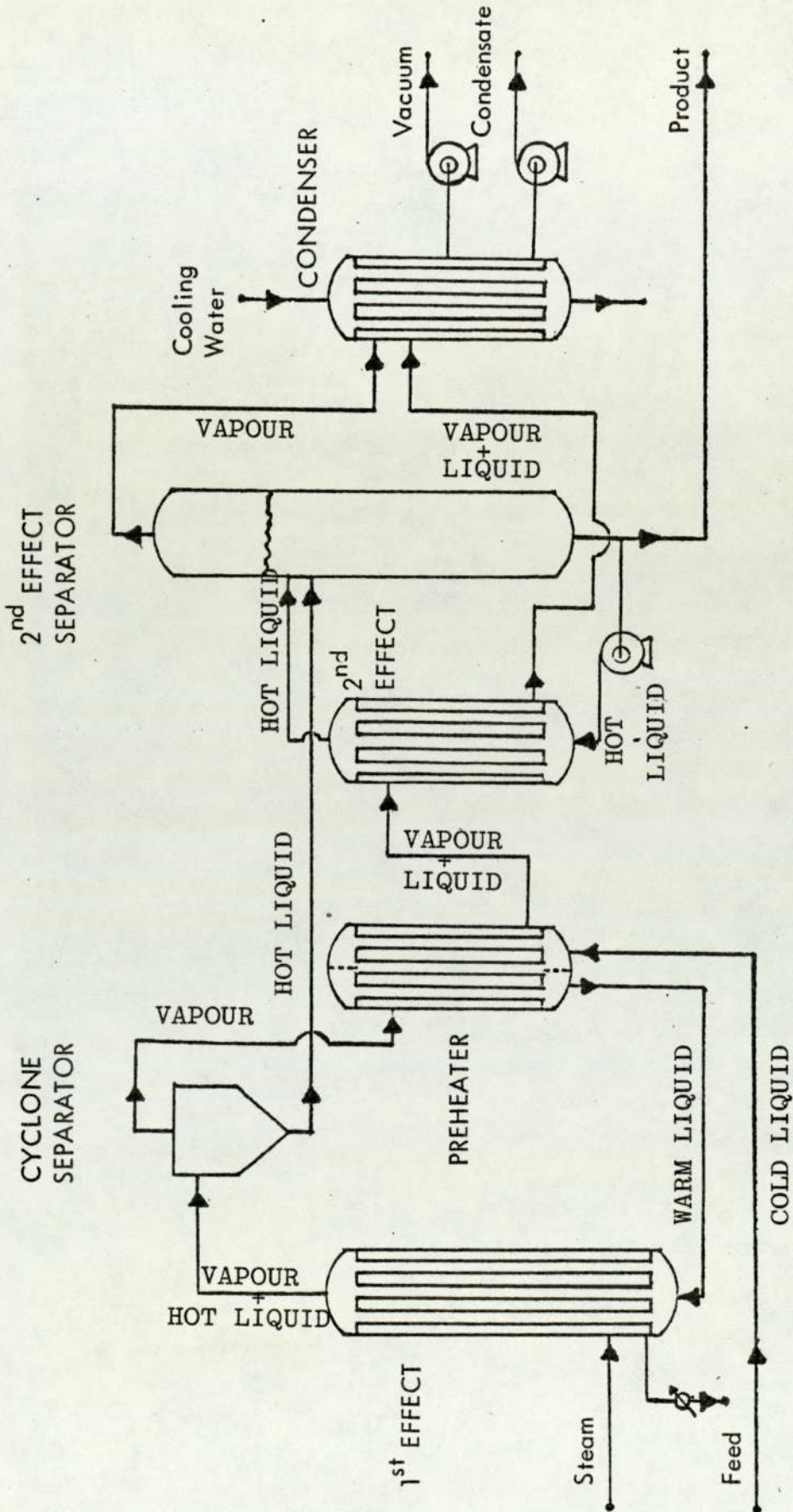
The liquid to be fed to the evaporator is first of all pumped by a centrifugal type pump to two header tanks situated eight metres above the plant. It is then gravity fed to the tubeside of the preheater unit where it is heated before passing

TABLE 5.1 - PHYSICAL DIMENSIONS OF THE DOUBLE EFFECT

EVAPORATOR HEAT EXCHANGERS

HEAT EXCHANGER DIMENSIONS	PREHEATER	FIRST EFFECT	SECOND EFFECT	CONDENSER
Exchanger Type	2:2	1:1	1:1	1:1
Number of Tubes	4	4	7	6
Tube Material	Copper	Copper	Copper	Copper
Tube Length (M)	1.416	2.590	1.065	2.197
Tube O.D. (MM)	22.2	22.2	28.6	28.7
Wall Thickness (MM)	1.65	3.20	3.20	3.30
Mass Tubes (kg)	4.9102	8.1922	8.9222	1.49496
Tubes Internal Vol. (M ³)	1.5933.10 ⁻³	2.9185.10 ⁻³	3.9468.10 ⁻³	6.6131.10 ⁻³
Shell Length (M)	1.397	2.590	1.065	2.197
Wall Thickness (MM)	3.40	4.76	6.81	4.76
Shell I.D. (MM)	101.6	101.6	152.4	155.7
Shell Volume (M ³)	7.7790.10 ⁻³	1.69384.10 ⁻²	1.51004.10 ⁻²	4.17654.10 ⁻²
Mass of Shell (kg)	22.3038	36.4427	13.4521	37.6069
Shell Material	Mild Steel	Mild Steel	Mild Steel	Mild Steel
Heat Transfer Area (M ²)	0.39555	0.72376	0.70632	1.18341

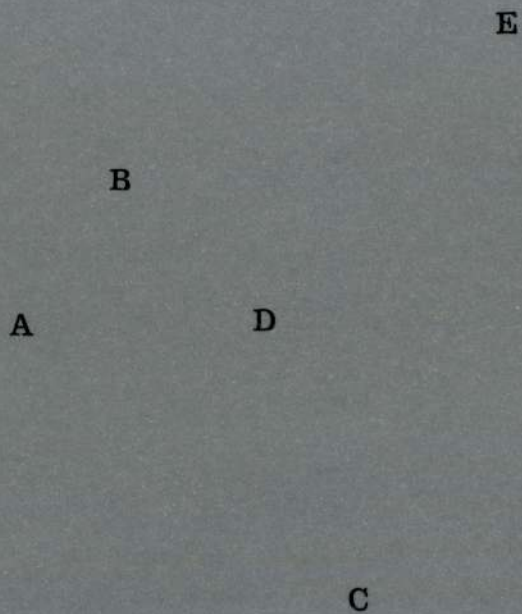
FIGURE 5.2



DOUBLE EFFECT EVAPORATOR

PROCESS DESCRIPTION

PLATE 1 - DOUBLE EFFECT EVAPORATOR (FRONT VIEW)



KEY A = First Effect
B = Preheater
C = Second Effect
D = Condenser
E = Second Effect Separator

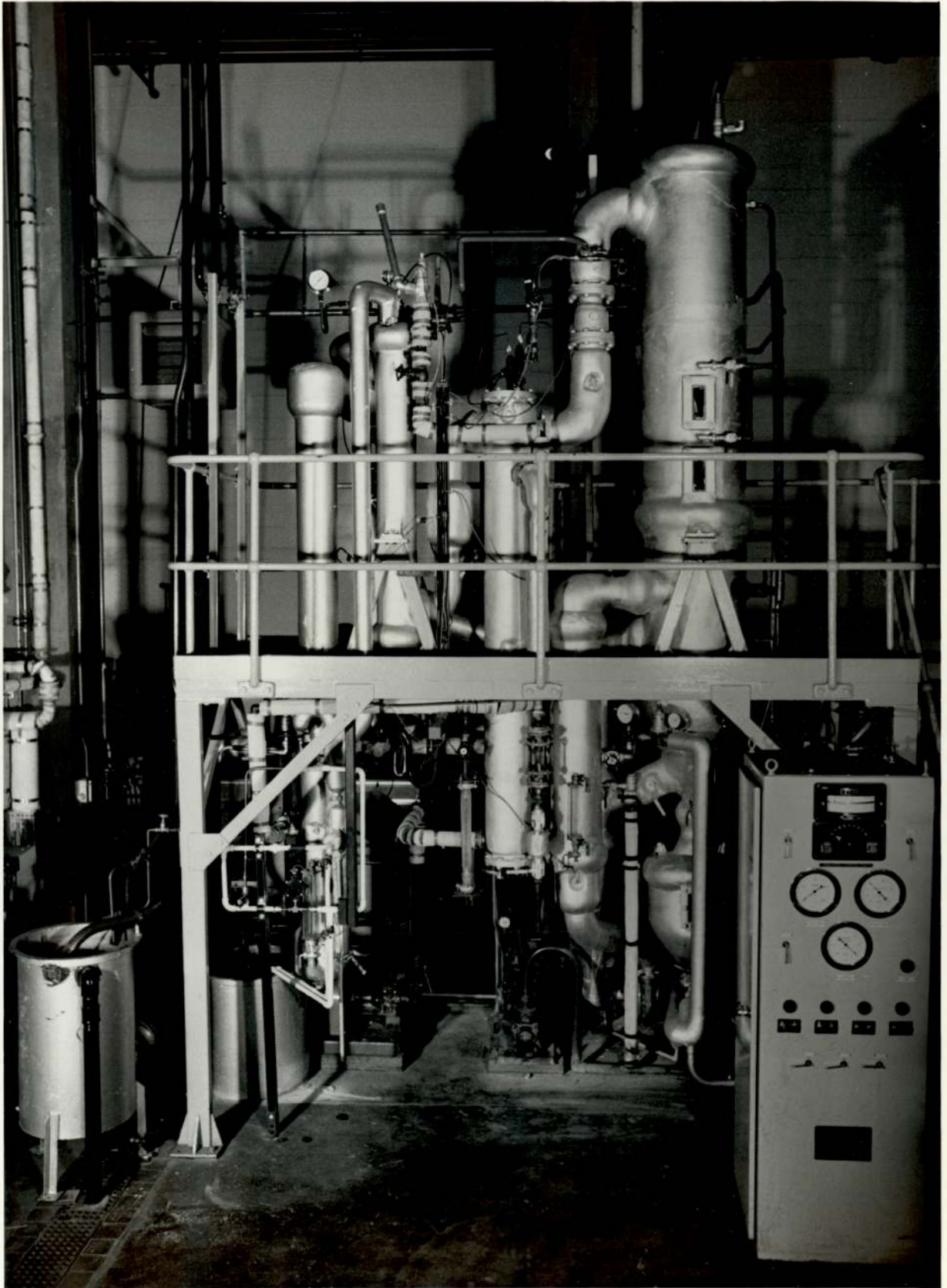
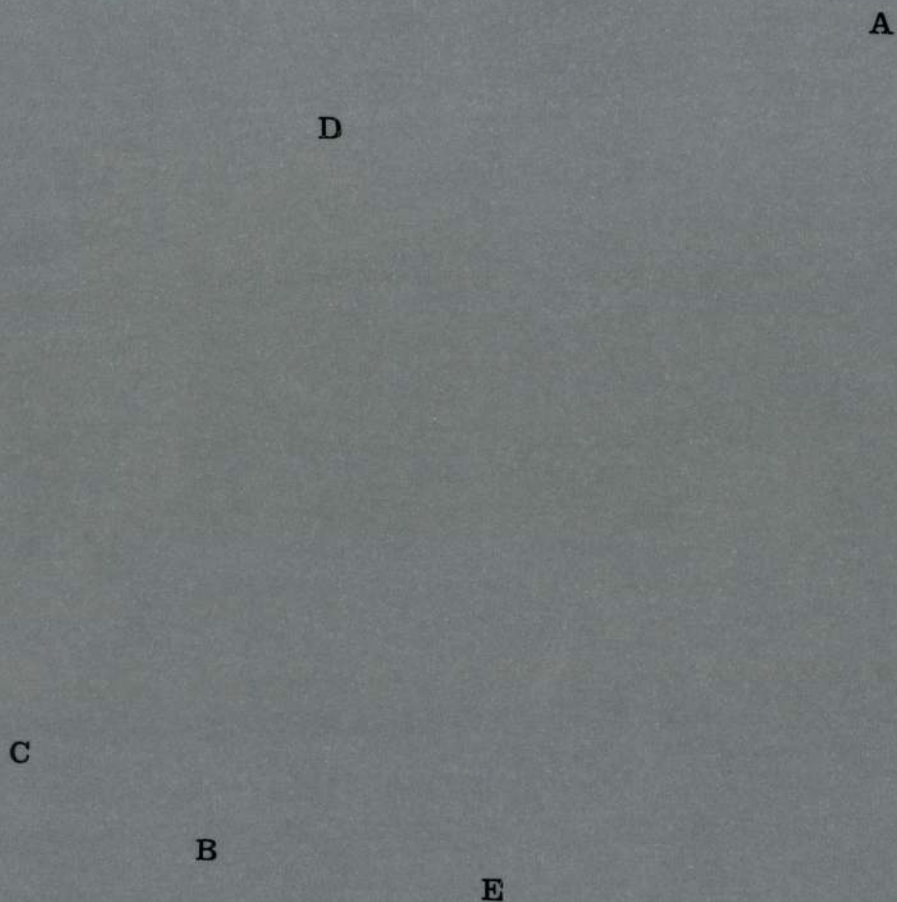
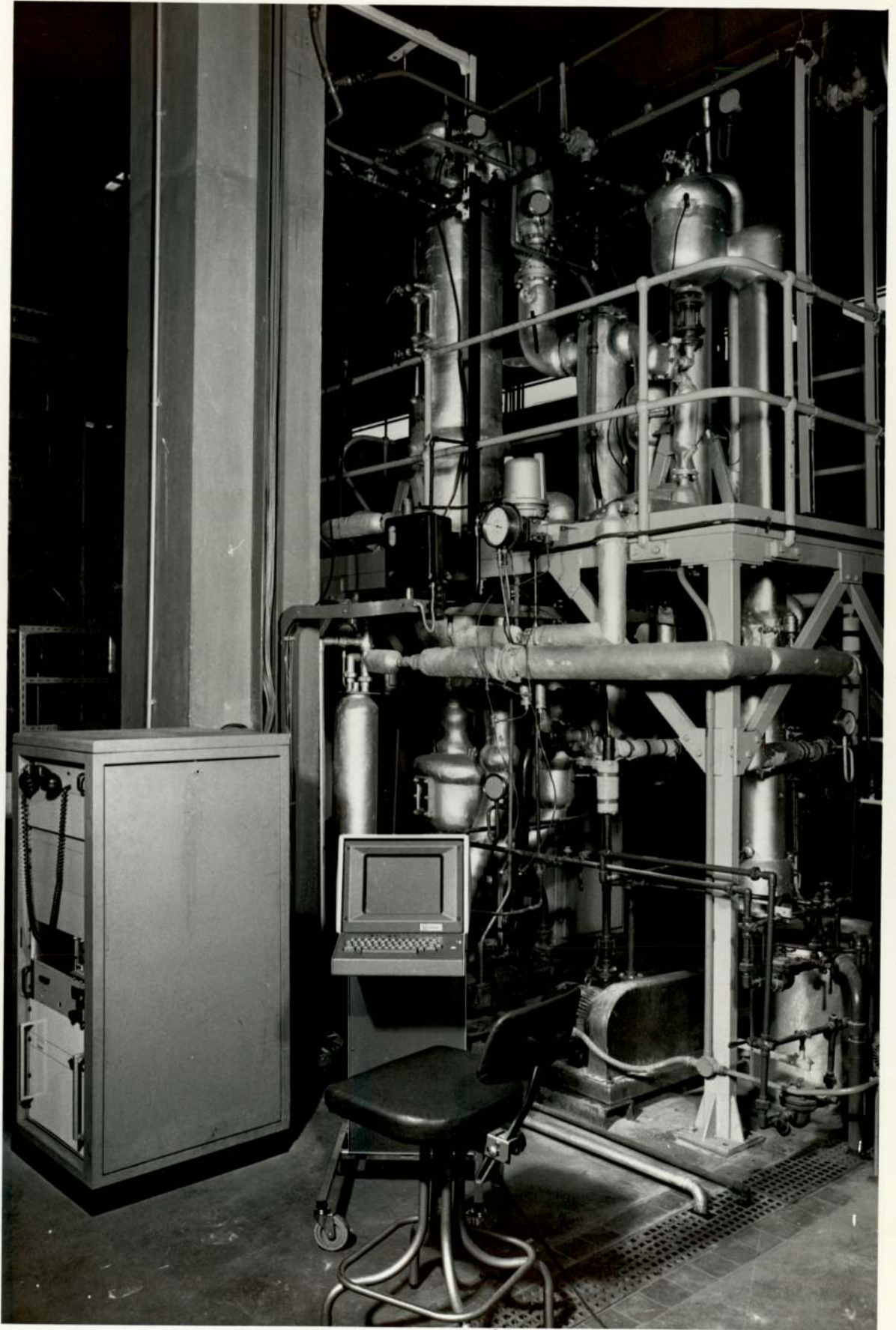


PLATE 2 - DOUBLE EFFECT EVAPORATOR (REAR VIEW SHOWING REMOTE
CABINET)



KEY A = Cyclone Separator
B = Remote Cabinet
C = Direct Telephone Link to Computer
Room
D = Cable Ducting to Computer Room
E = Visual Display Unit



on to the tubeside of the first effect. Heat transfer to the tube contents of the first effect is by condensation of steam onto the tube walls; steam supply to the first effect shell is from the departmental steam generator. The evaporation process which takes place inside the tubes of the first effect is rather complex and can be best visualised by dividing it into three regions. On entering the tubes the liquid is first of all heated to its boiling point, following which there is a region of nucleate boiling and slug flow. Finally comes the region where the vapour flows up the centre of the tube dragging a liquid film up the tube walls. In this region the film appears to climb up the tube walls and it is this physical feature which gives rise to the name of this evaporation unit. In the climbing film region evaporation is greatly promoted by the fact that the only phase in contact with the tube walls is the liquid phase. The vapour/liquid mixture leaving the first effect is separated by means of a cyclone separator into a liquid stream which passes directly to the second effect separator and a vapour stream which passes firstly to the shellside of the preheater and then to the second effect shell; condensation of a proportion of this vapour takes place in both units.

Liquid from the base of the second effect separator is pumped through the tubes of the second effect calandria by a centrifugal type pump. The heated liquid then passes back to the separator under the pressure of a hydrostatic head caused by the liquid level in the separator being above that of the liquid entry point. Since the vapour space in the second effect separator is subjected to the pull of the vacuum pump flash boiling occurs as the heated liquid from the tubes of the calandria enters.

Vapour from the second effect separator and vapour/liquid from the second effect calandria shell enter the shellside of the water cooled, vertical, shell and tube condenser, where all of the remaining vapour is condensed and then drawn off from the base of this unit by a positive displacement rotary pump (Mono pump).

When the evaporator is operated under vacuum conditions, the vacuum pump draws on the shellside of the condenser.

5.2.2. PROCESS OPERATION

START UP

Start up of the evaporator is achieved by adopting the following procedure:

(i) All electricity power supplies are turned on.

(ii) The water supply to the ground level feed tank and pump seals is turned on.

(iii) The cooling water supply to the condenser tubes is turned on.

(iv) The pump feeding the header tanks is switched on.

(v) When the header tanks are full, the second effect separator is half filled with liquid feed using the specially installed pipelines connecting the header tanks and the separator.

(vi) The condensate pump is primed and then switched on.

(vii) The vacuum pump is switched on and once the pressure has fallen to a satisfactory level the circulation pump is also switched on.

(viii) The required level of feed rate from the header tanks to the preheater tubes is established.

(ix) Once liquid has been observed passing through the section of glass tube in the liquid line coming from the cyclone separator, the required level of steam supply to the first effect shell is established. For the next five minutes the various vapour traps and bleed lines from the first effect shell are left open to allow all air to be purged from the system.

(x) The system is then allowed to come to steady state for a period of fifteen minutes before commencing experimentation.

RUNNING PROCEDURE

During the entire operating period the following checks are made at regular intervals:

- (i) A satisfactory vacuum is being maintained.
- (ii) The level in the second effect separator is not too high. This is necessary since there is no liquid take off from the separator and consequently, during certain operating conditions the level will rise.
- (iii) The pressure of the steam supply is constant.

SHUT DOWN

Shut down of the evaporator is achieved by the following procedure:

- (i) The steam supply is turned off and the first effect shell allowed to drain.
- (ii) The liquid feed supply to the preheater tubes is turned off.
- (iii) The circulation and vacuum pumps are turned off. This allows air to enter the system through the vacuum pump seals thus allowing the evaporator vapour spaces to return to atmospheric pressure.

(iv) The second effect separator, second effect tubes, preheater tubes and first effect tubes are all drained of liquid.

(v) When the condenser shell has been drained of liquid the condensate pump is switched off.

(vi) All cooling water and mains water supplies are turned off.

(vii) All electricity supplies are turned off.

5.2.2. PROCESS NOTATION

In order to distinguish between the many process variables, parameters and physical properties associated with the double effect evaporator the following method of process notation is adopted.

(i) Each stream of the double effect evaporator is arbitrarily assigned a reference number, as shown in Figure 5.3.

(ii) Each unit of the double effect evaporator is assigned a two letter mnemonic as follows,

Preheater - PH

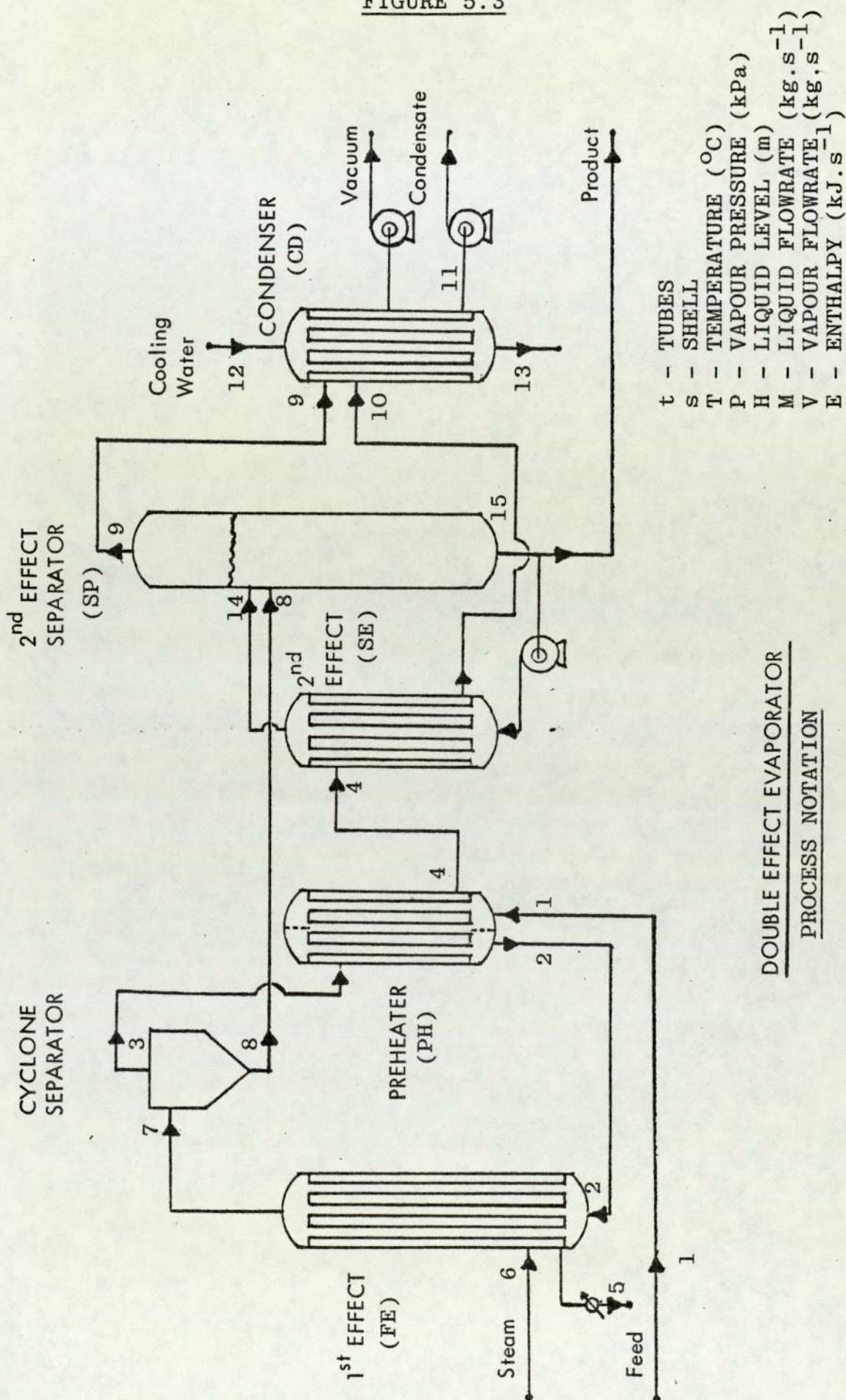
First Effect - FE

Second Effect - SE

Second Effect Separator - SP

Condenser - CD

FIGURE 5.3



(iii) Temperature, vapour pressure, liquid level, liquid flow, vapour flow and enthalpy are denoted by the symbols T, P, H, M, V and E respectively.

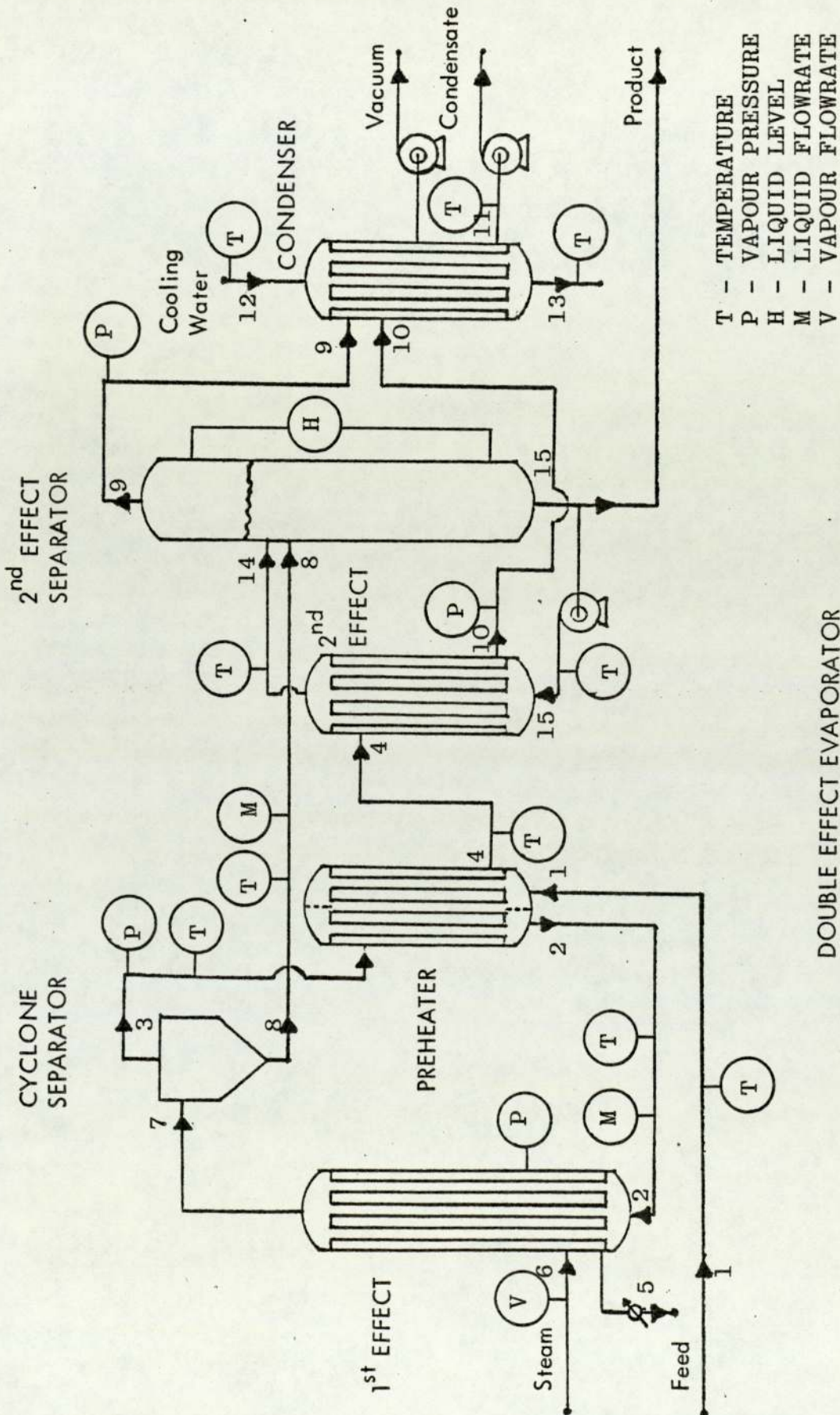
(iv) Subscript t refers to the tubeside and subscript s to the shellside.

Thus, T_{11} refers to the temperature of the condensate, M_1 refers to the liquid feed flowrate, H_{SP} refers to the liquid level in the second effect separator and if C were to represent specific heat then C_{FEt} would refer to the specific heat of the first effect tubes.

5.2.4. PROCESS INSTRUMENTATION

At the double effect evaporator eighteen process variables are measured by four different types of transducer. With the exception of the turbine flowmeter all of the transducers provide analogue signals in the form of voltages or currents. The output from the turbine flowmeter is in the form of a TTL compatible pulsed voltage, the frequency of which can be monitored by use of the counter input subinterface of the HADIOS data logger - see section 5.3.3. As all of the measurement signals are susceptible to corruption by electrical noise, all cabling is screened and all screens are insulated. The positioning and function of each transducer is shown schematically in Figure 5.4.

FIGURE 5.4



T - TEMPERATURE
P - VAPOUR PRESSURE
H - LIQUID LEVEL
M - LIQUID FLOWRATE
V - VAPOUR FLOWRATE

DOUBLE EFFECT EVAPORATOR

PROCESS INSTRUMENTATION

Temperature is measured by NiCr/NiAl thermocouples, using an isothermal (0°C) reference chamber as the cold junction. This isothermal reference chamber is incorporated into the data logger remote cabinet which is situated adjacent to the evaporator. Pressure measurement both absolute and differential, is by strain gauge transducer of the wheatstone bridge type. Two different types of flowmeter are used to make on-line measurements of flowrate. The flowrate to the first effect tubes is measured by a turbine flowmeter, whereas flow from the base of the cyclone separator is measured by a variable area magnetic flowmeter. The flowrate of cooling water to the condenser tubes is measured off-line by a rotameter. A summary of the specifications of each transducer installed at the double effect evaporator is given in Table 5.2.

TABLE 5.2 - DOUBLE EFFECT EVAPORATOR INSTRUMENTATION

CHANNEL NO.	VARIABLE NAME	INPUT RANGE	OUTPUT RANGE	INSTRUMENT TYPE	DESCRIPTION OF MEASURED VARIABLE
0	T ₁₂	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. cooling water to condenser tubes
1	T ₁₃	0-100°C	0.4.1MV	Thermocouple ⁴	Temp. cooling water to condenser tubes
2	T ₁	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. feed to preheater tubes
3	T ₂	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. feed from preheater tubes
4	T ₁₅	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. liquid to 2nd effect tubes
5	T ₁₄	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. liquid from 2nd effect tubes
6	T ₈	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. liquid from cyclone separator
7	T ₁₁	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. liquid from condenser shell
8	T ₄	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. liquid & vapour from preheater shell
9	T ₃	0-100°C	0-4.1MV	Thermocouple ⁴	Temp. vapour from cyclone separator
10	V ₆	0-50kPa	4-20 MA'	Differential pressure cell	Steam flowrate to 1st effect shell
11	H	0-30kPa	4-20 MA'	Differential pressure cell	Liquid height in 2nd effect separator

cont/...

TABLE 5.2 - DOUBLE EFFECT EVAPORATOR INSTRUMENTATION (cont)

CHANNEL NO.	VARIABLE NAME	INPUT RANGE	OUTPUT RANGE ²	INSTRUMENT TYPE	DESCRIPTION OF MEASURED VARIABLE
12	P ₃	0-170kPa	0-40MV	Strain Gauge	Pressure in cyclone separator
13	P ₉	0-170kPa	0-40MV	Pressure Cell	Pressure in 2nd effect separator vapour space
14	P ₆	100-200kPa	0-40MV	Strain Gauge	Pressure in 1st effect shell
15	P ₁₀	0-220kPa	0-40MV	Pressure Cell	Pressure in 2nd effect shell
16	M ₈	0-0.70kgs ⁻¹	4-20MA ¹	VARIABLE AREA Flowmeter	Liquid flowrate from cyclone separator
3 ₁₇	M ₂	0.0075 -0.075 kgs ⁻¹	0-5V TTL square wave	Turbine Flowmeter	Liquid flowrate from preheater tubes

- 1 Milliamper signals are converted into analogue voltages by connecting a resistance in parallel across the input terminals at the remote cabinet
- 2 All analogue voltages (channels 0 to 16 inc.) are scaled into the range 0 to 5V at the remote cabinet
- 3 The pulsed output of the turbine flowmeter is counted by the high speed counter input subinterface of HADIOS
- 4 All thermocouples are of the NiCr/NiAl type

5.3. THE HONEYWELL 316/HADIOS DATA ACQUISITION SYSTEM

The data acquisition and processing system used for the purpose of state identification of the double effect evaporator consists of the following three units:

(i) A Honeywell 316 Digital Computer and associated peripherals - see Plate 3.

(ii) A Honeywell Analogue Digital Input Output System (HADIOS) - see Plate 3.

(iii) A data logger remote cabinet - see Plate 2. This cabinet contains all the equipment necessary to produce and condition the measurement signals received from the double effect evaporator. On leaving the remote cabinet the measurement signals are in a form acceptable to HADIOS and in which they are more readily transmitted along the trunk cabling to the computer room.

The details relating to Honeywell units in the following sections are taken from standard Honeywell publications (3.2, 3.3).

5.3.1. THE HONEYWELL 316 DIGITAL COMPUTER

The Honeywell 316 computer was designed for both batch scientific applications and real-time,

KEY

- A = H316 Computer
- B = HADIOS
- C = Visual Display Unit
- D = Teletype
- E = Magnetic Tape Cassette Unit
- F = Paper Tape Reader
- G = Paper Tape Punch
- H = Floppy Disc Unit
- J = Hard Copy Unit

H

A

E

J

K

C

B

D

F

G



on-line data processing and control. The wide variety of applications which can be implemented include data acquisition, simulation and batch scientific computation.

These applications are undertaken using an instruction repertoire of seventy two commands, a memory cycle time of 1.6 s and input/output data handling at a maximum word rate of 156 kHz/s. The computer is a stored program, parallel binary type using two's complement machine code. A single 10-bit address with indexing and indirect addressing accesses a 16-bit (i.e. 5 octal digits plus sign bit) coincident-current ferrite core memory; one hardware index register is provided. The memory is a random access system expandable from 4K to 32K words - the computer used in this research has a memory of 16K words. Other features of the CPU include a real time clock, high speed multiply/divide and a single interrupt line.

The general characteristics of the Honeywell 316 computer are given in Table 5.3. and the instruction repertoire is summarised in Appendix B-1.

The H316 computer uses the DAP-16 symbolic assembly program language for translation of

TABLE 5.3 - HONEYWELL 316 LEADING PARTICULARS

Primary power	475 watts, 5.5 amps at 115 vac ±10% at 60± 2Hz
Type	Parallel binary, solid state
Addressing	Single addressing with indexing and indirect addressing
Word length	16 bits (single precision) 32 bits (double precision)
Machine code	Two's complement
Circuitry	Integrated
Signal levels	Active: 0 volts Passive: +6 volts
Memory type	Coincident-current ferrite core
Memory size	16K
Memory cycle time	1.6 μs
Instruction complement	72 instructions
<u>Speed</u> , Add	3.2 μs
Subtract	3.2 μs
Multiply	8.8 μs
Divide	17.6 μs
Standard memory protect	Designed to protect memory data in event of primary power failure
Standard interrupt	Single standard interrupt line
Input/Output Modes	Single word transfer Single word transfer with priority interrupts
Standard I/O lines	10 bit address bus (4 function code and 6 device address), 16 bit input bus, 16 bit output bus, Priority interrupt external control and sense lines

TABLE 5.3 - HONEYWELL 316 LEADING PARTICULARS (cont)

Standard teletype	Read paper tape at 10cps Punch paper tape at 10cps Print at 10cps Keyboard input Off-line paper tape preparation, reproduction and listing.
Environment	Room ambient for computer less I/O devices: 0 to 45°C
Cooling	Filtered forced air

TABLE 5.4 - HADIOS ADDRESS LINES

DEVICE	ADDRESS LINE
A.D.C. ¹	00 (AA) ² , 01 (BB)
COUNTER INPUT	02 (CC), 03 (DD)
DIGITAL INPUT	10 (II)
DIGITAL OUTPUT	13 (OO)
CONTROLLER	17 (YY)

¹ The two 16 channel multiplexers attached to the A.D.C. are numbered 00 and 01

² The codes in parenthesis are for cross referencing purposes with Appendix B-2

source programs to machine code. The other source program languages available to the user are BASIC-16 and FORTRAN.

5.3.1.1. PERIPHERAL EQUIPMENT

Six peripheral devices are connected to the I/O bus of the central processor and with the exception of the visual display unit are all to be found in the departmental computer laboratory. The peripherals are as follows:

(i) A high speed paper tape reader operating at 200 characters per second.

(ii) A high speed paper tape punch operating at 75 characters per second.

(iii) A magnetic tape cassette unit, used both for input and output; input at 375 bytes per second and output at 375 bytes per second.

(iv) A teletype operating at 10 characters per second.

(v) A Tektronics 4010-1 visual display unit (VDU). This VDU is capable of both graphical and character display and in the alphanumeric output mode operates at a rate of 200 baud. It is possible to use the VDU both in the computer room and adjacent to the double effect evaporator.

Attached to the VDU is a hardcopy unit, which produces permanent copies of the display when these are required. The hardcopy unit can be operated either by depressing a switch situated on the VDU or by programmed command.

(vi) An AED 3100P floppy disk drive unit.

5.3.2. THE HONEYWELL ANALOGUE DIGITAL INPUT OUTPUT SYSTEM (HADIOS)

The Honeywell Analogue Digital Input Output System (HADIOS) provides a flexible method of interfacing the H316 computer to a wide range of input/output devices in on-line applications. HADIOS consists basically of a controller, connected to the computer input/output data and control lines, which generates subsidiary data, addresses and controls for up to fifteen different subinterfaces. These subinterfaces can be analogue or digital and input or output. The modularity of construction of HADIOS allows systems to be simply configured to meet the particular requirements of each application.

The following standard subinterfaces are included in the HADIOS system used in this research:

(i) High level analogue inputs. This

subinterface consists of a single channel analogue to digital converter (A.D.C.), with a maximum conversion rate of 40 kHz, connected to two sixteen channel multiplexer subinterfaces. Thus, a maximum of thirty two separate analogue signals can be processed in turn by the ADC, the input signal (0 to 5v) being converted to a binary integer with ten bit resolution, i.e. 0 to 1023_{10} .

(ii) High speed counter input. The high speed counter input provides a method of monitoring the number of changes of level of a discrete input. The counter is incremented by a positive voltage pulse from logic '0' to logic '1'. The current contents of the counter can be obtained by a programmable command. Facilities for presetting the counter and interrupt generation when half full are also available and can both be achieved by programmable commands (see Appendix B-2). Each counter input subinterface card has an eight bit register with a range of 0 to 255_{10} . When this subinterface is being operated in the non-interrupt mode the eight bit register automatically returns to zero when a count of 255_{10} is reached whereas in the interrupt mode

when the 8 bit register is half full, i.e. it contains 127_{10} , an interrupt of the central processor is generated.

(iii) Logic level non-isolated input. This subinterface senses the voltages present on sixteen parallel input lines. The input signals are low impedance voltages switchable from logic '1' to logic '0'. The signals sensed are transferred to the A register of the computer as a sixteen bit pattern.

(iv) Logic level flip flop output. This subinterface enables a sixteen bit pattern to be output to sixteen parallel output lines. For each bit of the output value set to binary '1' the appropriate output line is raised to logic '1'. The output lines retain their current levels until the next value is transferred to the subinterface from the computer.

To summarise the following facilities were available:

- 32 Analogue inputs
- 1 Counter input
- 1 Logic level input
- 1 Logic level output

Details of the address line connections between

the controller and the various subinterfaces are given in Table 5.4.

Operation of these subinterfaces is achieved by the output of control signals (initiated by programmable command) from the computer to the HADIOS controller, which then carries out the majority of address and function decoding before initiating the required function of the requested subinterface. The system also has the ability to operate under interrupt control. This kind of operation is governed by the interrupt mask circuits of the HADIOS controller which can be set by the execution of programmable commands. Details of the programmable instructions used to operate HADIOS are given in Appendix B-3.

In this research seventeen of the analogue input channels were used to record the output of four pressure transducers, two differential pressure cells, one variable-inductance flowmeter and ten thermocouples. The counter input subinterface was used to record the pulsed output of a turbine flowmeter and logic level input and output options were used for remote program control and program status respectively.

5.3.3. REMOTE SIGNAL CONDITIONING

In order to convert the signals produced by the various measuring instruments into a form acceptable to HADIOS, a remote signal conditioning cabinet was constructed by departmental technicians. Within this cabinet all the analogue signals are converted, where necessary, to voltages and then amplified into the range 0 to 5v D.C. Also contained in this cabinet are the line drivers for the transmission of digital signals, the power supplies required by the transducers and the De La Rue Zerac isothermal reference chamber. The cabinet is situated adjacent to the evaporator both for convenience and also to minimise the effects of electrical noise during the transmission of the signals to the computer room.

5.4. CHAPTER REVIEW

The equipment used in the on-line application of various forms of the Kalman Filter is discussed. The double effect evaporator is described diagrammatically and details are given of the process occurring and the method of operation. The instrumentation associated with the evaporator is listed and the function of each transducer is tabulated. Finally, the more important features of the data acquisition system are described. A full account of the data acquisition system is not included in this thesis because such a description would not only be extremely lengthy but also unnecessary in order to understand the experiments carried out.

CHAPTER 6

HONEYWELL 316 COMPUTER SOFTWARE

6.1. INTRODUCTION

The H316 computer described in the previous chapter was used in this research for both on-line and simulation experiments concerned with the double effect evaporator. For such applications the available software ideally needs to be versatile, efficient in its use of computer memory, easy to implement and reasonably quick in its execution. Clearly it would be extremely difficult to combine all of these features within the same applications programs and so it was necessary to achieve some kind of compromise. This was done by combining standard software and user written programs in such a way that in the early stages of this research, when new techniques were being tested, the emphasis was put on ease of implementation and versatility whereas in the later stages the emphasis was placed upon speed of execution and efficient use of computer memory.

In order to achieve these specifications when constructing the required applications programs the following standard software was used,

- (1) BASIC-16 Interpreter.
- (2) FORTRAN Compiler.
- (3) DAP-16MOD2 Assembler.
- (4) OP-16 Real time operating system.
- (5) Utility packages to load relocatable

programs into core and dump them out again. This software was used to produce the following applications programs,

(1) HADIOS EXEC. - An interactive, on-line, data acquisition system which was made by combining the BASIC compiler with a number of subroutines written in the DAP-16 assembler language.

(2) O.L.D.F.P. - An On Line Digital Filtering Package which uses the OP-16 real time operating system to schedule programs written in DAP-16 and FORTRAN.

In addition the previously reported Aston Simulation Package (6.1) and the computer software used to drive the graphical capabilities of the Tektronics 4010-1 VDU will be discussed.

6.2. STANDARD SOFTWARE

6.2.1. BASIC-16 INTERPRETER

The Beginners All purpose Symbolic Instruction Code (BASIC) is an interactive, problem orientated, high-level language with a simple vocabulary and grammar. The language was originally developed at Dartmouth College and general details are well documented (6.2).

The BASIC compiler is interpretative in operation, i.e. each instruction is translated from source to machine code and executed whenever it is encountered. All constants are stored internally in floating point format but input may be in integer, fixed point or floating point form. The output format is adjusted by BASIC to provide maximum precision from six figure significance.

The BASIC-16 Interpreter (6.3) is the Honeywell version of BASIC for series 16 machines with memory size 4K words or more. In standard form, communication with BASIC-16 is from the V.D.U. or a teletype, but a machine code modification to the interpreter program (See Appendix B-3) enables input/output via the paper tape reader and punch. One feature of BASIC-16 which makes its use extremely attractive when

the computer used has no backing store is that source code can be punched onto paper tape using an off-line teletype, loaded into the computer at run time via the paper tape reader and then, after the necessary on-line editing to debug the program, the source code can be dumped onto paper tape via the paper tape punch.

An additional refinement provided in BASIC-16 is the CALL statement, which enables a FORTRAN/DAP-16 subroutine to be accessed from a BASIC program. The general form of the statement is

$$\text{ln CALL (sn, } A_1, A_2, \dots, A_n)$$

where, ln is the line number of the statement,

CALL is the statement operator,

sn is the subroutine reference number,

and, A_1 to A_n are arguments to be passed to the subroutine called. Unlike the CALL statement in FORTRAN, the subroutine is not accessed by name but by a reference number to an entry in a table containing the starting addresses of the subroutines. The arguments A_1 to A_n correspond to the dummy arguments of the FORTRAN subroutine definition. Since all BASIC variables are real in the FORTRAN sense, the dummy arguments in the FORTRAN or DAP-16 subroutine must also be real

and any integer numbers required must be converted internally. Where a FORTRAN dummy argument is a subscripted variable, the corresponding BASIC argument is the first subscript required of the array, e.g. CALL (1, X, A(0), B(0,0)). This distinction is necessary because BASIC arrays are numbered from zero whereas FORTRAN arrays are numbered from one.

6.2.2. FORTRAN COMPILER

The Honeywell FORTRAN IV compiler has been produced for series 16 machines according to the American Standards Association specification (6.4). Details of the programming language are well documented (6.5). Operation of the compiler in the batch mode requires the addition of simple control characters (\$O or \$END) to terminate each program. Peripheral device codes are as follows,

- 1 - Teletype
- 2 - paper tape reader/punch

In Honeywell FORTRAN (6.6) a number of non-standard refinements are available which permit both direct access to the computer's memory and modifications of the main arithmetic register

(the 'A' register). A summary of these instructions is given in Appendix D-2.

FORTTRAN source tapes are prepared off-line on paper tape in the standard format. The object code generated by the compiler is normally directed to the paper tape punch and the listings are output to the teletype.

6.2.3. DAP-16MOD2 ASSEMBLER

To avoid programming directly in machine code, a symbolic assembler, DAP-16MOD2, is provided by the manufacturer (6.7). Each machine operation is assigned a symbolic name (a summary of DAP-16 instructions is given in Appendix B-1) and where necessary, each address referenced by an instruction is given a symbolic label. The assembler is a 'one for one' language, i.e. one symbolic instruction corresponds to one machine code operation, except in the case of pseudo-operations, which request action by the assembler rather than specifying an operation code. The source code, punched off-line on paper tape, is usually assembled in two passes in order to minimise the object code.

The assembler produces two independent

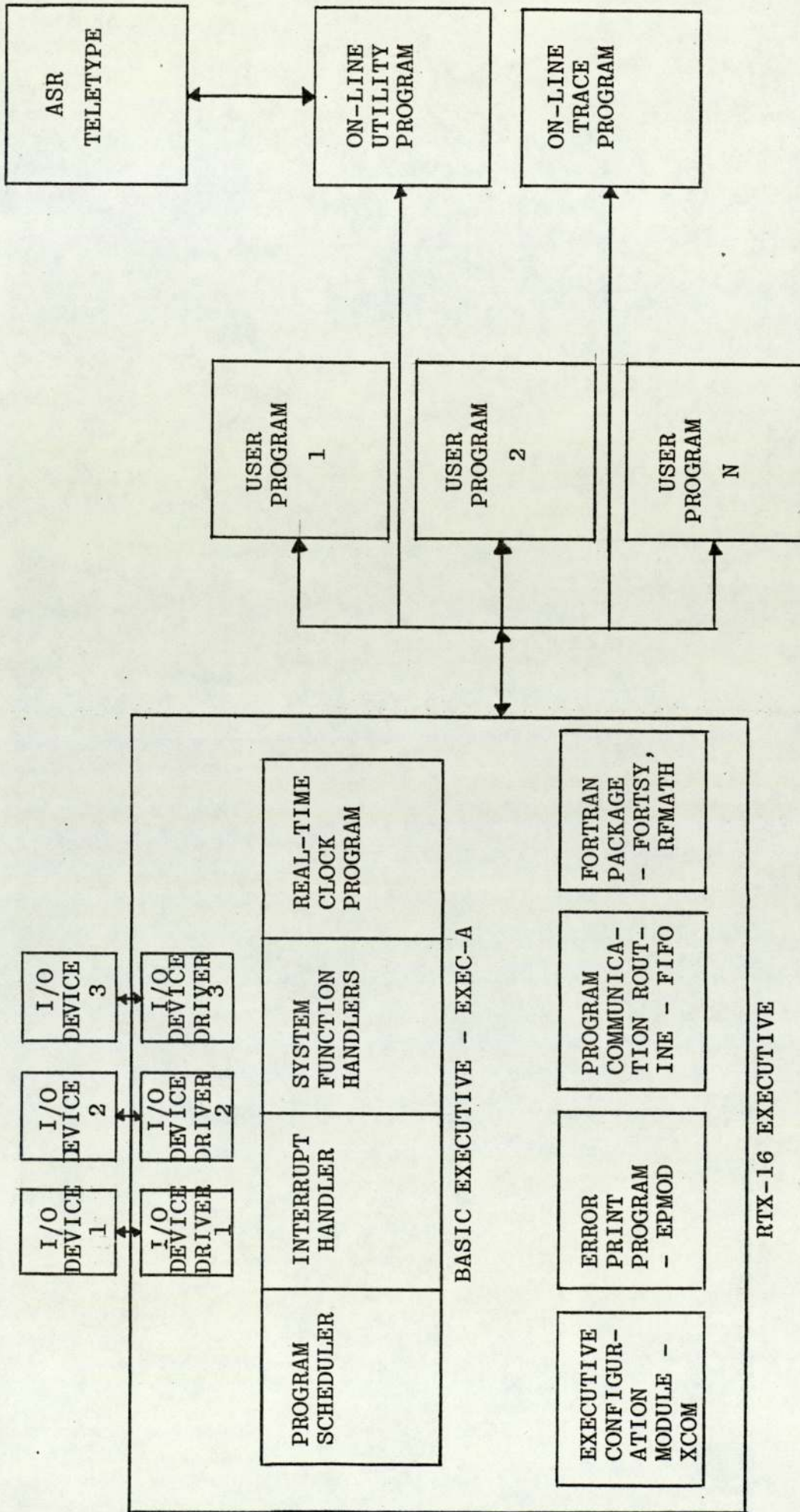
outputs. The first is the object code which is punched onto paper tape for further processing by the loader, and the second is the assembly listing which is printed at the teletype. Included in the listing are programmer comments, any error messages and an octal representation of each machine code instruction or data word. Examples of DAP-16MOD2 assembly listings are included in Appendices C and E.

6.2.4. OP-16 REAL TIME OPERATING SYSTEM

OP-16 is a small multiprogramming operating system complete with I/O drivers, utility and support programs, debugging aids, and on-line peripheral device test programs. It is capable of operating in a core-only or core/secondary storage environment and satisfies the requirements of a small, efficient programming system to implement real-time data acquisition and control.

Any OP-16 system is composed of the following components (see Figure 6.1),

- (1) RTX-16 Real-Time Executive
- (2) Utility routines (optional)
- (3) Debugging Aids
- (4) Real-time peripheral device drivers and test programs



GENERAL STRUCTURE OF OP-16 SYSTEMS

FIGURE 6.1

(5) Fortran package

(6) User programs (DAP-16 or FORTRAN)

The RTX-16 Executive is the central and most important component of any OP-16 system. Due to its modular nature it can be configured so that it provides the necessary facilities for any unique system. The capabilities and thus, the components included in any OP-16 system are determined by the configuration module (XCOM) which is written by the user and included in the RTX-16 executive. In its most general form the RTX-16 executive can perform the following functions,

(1) Execute programs according to their priority.

(2) Keep track of the co-ordination requirements of programs, devices and core storage.

(3) Handle the interrupts which communicate external conditions to the Executive and its programs.

(4) Keep track of the time of day in order to execute programs at certain times or after a certain delay.

(5) Handle communications between programs and the Executive.

(6) Handle communications between the operator and the Executive.

(7) Detect and report errors in the system or individual programs.

(8) Perform the necessary book keeping for a multiprogramming multilevel system.

The general principles involved in writing the user programs and configuring an OP-16 system will not be discussed here as they are well documented in the manufacturer's user guide (6.8). However, four general features of the RTX-16 executive which are of particular significance when developing on-line filtering programs will now be discussed and then in section 6.6 the configuration of the RTX-16 executive used in this research will be described.

6.2.4.1. SYSTEM FUNCTIONS

The RTX-16 executive provides a number of system functions which can be requested from either a user program (DAP-16 or FORTRAN) known by the Executive or by the on-line utility program (ONLCUP). These system functions can be summarised as follows:

FUNCTION 1 - REQUEST PROGRAM

The requested program is not started up by this function but merely requested. The Executive Scheduler will then start the program as soon as possible.

FUNCTION 2 - SCHEDULE LABEL

The major use of the Schedule Label function is that a program which services another can call the first one back after its service is complete. Thus, if program 1 requests the execution of program 2 and then waits for the second program to run, the first program can be restarted by a schedule label function at the end of program 2.

FUNCTION 3 - CONNECT CLOCK

This function is used to connect a program to the clock for automatic initiation by the Executive Real Time clock program. This function can be used to cause the periodic scheduling of a program or to cause a program to be executed after a time delay (this puts the program into a wait state).

FUNCTION 4 - DISCONNECT CLOCK

This function requests the executive to stop the periodic execution of a program or to cancel the automatic resumption of a program in the wait state.

FUNCTION 5 - CONNECT INTERRUPT

This function requests the executive to set the interrupt mask so that the specified peripheral device may interrupt the central processor. The connect interrupt function also causes the Executive to establish the necessary linkages with the interrupt response code specified by the user program for the peripheral device.

FUNCTION 6 - DISCONNECT INTERRUPT

This function informs the Executive that the calling program no longer wishes to respond to the named interrupt.

FUNCTION 7 - TERMINATE

When this function is executed the Executive is informed that this program has finished and as a result control is returned to the executive with no return to the program.

FUNCTION 8 - WAIT

This function informs the Executive that the program wishes to suspend execution until it is restarted by a schedule label function.

The programming details associated with the above system functions are given in Appendix D-1

for DAP-16 programs, Appendix D-2 for FORTRAN programs and Appendix D-3 when used in conjunction with the utility program (ONCLUP).

6.2.4.2. INTERRUPT HANDLING

Any peripheral device, including the computer's real time clock, which interrupts the central processor brings into action the Executive Interrupt Handler. Whenever an interrupt occurs, the Interrupt Handler determines the source of the interrupt and jumps to the user's interrupt response code for that interrupt. The linkages between the Interrupt Handler and the interrupt response code are established by the Executive Interrupt Definition Table which is part of the Executive Configuration Module (XCOM). The user's interrupt response code needs to be brief and after its execution control returns to the Interrupt Handler which optionally schedules a label in the user's program and then returns control to the Scheduler. When an interrupt has occurred all non-interrupt code will be suspended until all interrupts have been serviced.

6.2.4.3. THE ON-LINE UTILITY PROGRAM (ONLCUP)

The On-line Utility Program is run under the control of the RTX-16 Executive and provides the following on-line facilities:

- (i) Program debugging tools
- (ii) Data transfer and verification between a variety of storage devices and external media.
- (iii) Optional operator control over the initiation and termination of periodic programs and initiations of 'single shot' programs
- (iv) Control of the location and size of the core area available to the operator for on-line manipulations
- (v) Printing and adjusting of the time of day.

Essentially ONLCUP provides a conversational interface between the operator and the OP-16 operating system via a teletype or VDU. Use of ONLCUP is quite straightforward and it can be started up most of the time that the RTX-16 Executive is running. Typing a dollar sign (\$) on the teletype or VDU causes an interrupt of the central processor and as a result of this ONLCUP is started up. On start up 'SF=' is typed on the teletype and then ONLCUP remains in the receive

mode until a legitimate command is received. A list of the commands available when using the ONLCUP program is given in Appendix D-3. The utility program can be terminated at any time by typing in an exclamation mark (!) when the message 'SF=' is output.

6.2.4.4. THE FORTRAN PACKAGE

The FORTRAN package which can be incorporated in the RTX-16 Executive provides one method of running FORTRAN programs in an OP-16 system. It consists of two main sections,

- (i) OPED - the input/output editor
- (ii) RFMATH - the re-entrant floating point mathematics routines.

Providing a FORTRAN program is constructed using the special OP-16 Fortran Libraries both of the above facilities are used. This means that the total core requirements of a number of FORTRAN programs is reduced due to the use of these common subroutines.

FORTTRAN programs to be run under OP-16 should begin and finish with the HEADER and TERMINATE statements respectively but apart from this they can be written in standard FORTRAN IV. Details of the above statements and the other special

features of Honeywell Extended Fortran permitted in programs to be run under OP-16 are given in Appendix D-2.

6.2.5. UTILITY PROGRAMS

When running a program written in either the FORTRAN or the DAP-16MOD2 languages, the first step is the off-line preparation of a source tape. The result of a successful compilation or assembly of this source code is an object code tape, which must be loaded into the desired locations in core and then dumped onto paper tape to form a self-loading system tape (SLST). This is achieved by loading an SLST of the object tape loader (LDR-APM) into memory and utilising it to enter the object code tape of the program. Object tapes of supporting programs and/or library subroutines are loaded similarly. When all of the necessary object code programs have been loaded an SLST of the punch and load program (PAL-AP) is loaded into core and is then used to produce an SLST of the complete program on either paper tape or magnetic tape cassette.

6.2.5.1. OBJECT LOADER (LDR-APM)

Object code produced by the DAP-16MOD2 assembler or the FORTRAN compiler is processed by a loader to form a core image in memory. References to external names such as library or user written subroutines are also resolved. To the loader, object code tapes of programs written in both DAP-16MOD2 or FORTRAN are identical and so programs from both of these sources can be combined.

The memory of a 316 computer with 16K words of core is divided into 32 sectors each containing 512 words of memory. Since an instruction word of sixteen bits requires four bits to represent a sufficient number of operands, direct addressing is only possible within the same sector or between any sector and the sector designated as base sector (base sector is usually the lowest sector of memory but by use of the base sector relocation program shown in Appendix B-4 it can be changed to other sectors in core). This restriction on direct addressing means that when loading object code tapes to form a program which occupies more than one sector of memory it is necessary to provide indirect address links to satisfy intersector references. The loader handles these indirect

address links in three ways. The first method is to locate these links in base sector. This is perhaps the most straightforward method and as a result is widely used. Alternative methods involve the location of these links within the same sector as the instruction requiring indirect addressing, this is referred to as desectorised loading. The second method achieves this by use of the SETB (set base) pseudo operation within a DAP-16MOD2 program and the third method by allocating an area of core at load time. When using the second and third ways of satisfying intersector references care is needed to ensure all links are handled correctly and consequently these methods are only used when the first method is difficult to implement.

As mentioned above it is possible to change the sector which is designated as Base Sector by using the Base Sector Relocation program shown in Appendix B-4. This feature of the H316 computer is extremely useful when constructing packages which involve the BASIC-16 Interpreter because, in such cases, there are only 48 locations available in sector 0 for the intersector references required by the FORTRAN or DAP-16 subroutines. To avoid having to construct these

packages by using a desectorised load, the BASIC-16 Interpreter uses Sector 0 as Base Sector and the FORTRAN/DAP-16 Subroutines use Sector 37 as Base Sector. This means that every time the Basic CALL statement is executed Base Sector needs to be relocated and this is done by the DAP-16 program shown in Appendix B-4.

When loading object tapes the loader provides a method of checking which programs have been loaded and which ones are still required by giving the operator the option of starting the execution of the loader two locations above its normal entry point. This action causes a memory map of the routines involved to be displayed on either the V.D.U. or the teletype. The memory map obtained also gives information about the intersector linkages which are referred to as BASE by the loader.

6.2.5.2. PUNCH AND LOAD PROGRAM (PAL-AP)

Program tapes that are called self loading system tapes (SLST) are ones which can be loaded into core by use of the key-in loader (a small machine code program which permanently occupies the first 17₈ locations of core) and a two-part self contained loader. SLST's are produced by

using PAL-AP to dump the specified area of computer memory onto paper tape. PAL-AP's first action is to punch out a two part self-contained loader that is used to supplement the key-in loader at program load time. The contents of the memory are then punched out.

6.3. GRAPHICAL SOFTWARE

One of the most attractive features of the Tektronics 4010-1 VDU is that it is capable of providing graphical displays. To drive this graphical capability the manufacturers provide a set of subroutines (6.9) which, because they are written in FORTRAN, are completely machine independent. Thus, after the addition of two simple machine code programs, one for input and the other for output, a graphical system can be easily implemented on the H316 computer.

6.3.1. EXTENDED BASIC GRAPHICS

EXTENDED BASIC GRAPHICS is a package based on the BASIC-16 Interpreter and nine FORTRAN subroutines which implement the full graphical facilities of the Tektronics 4010-1 VDU. The nine FORTRAN subroutines, which can be accessed from a BASIC program via the CALL statement, are essentially communications programs which provide both efficient access to the manufacturers graph plotting software and compatibility between the BASIC and FORTRAN sections of the package. Brief details of the facilities available to the BASIC programmes are given below. For full details

refer to the manufacturer's user guide (6.9) and the EXTENDED BASIC GRAPHICS user manual (6.10).

Subroutine 1

Purpose : To enter or leave the graphics mode.

CALL : CALL (1, F, X, Y)

Arguments : F = 1 enter graphics mode

F ≠ 1 leave graphics mode

X, Y screen coordinates of beam after termination.

Subroutine 2

Purpose : To set the virtual or screen windows.

CALL : CALL (2, F, XO, X1, YO, Y1)

Arguments : F = 1 set virtual window

F ≠ 1 set screen window

XO, X1, YO, Y1 virtual or screen coordinates.

Subroutine 3

Purpose : To perform the following graphical functions; draw a line, move the graphical cursor and plot a point.

CALL : CALL (3, F, X, Y)

Arguments : F = 1 to 12 inclusive - type of graphical function - see (6.10)

X, Y Co-ordinates moved to by the graphical cursor.

Subroutine 4

Purpose : To draw a dashed line

CALL : CALL (4, F, X, Y, L)

Arguments : F = 1 to 4 - type of dashed line -
see (6.10).

X, Y Co-ordinates moved to by the
graphical cursor.

L Dashed line specification -
see (6.9).

Subroutine 5

Purpose : To provide graphical input via the cursor

CALL : CALL (5, F, C, X, Y)

F = 1 Virtual cursor

F \neq 1 Screen cursor

X, Y Co-ordinates of cursor

C Code for keyboard character input -
see (6.9).

Subroutine 6

Purpose : To output alphanumeric characters

CALL : CALL (6, C)

Argument : C Code for character output - see (6.9).

Subroutine 7

Purpose : To provide facilities for adjusting the
position of the alphanumeric cursor.

CALL : CALL (7, F)

Arguments : F = 1 to 12 - function required - see
(6.10).

Subroutine 8

Purpose : To examine or change the common area of
the FORTRAN subroutines

CALL : CALL (8, F, A(1))

Arguments : F = 1 Copy COMMON area into the array A
F \neq 1 Copy Array A into the COMMON area
A Array dimensioned A(55) in the
BASIC program.

Subroutine 9

Purpose : To input a character from the keyboard

CALL : CALL (9, C)

Argument : C Code for character input - see (6.9).

The construction of the Extended Basic Graphics package is achieved by the following procedure:

(1) Load an SLST of the BASIC-16 Interpreter into sectors 0 to 7.

(2) Using LDR-APM, load the object tapes of the nine FORTRAN subroutines accessible from BASIC into the top of core.

(3) Load the object tapes of manufacturers software.

(4) Load the object tapes of the FORTRAN library subroutines.

During the loading of the above subroutines the intersector references are located in sector 37 and when the package is being operated the Base Sector relocation program shown in Appendix B-4 is used to change Base Sector when the need arises.

The above loading procedure means that the middle section of core is available for the BASIC program and variable storage. Following a successful load of the various components, an SLST of the Extended Basic Graphics package can be obtained using PAL-AP.

An example of a BASIC program for use with this package is given in Appendix F-8.

6.4. THE ASTON SIMULATION PACKAGE - ASP

The Aston Simulation Package was developed by Gay and Payne (6.1) so that interactive simulation could be carried out on a digital computer. ASP is based on the BASIC-16 Interpreter and four FORTRAN subroutines which carry out the task of numerical integration by either the fourth order Runge-Kutta method or the Modified Euler method. The use of an interactive language restores to the user the advantages of the direct access which is a feature of analogue simulation whilst avoiding the disadvantages of using an analogue computer. In addition, the use of FORTRAN subroutines to perform the integration of first order differential equations reduces the knowledge of numerical analysis demanded of the user.

The FORTRAN subroutines which can be accessed from a BASIC program are as follows:

Subroutine 1 - FORTRAN NAME 'ZERO'

Purpose : To initialise the COMMON area of the FORTRAN subroutines and set the independent variable to zero.

CALL : CALL (1, T)

Arguments : T independent variable.

Subroutine 2 - FORTRAN NAME 'PRNTF'

Purpose : To control output from the BASIC program
and signal the end of the simulation.

CALL : CALL (2, P, E, I1, I2)

Arguments : P the required print out interval.
E the value of the independent variable
at which the run will terminate.
I1 when the independent variable
reaches E, I1 is returned with a
value of 2.0, otherwise I1 = 1.0.
I2 when the independent variable is
zero or a multiple of P, I2 is
returned with a value of 2.0,
otherwise it equals 1.0.

Subroutine 3 - FORTRAN NAME 'INTI'

Purpose : To integrate the independent variable
and perform the necessary housekeeping
operations for the integration
procedure.

CALL : CALL (3, T, H, K)

Arguments : T independent variable
H integration step length
K integration order (K=2.0 for modified
Euler and K=4.0 for Runge Kutta 4)

Subroutine 4 - FORTRAN NAME 'INTX'

Purpose : To integrate the dependent variable.

CALL : CALL (4, X, DX)

Arguments : X dependent variable.

DX derivative of X.

A typical layout for a BASIC program using the four subroutines discussed above is given in Figure 6.2, and contains the following four main sections:

1. Initialisation section

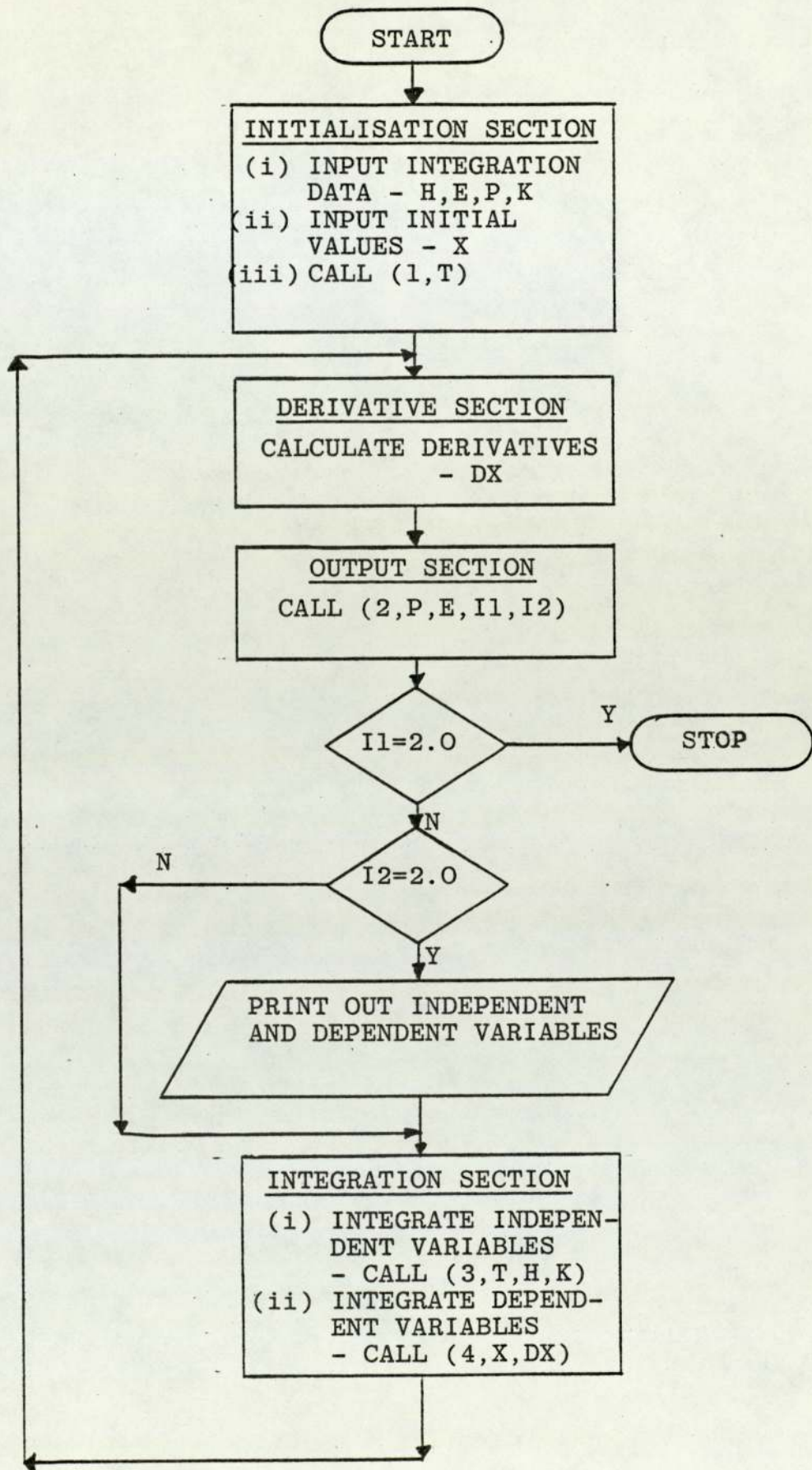
In this section of the program the integration data (H, E, P, K) and the initial values of the dependent variables are input, preliminary calculations are carried out and then subroutine ZERO is called to initialise COMMON and set the independent variable to zero.

2. Derivative section

This section contains the algebraic equations that establish the derivatives of the variables to be integrated.

3. Output section

First of all subroutine PRNTRF is called and then the flags I1 and I2 are tested to see if output is due or whether the end of the simulation has been reached.



GENERAL STRUCTURE OF ASP PROGRAMS

FIGURE 6.2.

4. Integration section

This section consists of a list of CALL statements starting with the independent integration routine INTI, and followed by the CALL statements to routine INTX to integrate the dependent variables. The last statement of this section directs the computation to the first line of the derivative section.

An example of a BASIC program written for use with ASP is given in Appendix F-4.

An SLST of ASP can be obtained by following a similar procedure to that described previously for Extended Basic Graphics.

6.5. THE HADIOS EXECUTIVE PACKAGE MK.2.

As was discussed earlier it was decided that for the preliminary on-line experiments carried out in this research the computer software should be versatile and easily implemented. Since the BASIC-16 Interpreter offers excellent facilities for the modification of user programs, a software package which includes facilities for BASIC programs to access the HADIOS system (see 5.3.2) regularly at a chosen time frequency would satisfy the stated requirements. This package, known as The HADIOS EXECUTIVE PACKAGE Mk.2, consists of two main components,

(1) BASIC-16 Interpreter

(2) HADIOS EXECUTIVE PROGRAM Revision 01., which communicate by the BASIC CALL statement. The HADIOS EXECUTIVE PROGRAM was written to perform the following functions,

(i) To initiate regular access of the HADIOS system at the required frequency. This action is caused by a CALL statement in the BASIC program.

(ii) At the desired time to access (scan) the HADIOS devices required by the BASIC program.

(iii) To handle all interrupts of the computer.

(iv) To provide an idling loop in which to wait for the next scan.

(v) To terminate the execution of the HADIOS EXECUTIVE PROGRAM after the required number of scans have taken place.

(vi) To check for error conditions.

The HADIOS EXECUTIVE PROGRAM is written in the DAP-16MOD2 language and it services the following HADIOS devices,

(a) The analogue to digital converter - a total of 32 analogue channels can be accessed via 2 multiplexers.

(b) The counter input.

(c) Digital input

(d) Digital output.

A description of the HADIOS EXECUTIVE PROGRAM and an assembly listing are given in Appendix C-1.

The construction of the HADIOS EXECUTIVE PACKAGE Mk.2 is a rather complex procedure and so will not be discussed here but is included in Appendix C-5.

6.5.1. DETAILS OF THE 'CALLS' FROM A BASIC PROGRAM

The two CALLS which should be included in a

BASIC program in order to set up communication with the HADIOS EXECUTIVE PROGRAM are,

1. CALL (1,P₁,P₂,P₃,P₄,P₅,P₆,P₇,P₈,P₉,P₁₀,A(0))
- the fundamental CALL which gives the necessary information to the executive for the setting up of regular scanning of the HADIOS devices.
2. CALL (2)
- the secondary CALL on the executive which returns the computer to the idling loop if more scans are required.

PARAMETERS - P_i

The parameters required in the CALL statements are,

- P₁ - HADIOS devices required - see Table 6.1.
- P₂ - Honeywell Real Time Clock Interrupt Frequency (secs); this is the frequency at which scanning takes place.
- P₃ - Total number of Real Time Clock Interrupts (or scans) required.
- P₄ - Ensemble number (>1) - only applicable for analogue measurements (see DATA below).
- P₅ - First analogue channel number.

TABLE 6.1 - DEVICES SELECTED TABLE

VALUE OF P ₁	DEVICES SELECTED
1	Analogue inputs only
8	Counter inputs only
9	Analogue + Counter
64	Digital input only
65	Digital input + Analogue
72	Digital input + Counter
73	Digital input + Counter + Analogue
512	Digital output only
513	Digital output + Analogue
520	Digital output + Counter
521	Digital output + Analogue + Counter
576	Digital output + Digital input
577	Digital output + Digital input + Analogue
584	Digital output + Digital input + Counter
585	Digital output + Digital input + Counter + Analogue

- P_6 - Last analogue channel number.
- P_7 - Type of counter scanning required - only applicable if the counter option is selected.
- = 0 - count of pulses during a clock interval starting with P_8 ($0 \leq P_8 < 255_{10}$) in the counter register at the beginning of each interval.
 - = 1 - Cumulative count of pulses over the total number of scans - P_8 should be set to zero. When using this type of counter scanning the counter register is set to zero at the start of the first scan but from then on is never reset by the program. Care is needed in interpreting the value returned to the user's program because when a count of 255_{10} is achieved the register returns to zero.
 - = 2 - counter interrupts when half full, i.e. when the counter register contains 127_{10} . When using this option the counter register is set to the preset value ($0 < P_8 < 126_{10}$) at the start of every clock interval.

P₈ - Counter preset value.

P₉ - Type of digital input required - only applicable if the digital input option is selected.

= 0 - 16 bit pattern input is converted to the real equivalent of the binary integer and is stored in A(34). The 16 bit word is treated as 1 sign bit and 15 magnitude bits and will be converted to an integer in the range -32768_{10} to 32767_{10} .

= 1 - 16 bit pattern is stored in a 16 element array, A(34) to A(52), each element being either 1 or 0.

P₁₀ - Type of digital output required - only applicable if the digital output option is selected.

= 0 - the real number stored in A(35), by the BASIC program is converted to the equivalent binary integer and output as a 16 bit pattern. The value of A(35) must be in the range -32768_{10} to 32767_{10} and will be converted to its binary equivalent (1 sign bit and 15 magnitude bits).

- = 1 - The 16 element array, A(53) to A(68), each element having been set to either 1 or 0 by the BASIC program, is output as the corresponding 16 bit pattern.
- = 2 - special purpose output - see Appendix C-6.

DATA STORAGE - A(I)

The data values transmitted to or received from the HADIOS EXECUTIVE PROGRAM are stored in the one dimensional array A(I), which must be dimensioned A(68) in the BASIC program.

A(0) to A(31) - value of respective analogue input channel at a clock interrupt. These values are the averages of the ensemble number of readings (P_4) taken at the clock interrupt.

A(32) - time between the preset value being output at a clock interrupt and the first half full counter interrupt (secs).

A(33) - value in the counter input register at a clock interrupt.

A(34) - Digital input value.

A(35) - Digital output value.

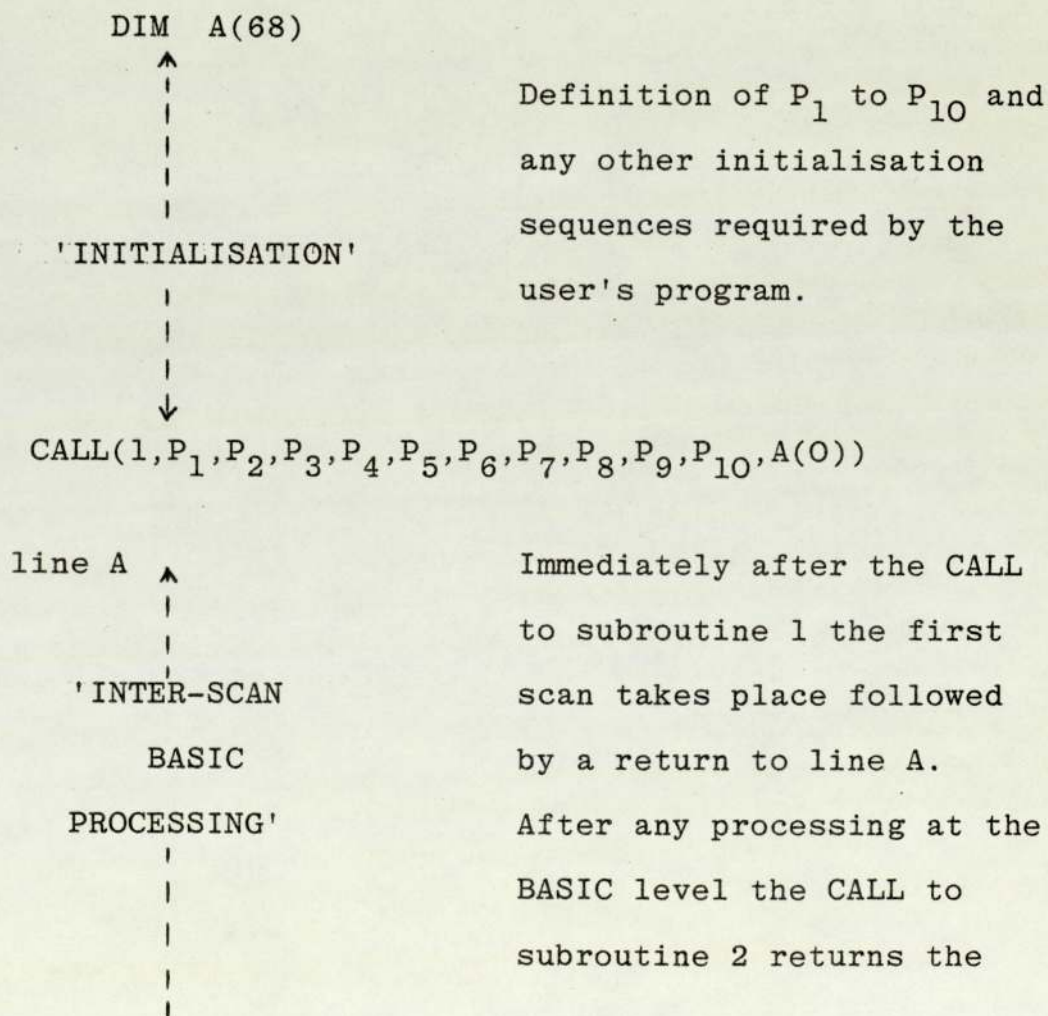
A(36) - Number of half full counter interrupts
since the last clock interrupt.

A(37) to A(52) - digital input array ($P_9=1$)

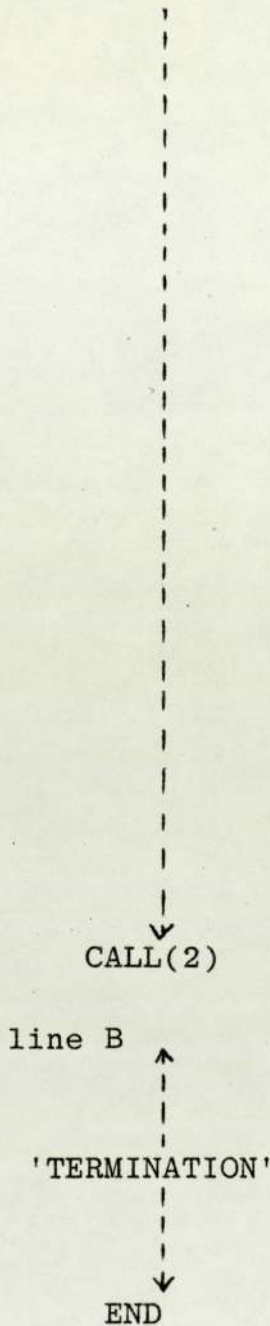
A(53) to A(68) - digital output array ($P_{10}=1$)

6.5.2. STRUCTURE OF THE BASIC PROGRAM

BASIC programs written for use with the Hados Executive Package Mk.2 should have the following structure,



program to the idling loop where the next clock interrupt is awaited. After the clock interrupt and the subsequent scan have taken place a return to line A again takes place. This sequence is repeated until a total of P_3 scans have taken place when, after the normal Inter-scan Basic processing, the CALL to subroutine 2 causes the program to return to line B.



Any termination sequences required by the user's program

6.5.3. ERRORS GENERATED BY THE HADIOS EXECUTIVE PACKAGE MK.2.

Providing the BASIC program has been written according to the above structure and if sufficient

execution time has been allowed for Inter Scan Basic Processing the Hados Executive Package should run without error. However, to safeguard against corruption of the machine code by either mistakes in the BASIC program or incorrect parameter values, a number of error checks are made by the Hados Executive Program. The errors generated are in addition to the standard BASIC-16 errors and are tabulated in Appendix C-7.

In situations where the premature termination of the program is desirable this can be done by setting sense switch two. This action is preferable to the more normal use of sense switch one because it causes all interrupts to be disabled.

6.6. THE OLDFP EXECUTIVE

The first stage in the construction of an OP-16 system is to consider what features the system should include in order to provide the desired facilities. Thus, in the case of the On Line Digital Filtering Package (referred to as OLDFP from now on) the main features required of the executive section of the package are as follows:

(i) To provide a Fortran capability so that Fortran programs can be scheduled and executed when required to do so by the user.

(ii) To handle output to the high speed punch so that the results of the filtering procedure can be punched out onto paper tape.

(iii) To handle input and output to the teletype or VDU so that the user can communicate with both the utility program (ONLCUP) and the Fortran programs known to the executive. The presence of this feature is also required so that the executive can advise the user of any errors which may occur.

(iv) To handle all forms of communication with HADIOS; this includes interrupts.

(v) To provide the user with the on-line utilities needed to interrogate and control the system.

Once the above specifications have been stated the components required to make up the Executive were chosen. As shown by Table 6.2 the first component of the OLDFP Executive is the RTX-16 Executive and this meets the first specification. The second specification is met by the fourth component, i.e. the High Speed Punch Driver, the third by component 3, i.e. the ASR Input/Output Driver and the fourth specification by the second component, i.e. the Hados Input/Output Drivers. Finally, component 5, the On-Line Utility Program satisfies the fifth specification. Apart from the Executive Configuration Module, XCOM, and the supervisory program for the Hados Drivers, all of the components required to construct the OLDFP Executive are standard items of software.

Having decided on the required components, the next task was to write those items of software which are non standard. The Hados Supervisory program and the reasons for its presence will be discussed in the next section. The construction of XCOM was a more complex task since it is in this module that the capabilities of the Executive are established.

TABLE 6.2 - COMPONENTS OF THE OLDFP EXECUTIVE

NAME	COMPONENTS
<p>1 RTX-16 EXECUTIVE</p>	<p>(i) Basic Executive - EXEC-A (ii) Executive Configuration Module - XCOM (iii) Program Communication Routine - FIFO (iv) Error Print Program - EPMOD (v) Fortran Package - FORTSY - RMATH (Routines 1-6)</p>
<p>2 HADIOS INPUT/ OUTPUT DRIVERS</p>	<p>(i) Hadios Digital Input/ Output Driver - HAD-DIO (ii) Hadios Analogue Input Driver - HAD-ANI (iii) Supervisory program - H1-H6</p>
<p>3 ASR TELETYPE INPUT/OUTPUT DRIVER</p>	<p>(i) ASR Driver Program - ASRD-S (ii) Fortran Input/Output Extensions ASF1/2.</p>
<p>4 HIGH SPEED PUNCH DRIVER</p>	<p>(i) High Speed Punch Driver Program HSPD-H (ii) Forced 8 parity check - C\$OPF</p>
<p>5 ON-LINE UTILITY PROGRAM-ONLCUP</p>	<p>See Reference 6.11</p>

Apart from its complexity a number of obscure points had to be resolved during the writing of XCOM. These points, together with others which were encountered, are reviewed in Appendix E-5. The construction of XCOM is discussed in section 6.6.3 and Appendix E-2.

Finally, it had to be decided how the system was to be loaded into the computer's memory, as this had a bearing on some of the more minor components required. The loading procedure was not easily determined since the manufacturer's user's guide (6.8) is not at all clear on this point. The general procedure which was eventually used is as follows;

(i) The system was separated into the five components shown in Table 6.2,

(ii) Each component was loaded separately. So that the Executive program knows where the other components are, a program specifying their start addresses (see Appendix E-4) was link loaded with the RTX-16 Executive. Similarly, so that the other four components know the key locations in the Executive the routine XLOCS was link loaded with each of them. After a number of trial loads, the optimum useage of core was

achieved and, as will be discussed in more detail later on, the OLDFP Executive was then constructed.

6.6.1. THE HADIOS INPUT/OUTPUT DRIVERS

The OP-16 standard software includes two drivers to access HADIOS namely HAD-DIO, the digital input/output driver, and HAD-ANI, the analogue input driver. Since these programs are more easily requested by programs written in DAP-16 and because they are rather rudimentary in nature, it was decided to write a supervisory program. This program, shown in Appendix E-1, has six functions, five of which can be simply requested from a FORTRAN program using the REQUEST function, see Appendix D-2.

The six functions performed by the supervisory program are,

- H1 - Digital input, input value is stored in location BUFD
- H2 - Digital output, output value should be stored in location BUFD+1.
- H3 - Counter input, input value is stored in location BUFD+2.
- H4 - Preset counter, output value should be stored in location BUFD+3.

H5 - Counter half-full interrupt response code, the time (mil, secs., mins., hrs) at which the interrupt occurs in stored locations BUFD+8,9,10,11.

H6 - Analogue input, analogue channels 0-16 are scanned an ensemble number of times and the average value for each channel is stored in the corresponding element of an array in the Fortran program.

With the exception of function H5, the time (mil, secs., mins., hrs) at which the function was executed is stored in locations BUFD+4,5,6 and 7.

When the half-full counter interrupt option is required the counter should be preset to a value between 0 and 127 using H4 before using the CONNECT INTERRUPT system function to establish the necessary links between the Interrupt Handler and H5.

Prior to the use of the analogue input function the required ensemble should be stored in location ENS1; the default value is 10, also the start address of the Fortran array in which the values of the analogue channels are to be stored should be put into location ADDR5.

The addresses of locations ENS, BUFD and ADDR5 can be specified in Fortran programs by

declaring them to be external variables and then satisfying them at load time by loading an object tape of a program similar to those shown in Appendices E-3 and G-9. Alternatively the utility program can be used to store the values when the OLDFP Executive is running.

6.6.2. FORTRAN PROGRAMS

In order to write the Executive Configuration Module (XCOM) the number and names of the Fortran programs known by the Executive needs to be specified. It was felt that for on-line filtering experiments two Fortran programs would be required, the first of them, called IN, to initialise the various parameters associated with Kalman Filtering and data acquisition, and the second, called HT, to perform the periodic tasks of filtering and data acquisition. Both of these programs will be discussed in chapter 9.

6.6.3. THE EXECUTIVE CONFIGURATION MODULE (XCOM)

The Executive Configuration Module (XCOM) consists of a series of tables written by the user. These tables provide the information required by the executive to operate the system and are

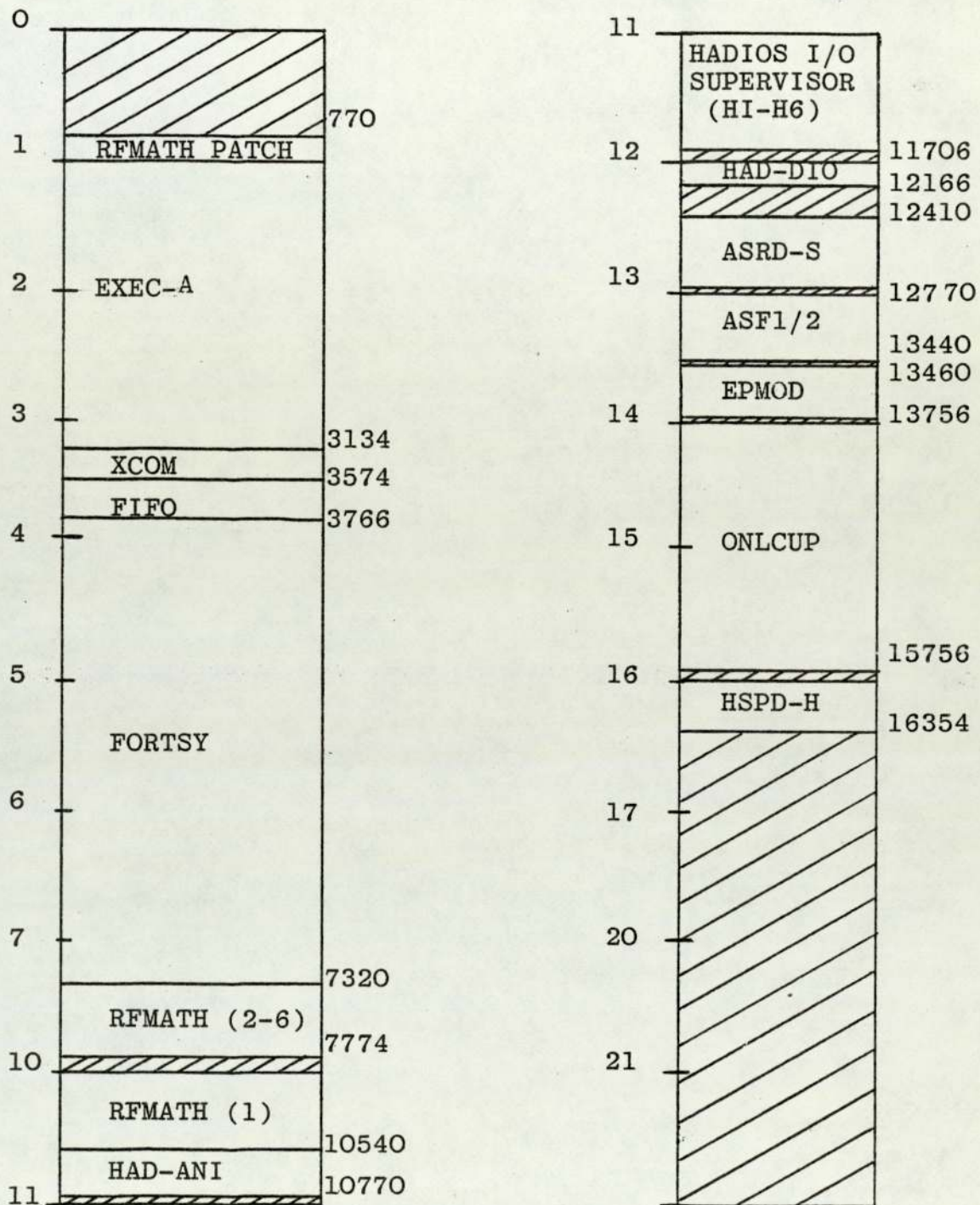
described in detail in the manufacturer's user guide (6.8). A listing and brief description of the XCOM module used in the OLDFP Executive is given in Appendix E-2.


6.6.4. CONSTRUCTION OF THE OLDFP EXECUTIVE

The detailed loading procedures for the five sections of the OLDFP Executive shown in Table 6.2 are given in Appendices E4 to E8. Having loaded all of these programs into core as described and made the necessary corrections to the standard software, see Appendix E-9, the punch and load program is used to produce an SLST of the complete executive program, locations 770_8 to 16377_8 . The layout of the various sections of the executive is shown in Figure 6.3.

6.6.5. OPERATING PROCEDURE

The OLDFP Executive is started up by setting the P/Y register of the computer to 1000_8 and pressing the start button. The Executive then performs its own initialisation procedures following which the user should enter the utility program and make the following modifications,



 INDICATES VACANT CORE

LAYOUT OF THE OLDFP EXECUTIVE

FIGURE 6.3.

(i) Location 3241_8 should be set to the first executable instruction of HT.

(ii) Location 3246_8 should be set to the first executable instruction of IN.

(iii) Any other address links, see section 6.6.1, should be established.

The OLDFP Executive is now capable of running two Fortran programs (HT and IN) and servicing input and output to the teletype, the high speed punch and HADIOS. In addition the utility program (ONLCUP) provides a conversational interface between the operator and the executive. The system errors which may be detected whilst the OLDFP executive is running are tabulated in Appendix E-10.

In order to assess the reliability of the OLDFP Executive a dynamic logging program (program HT) was written in Extended Honeywell Fortran. This program includes all of the features of OP-16 which are to be used in the on-line filtering programs discussed in chapter 9. Apart from testing the OLDFP Executive this program also gives a good example of how to operate the system. A listing of this program together with the print out obtained during a typical operating session are included in Appendix E-13.

6.7. CHAPTER REVIEW

The standard Honeywell software has been discussed with special reference to those programs used to construct the applications packages required by this research. The graphical software and the previously reported Aston Simulation Package were then described and finally the two main on-line applications packages were described. The first of these is the HADIOS EXECUTIVE PACKAGE which is mainly for use where the editing facilities and versatility of the BASIC-16 Interpreter are required. The second package is the O.L.D.F.P. EXECUTIVE which when combined with the Fortran Programs discussed in chapter 9, provides a facility for on-line filtering experiments.

CHAPTER 7

STEADY STATE ANALYSIS OF THE DOUBLE
EFFECT EVAPORATOR

7.1. INTRODUCTION

Before any on-line filtering experiments can be carried out with the double effect evaporator a great deal of preparatory work is required in order to produce both the required statistical information and a mathematical model to describe the transient and steady state behaviour of the evaporator. The work carried out in this area can be conveniently separated into the categories of steady state and dynamic analysis.

This chapter is based around a steady state analysis of the evaporator which has the following objectives:

(i) To produce correlations relating the electrical signals generated by the evaporator instrumentation to the measured process variables.

(ii) An analysis of the measurement statistics under operating conditions is required to produce a suitable value for the $R(k)$ matrix (see section 2.2.6.2).

(iii) An attempt to quantify the heat losses that exist within the evaporator system must be made.

(iv) An equation describing the overall heat transfer coefficient for each of the four items of heat exchange equipment must be obtained.

The fourth objective is of particular importance because, as was discussed previously in the literature survey, the predicted transient response of outlet temperatures from heat exchange equipment is highly susceptible to changes in the heat transfer coefficient.

7.2. INSTRUMENT CALIBRATION

When mechanical/electrical transducers operate in a pilot plant environment, frequent calibrations are required in order to maintain consistent on-line results. Since the electrical signals generated by the instrumentation installed at the evaporator are corrupted by noise the calibrations are carried out on-line to the computer so that large sample sizes can be averaged, thus filtering numerically.

All of the instruments calibrated were assumed to have a linear correlation of the form,

$$y = a + bx \quad - (7.1)$$

where, y is the value of the measured process variable,

x is the corresponding reading obtained from the transducer by the H316/HADIOS data acquisition system,

and, a and b are the intercept and slope of the line respectively.

When sufficient data sets (x, y) have been collected a linear regression is carried out to determine the values of the coefficients ' a ' and

'b'. The BASIC program which was used in conjunction with the Hados Executive Package Mk.2 to obtain the x values is shown in Appendix F1. The computer program used to carry out the linear regression, MULREG, is a library program supplied as standard software with the University's Hewlett Packard 2000 Access Computer (7.1), see Appendix H-9 for the data layout for this program. The individual values of 'a', 'b' and the related correlation coefficients are given for each instrument in Table 7.1.

The methods used to obtain the y values necessary for the linear regressions discussed above differ for each instrument and so they will now be described individually.

7.2.1. THERMOCOUPLES

Measurement of temperature by thermocouple is considerably more accurate than alternative methods against which to calibrate, particularly when an isothermal reference chamber is available as a cold junction. Consequently, calibration is only really necessary to determine the intercept, a, as it can not be predicted accurately due to

TABLE 7.1 - DOUBLE EFFECT EVAPORATOR INSTRUMENT
CALIBRATIONS

Channel [*] No	INTERCEPT (a)	SLOPE (b)	CORRELATION COEFFICIENT	UNITS OF MEASURED VARIABLE
0	-1.2576	0.119093	0.9999787	°C
1	-1.2338	0.119093	0.9999798	°C
2	-1.3624	0.119093	0.9999862	°C
3	-1.2767	0.119093	0.9999851	°C
4	-1.2433	0.119093	0.9999193	°C
5	-1.1433	0.119093	0.9999526	°C
6	-1.2576	0.119093	0.9999698	°C
7	-1.2243	0.119093	0.9999898	°C
8	-1.2719	0.119093	0.9999873	°C
9	-1.1481	0.119093	0.9999746	°C
10	$0.914674 \cdot 10^{-2}$	$0.637724 \cdot 10^{-4}$	0.982832	kg.s ⁻¹
11	-0.1464545	$0.2815899 \cdot 10^{-2}$	0.9985759	M
12	-2.688774	0.2222609	0.999558	kPa
13	-4.021423	0.2164091	0.9996781	kPa
14	98.17401	0.1379926	0.99973226	kPa
15	-9.298042	0.2931805	0.9998976	kPa
16	$-0.2937119 \cdot 10^{-2}$	$0.4728656 \cdot 10^{-4}$	0.9810014	kg.s ⁻¹
17	$0.25 \cdot 10^{-3}$	$0.695 \cdot 10^{-4}$	0.999981	kg.s ⁻¹

*For a full description of all of the evaporator instrumentation see Table 5.2 and Figure 5.4

the nature of the amplifiers installed in the remote signal conditioning cabinet.

The value of the slope, b , was calculated using data supplied by the manufacturer (7.2) as follows.

Temperature (y)	Output*	Reading (x)
0°C	0v	0
100°C	4.1v	840

Thus,

$$b = 0.119093 \quad - (7.2)$$

*The output of each thermocouple is amplified by a gain of 1000.

The value of the intercept, a , corresponding to this slope was then found by collecting a series of values of the analogue reading (x_0) when the thermocouple was immersed in melting ice contained in a vacuum flask. The value of 'a' can then be calculated as follows,

$$a = -0.119093 \cdot x_0 \quad - (7.3)$$

As a check on the accuracy of the manufacturer's data these calibrations were then checked using a portable electronic thermocouple having an accuracy of $\pm 0.1^\circ\text{C}$.

7.2.2. DIFFERENTIAL PRESSURE CELLS

The differential pressure cell used for measuring the steam flow into the first effect is used in conjunction with an orifice plate. The pressure drop across the orifice plate was measured by use of a rotary differential pressure gauge; this reading being recorded along with the corresponding analogue signal generated by the differential pressure cell.

The measured pressure drop was then used to calculate a theoretical steam flowrate by following the procedure given by the relevant British Standard (7.3). The values of 'a' and 'b' were then calculated by carrying out a linear regression on the analogue readings (x) and the calculated flowrates (y). As a final check, the amount of condensate coming from the first effect shell was measured after the evaporator had been operating at steady state for some time. This flowrate was then compared with that predicted by the correlation and after ten such checks an agreement of $\pm 3\%$ was obtained.

The correlation for the differential pressure cell used for measuring the height of liquid in the

second effect separator was determined by recording the analogue readings at ten known liquid heights.

7.2.3. STRAIN GAUGE PRESSURE TRANSDUCERS

To determine the correlations for the three pressure transducers associated with vacuum pressures, analogue readings were recorded at regular intervals in the range 7 to 101 kPa. Before taking the readings the vacuum pump was switched on so that the pressure in the evaporator could be reduced to the desired level. The evaporator was then sealed off from the atmosphere and the vacuum pump was switched off. After a few minutes observation to ensure that the reading was steady, the pressure was measured using a mercury in glass manometer connected to the relevant evaporator unit.

The transducer associated with the steam pressure on the shellside of the first effect was calibrated by allowing a flow of steam into the first effect and when steady state had been reached, sealing off the shell from the steam supply and the atmosphere, and then measuring the pressure with a C-spring bourdon tube pressure gauge.

7.2.4. FLOWMETERS

Both the variable area magnetic flowmeter and the turbine flowmeter were calibrated by disconnecting the respective downstream pipe and measuring the flowrate through the flowmeter using a stop watch and a two litre measuring cylinder. Whilst this measurement was being taken the temperature of the water leaving the flowmeter was measured so that the liquid density could be found and hence the volumetric flowrate converted to a mass flowrate.

7.3. ON-LINE STEADY STATE EXPERIMENTS

Within the evaporator system there is rarely a situation of true steady state due to the fluctuating level in the second effect separator and the random disturbances on the inputs to the system. However, it can reasonably be assumed that despite the above fluctuations all other variables will remain constant once thermal equilibrium has been reached within the evaporator. The validity of this assumption can be demonstrated by an examination of the standard deviations of the temperature measurements when data is collected every three seconds over a five minute time interval.

7.3.1. MATHEMATICAL MODEL

The steady state mathematical model of the evaporator has been developed using the process notation discussed in section 5.2.3. Thus, the following subscripted variables are used:

T = temperature ($^{\circ}\text{C}$)

V = vapour flowrate ($\text{kg}\cdot\text{s}^{-1}$)

M = liquid flowrate ($\text{kg}\cdot\text{s}^{-1}$)

E = enthalpy ($\text{kJ}\cdot\text{s}^{-1}$)

In addition the following unsubscripted variable is used in the model:

H1 = the accumulation in the second effect separator ($\text{kg}\cdot\text{s}^{-1}$)

The following assumptions are made:

(i) The specific heat of the liquid is a constant denoted by C_p , where $C_p = 4.1868 \text{ kJ}\cdot\text{kg}^{-1}\cdot\text{°C}^{-1}$.

(ii) The saturated vapour enthalpy can be represented by the function $G(T)$ ($\text{kJ}\cdot\text{kg}^{-1}\cdot\text{°C}^{-1}$), where,

$$G(T) = C_p T + \lambda(T) \quad - (7.4)$$

where $\lambda(T)$ is the latent heat of vapourisation at temperature T in $\text{kJ}\cdot\text{kg}^{-1}$.

(iii) There is no mass loss from the system.

(iv) Where vapour and liquid mixtures exist they do so at the saturated vapour temperature corresponding to the operating pressure.

(v) The heat exchanger shells are well mixed regions so that the exit and shell temperatures are equal.

(vi) Heat transfer from the heat exchanger shells is due to vapour condensation alone.

(vii) All steam entering the first effect shell condenses to liquid at a temperature of 100°C .

$$\text{Thus, } M_5 = V_6 \quad - (7.5)$$

$$T_5 = 100^\circ\text{C} \quad - (7.6)$$

(viii) The cyclone separator is 100% efficient, there is no condensation at this point in the system and the separation is isothermal.

$$\text{Thus, } M_8 = M_7 \quad - (7.7)$$

$$V_3 = V_7 \quad - (7.8)$$

$$T_3 = T_8 = T_7 \quad - (7.9)$$

(ix) Liquid in the second effect separator is well mixed and at a uniform temperature.

$$\text{Thus, } T_{\text{Hl}} = T_{15} \quad - (7.10)$$

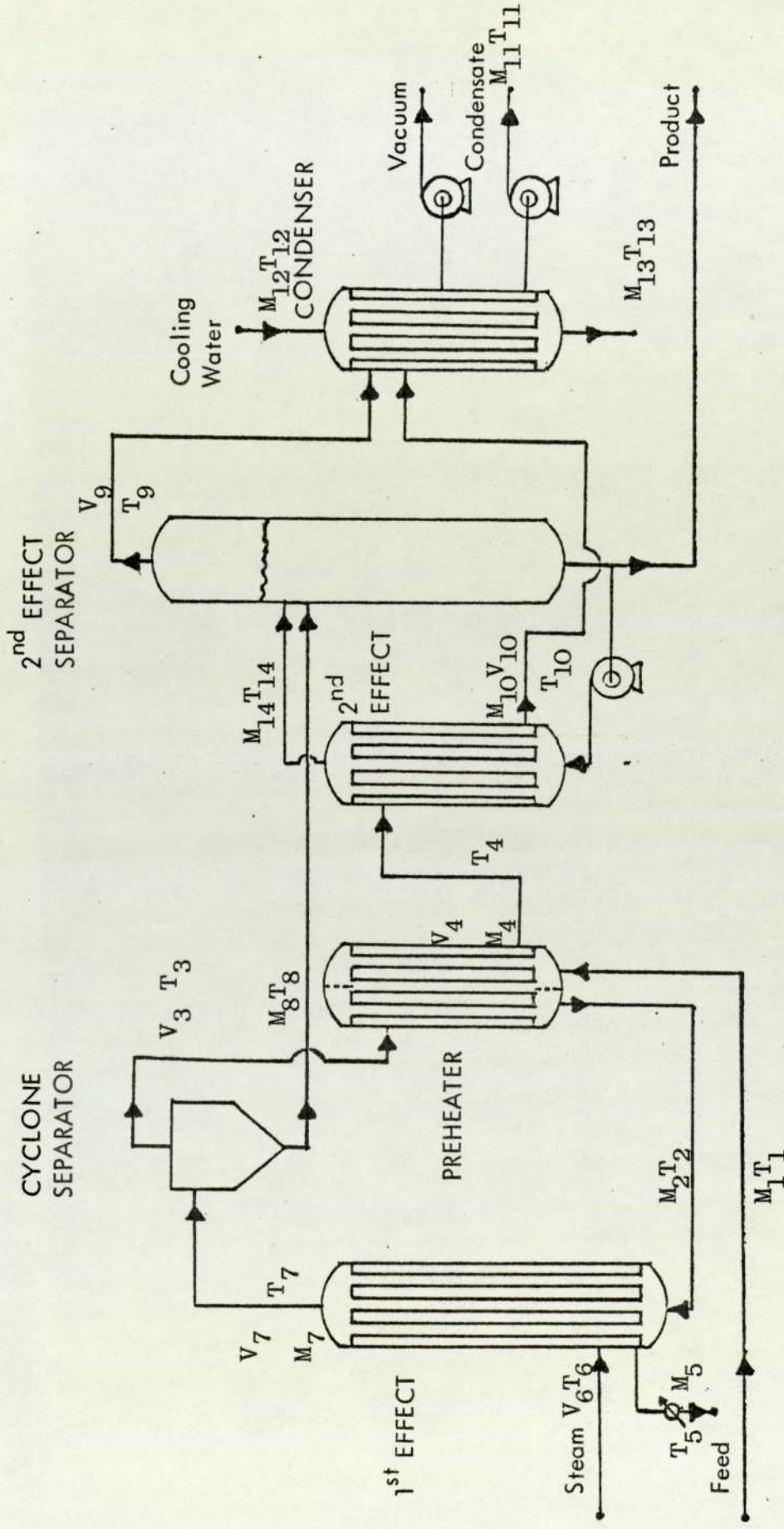
where, T_{Hl} = the temperature of the accumulated liquid.

(x) The circulation pump provides a constant circulation through the tubes of the second effect calandria.

$$\text{Thus, } M_{14} = M_{15} \quad - (7.11)$$

The steady state mathematical model of the evaporator can now be written out as a series of mass and energy balances over each heat exchange unit. The relevant process variables associated with the steady state mathematical model are shown in Figure 7.1.

FIGURE 7.1



DOUBLE EFFECT EVAPORATOR

STEADY STATE MODEL PROCESS VARIABLES

7.3.1.1. MASS BALANCES

PREHEATER tubes $M_1 = M_2$ - (7.12)

shell $V_3 = M_4 + V_4$ - (7.13)

FIRST EFFECT tubes $M_2 = V_7 + M_7$ - (7.14)

shell $V_6 = M_5$ - (7.15)

SECOND EFFECT tubes $M_{15} = M_{14}$ - (7.16)

shell $M_4 + V_4 = M_{10} + V_{10}$ - (7.17)

SECOND EFFECT SEPARATOR $M_{14} + M_8 = V_9 + M_{15} + H_L$
- (7.18)

CONDENSER tubes $M_{12} = M_{18}$ - (7.19)

shell $M_{10} + V_{10} + V_9 = M_{11}$
- (7.20)

7.3.1.2. ENERGY BALANCES

PREHEATER $E_1 + E_3 = E_2 + E_4$
- (7.21)

FIRST EFFECT $E_2 + E_6 = E_7 + E_5$
- (7.22)

SECOND EFFECT $E_4 + E_{15} = E_{14} + E_{10}$
- (7.23)

SECOND EFFECT SEPARATOR $E_{14} + E_8 = E_9 + E_{15} + E_{H_L}$
- (7.24)

CONDENSER $E_{12} + E_9 + E_{10} = E_{13} + E_{11}$
- (7.25)

In equation 7.24 E_{HI} is used to denote the enthalpy of the accumulation in the second effect separator.

Based on the assumptions made and taking the datum temperature as 0°C , the enthalpy terms are defined by,

$$E_1 = M_1 \cdot C_p \cdot T_1 \quad - (7.26)$$

$$E_2 = M_2 \cdot C_p \cdot T_2 \quad - (7.27)$$

$$E_3 = V_3 \cdot G(T_3) \quad - (7.28)$$

$$E_4 = V_4 \cdot G(T_4) + M_4 \cdot C_p \cdot T_4 \quad - (7.29)$$

$$E_5 = M_5 \cdot C_p \cdot T_5 \quad - (7.30)$$

$$E_6 = V_6 \cdot G(T_6) \quad - (7.31)$$

$$E_7 = V_7 \cdot G(T_7) + M_7 \cdot C_p \cdot T_7 \quad - (7.32)$$

$$E_8 = M_8 \cdot C_p \cdot T_8 \quad - (7.33)$$

$$E_9 = V_9 \cdot G(T_9) \quad - (7.34)$$

$$E_{10} = V_{10} \cdot G(T_{10}) + M_{10} \cdot C_p \cdot T_{10} \quad - (7.35)$$

$$E_{11} = M_{11} \cdot C_p \cdot T_{11} \quad - (7.36)$$

$$E_{12} = M_{12} \cdot C_p \cdot T_{12} \quad - (7.37)$$

$$E_{13} = M_{13} \cdot C_p \cdot T_{13} \quad - (7.38)$$

$$E_{14} = M_{14} \cdot C_p \cdot T_{14} \quad - (7.39)$$

$$E_{15} = M_{15} \cdot C_p \cdot T_{15} \quad - (7.40)$$

$$E_{HI} = H_1 \cdot C_p \cdot T_{HI} \quad - (7.41)$$

Thus, the steady state mathematical model of the double effect evaporator consists of the 37 equations defined by 7.5 to 7.41.

7.3.2. THERMODYNAMIC CORRELATIONS

Before attempting the solution of a mathematical model associated with the evaporator a number of correlations relating physical and thermodynamic properties with process variables need to be developed. Since all of the models of the evaporator are to be solved on-line in real time, it is convenient to correlate the available data into algebraic equations, thus avoiding the storage and interpolation of large arrays of physical property data. For the liquid and solid phases the relevant physical properties are assumed constant, see Table 7.2, but for the vapour phase, the variation of pressure, density, latent heat and enthalpy with temperature is greater and must be considered.

Pressure - temperature

In on-line steady state and dynamic experiments, it is necessary to obtain a correlation relating the temperature of steam to its pressure. An

TABLE 7.2 - LIQUID AND SOLID PHASE PHYSICAL PROPERTIES
RELEVANT TO THE DOUBLE EFFECT EVAPORATOR

PROPERTY	VALUE
C_p -Specific heat of liquid water	4.1868 $\text{kJ.kg}^{-1}.\text{°C}^{-1}$
C_t -Specific heat of copper- tubes	0.38494 $\text{kJ.kg}^{-1}.\text{°C}^{-1}$
C_s -Specific heat of mild steel- shells	0.45186 $\text{kJ.kg}^{-1}.\text{°C}^{-1}$
ρ_L -Density of liquid water	985.22 kg.m^{-3}
C_v -Specific heat of steam	1.9 $\text{kJ.kg}^{-1}.\text{°C}^{-1}$

algorithm has been developed by Richards (7.4) where the pressure P (kPa) is related to temperature T(K) by the function,

$$P = P_s \cdot \exp (13.3185 \cdot \bar{T} - 1.976 \cdot \bar{T}^2 - 0.6445 \cdot \bar{T}^3 - 0.1299 \cdot \bar{T}^4) \quad - (7.42)$$

where, $\bar{T} = 1 - \frac{T_s}{T}$

and, T_s and P_s are the temperature and pressure of saturated steam at atmospheric pressure (373.15 K and 101.325 kPa)

To calculate T when P is given, a first estimate of \bar{T} is formed by ignoring all but the first term of the polynomial in equation 6.2.1,

$$\bar{T}_1 = \frac{\log_e \left(\frac{P}{101.325} \right)}{13.3185} \quad - (7.43)$$

and successively more accurate estimates are obtained by recursive calculations,

$$\bar{T}_{n+1} = \frac{\bar{T}_1 + ((0.1299 \bar{T}_n + 0.6445) \cdot \bar{T}_n + 1.976) \bar{T}_n^2}{13.3185} \quad - (7.44)$$

when, for some small value of n, $|\bar{T}_{n+1} - \bar{T}_n| < E$, where E is a small constant, the desired estimate of T ($^{\circ}\text{C}$) can be found as follows,

$$T = \frac{373.15}{(1 - \bar{T}_{n+1})} - 273.15 \quad - (7.45)$$

Appendix F-2 shows a Basic program written to test the above algorithm and a print out of some of the calculated values.

Density - temperature

Using density - temperature data taken from steam tables (7.5) a linear regression analysis was carried out using the MULREG computer program (7.1). The best fit was obtained by correlating the natural logarithms of the data. The resulting equation is,

$$\log_e (\rho_v(T).1000) = 1.93.\log_e T - 3.1487$$

- (7.46)

where, ρ_v is the vapour density (kg.m^{-3}) and the correlation coefficient is 0.9935.

Latent heat - temperature

The resulting equation from a similar linear regression analysis on latent heat - temperature data is,

$$\lambda(T) = 2501.64 - 2.407.T \quad - (7.47)$$

where, λ is the latent heat (kJ.kg^{-1}) and the correlation coefficient is 0.9997.

Enthalpy - temperature

The relationship between vapour enthalpy and temperature is given by,

$$G(T) = C_p \cdot T + \lambda(T)$$

where, $G(T)$ is the vapour enthalpy ($\text{kJ} \cdot \text{kg}^{-1}$) and

C_p , the specific heat, is $4.1868 \text{ kJ} \cdot \text{kg}^{-1} \cdot ^\circ\text{C}^{-1}$.

Combining the above equation and equation 7.47, the resulting equation is,

$$G(T) = 1.7798 \cdot T + 2501.64 \quad - (7.48)$$

7.3.3. SOLUTION OF THE STEADY STATE MODEL

Substitution of equations 7.26 to 7.41 into 7.21 to 7.25 leaves a mathematical model consisting of 14 linear simultaneous equations which contain 35 process variables. The following measurements are available from the double effect evaporator:

Direct Measurements:- $T_1, T_2, T_3, T_4, T_8, T_{11}, T_{12}, T_{13}, T_{14}$

$T_{15}, M_2, M_8, V_6, H.$

Indirect Measurements (temperatures obtained from pressures by using the temperature-pressure correlation):- $T_6, T_9, T_{10}.$

Off-line Measurements:- $M_{12}, M_{11}, M_5.$

Thus, since $M_5 = V_6$ (equation 7.15), 19 measurements are available leaving 16 unknown process variables: If we now take into account assumptions (vii), (viii) and (ix) the mathematical model now consists of 19 equations containing 16 unknowns. This leads to a redundancy of either three equations or three measurements.

In this particular solution the measurement of height in the second effect separator, H , is discarded as it is known to be a noisy variable. This fact is particularly noticeable at high steam flowrates when the boiling which takes place in the second effect separator makes it almost impossible to make an accurate measurement of height. In order to determine the heat loss from the first effect of the evaporator the measurement of steam flowrate, V_6 , is also discarded. This leaves one redundant equation which was found by examination to be equation 7.25. Thus, the steady state model of the evaporator now consists of 18 linear simultaneous equations containing the following 18 unknown process variables,

$M_1, M_4, M_5, M_7, M_{10}, M_{13}, M_{14}, M_{15}, V_3, V_4, V_6, V_7, V_9, V_{10},$
 $T_5, T_7, T_H, H_1.$

Solution of the steady state model now proceeds as follows. The equations associated with the first effect are solved in a stepwise manner, i.e. by elimination and substitution.

From equation 7.12,

$$M_1 = M_2 \quad - (7.49)$$

From equation 7.14 and assumption (viii),

$$V_3 = M_2 - M_8 \quad - (7.50)$$

$$M_7 = M_8 \quad - (7.51)$$

$$V_7 = V_3 \quad - (7.52)$$

From equation 7.13 and 7.21,

$$M_4 = (E_1 + E_3 - E_2 - V_3 \cdot G(T_4)) / (C_p \cdot T_4 - G(T_4)) \quad - (7.53)$$

From equation 7.13,

$$V_4 = V_3 - M_4 \quad - (7.54)$$

Since T_3 is rarely exactly the same as T_8 , equation 7.9 is modified as follows,

$$T_7 = (T_3 + T_8) / 2 \quad - (7.55)$$

From equations 7.22 and 7.15,

$$V_6 = (E_7 - E_2) / (G(T_6) - C_p \cdot T_5) \quad - (7.56)$$

From equation 7.15,

$$M_5 = V_6 \quad - (7.57)$$

Finally, using assumption (vii),

$$T_5 = 100^\circ\text{C} \quad - (7.58)$$

The heat loss from the first effect can now be calculated using the measured value of the steam flowrate,

$$\begin{aligned} \text{Heat Loss} &= (V_6(\text{measured}) - V_6(\text{calculated})) \\ &\cdot G(T_6) \end{aligned} \quad - (7.59)$$

The unknown process variables associated with the second effect can now be determined by setting up a set of five simultaneous equations and then solving them by Gauss-Jordan Elimination.

From equation 7.20,

$$M_{10} + V_{10} + V_9 = M_{11} \quad - (7.60)$$

From equation 7.17,

$$M_{10} + V_{10} = M_4 + V_4 \quad - (7.61)$$

From equations 7.23 and 7.16,

$$\begin{aligned} M_{14} C_p (T_{14} - T_{15}) + M_{10} \cdot C_p \cdot T_{10} + V_{10} \cdot G(T_{10}) \\ = E_4 \end{aligned} \quad - (7.62)$$

From equations 7.24 and 7.16, and assumption (ix),

$$M_{14} \cdot C_p \cdot (T_{15} - T_{14}) + V_9 \cdot G(T_9) + M_A \cdot C_p \cdot T_{15} = E_8 \quad - (7.63)$$

From equations 7.18 and 7.16,

$$V_9 + H_1 = M_8 \quad - (7.64)$$

The right hand side of equations 7.60 and 7.64 contain only measured or previously calculated variables and so these equations can be solved simultaneously for the unknown process variables,

$$M_{10}, M_{14}, V_9, V_{10}, H_1.$$

To complete the solution, from equation 7.16,

$$M_{15} = M_{14} \quad - (7.65)$$

From assumption (ix),

$$T_H = T_{15} \quad - (7.66)$$

From equation 7.19,

$$M_{13} = M_{12} \quad - (7.67)$$

The heat loss from the second effect can now be determined by using equation 7.25 to provide a calculated value for M_{11} ,

$$\text{Heat Loss} = (M_{11}(\text{measured}) - M_{11}(\text{calculated}))$$

$$*C_p * T_{11} \quad - (7.68)$$

7.3.4. DISCUSSION OF STEADY STATE COMPUTER
PROGRAM AND EXPERIMENTAL PROCEDURE

The computer program for acquiring the experimental data and solving the steady state model as shown by equations 7.49 to 7.68 is listed in Appendix F-3 and the flowchart is shown in Figure 7.2. This program is written in the Basic Language and is run in conjunction with the Hados Executive Package Mk.2, see section 6.5.

Once the experimental equipment has been set up as discussed in Chapter 5, the evaporator is allowed to come to steady state for a period of time not less than twenty minutes. The steady state computer program is then set into operation and after the date, the time, the interval between scans and the total number of scans in the experiment have been entered as data via the VDU, a message informing the user that the experiment has started is printed out and data acquisition commences.

During the experimental period accurate measurements of the cooling water flowrate (M_{12}), the condensate flowrate (M_{11}) and the steam

FLOWCHART OF ON-LINE STEADY STATE LOGGING PROGRAM

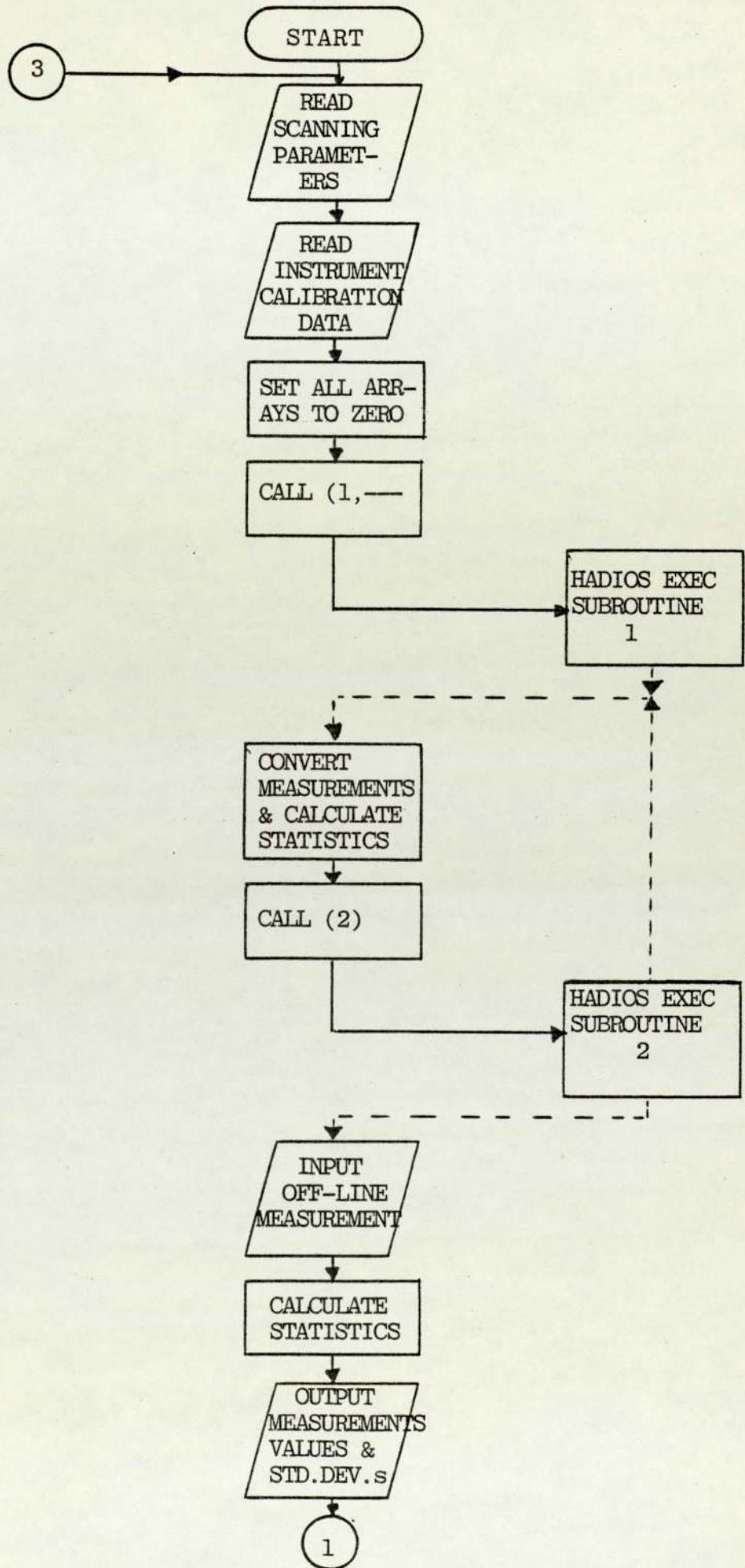


FIGURE 7.2

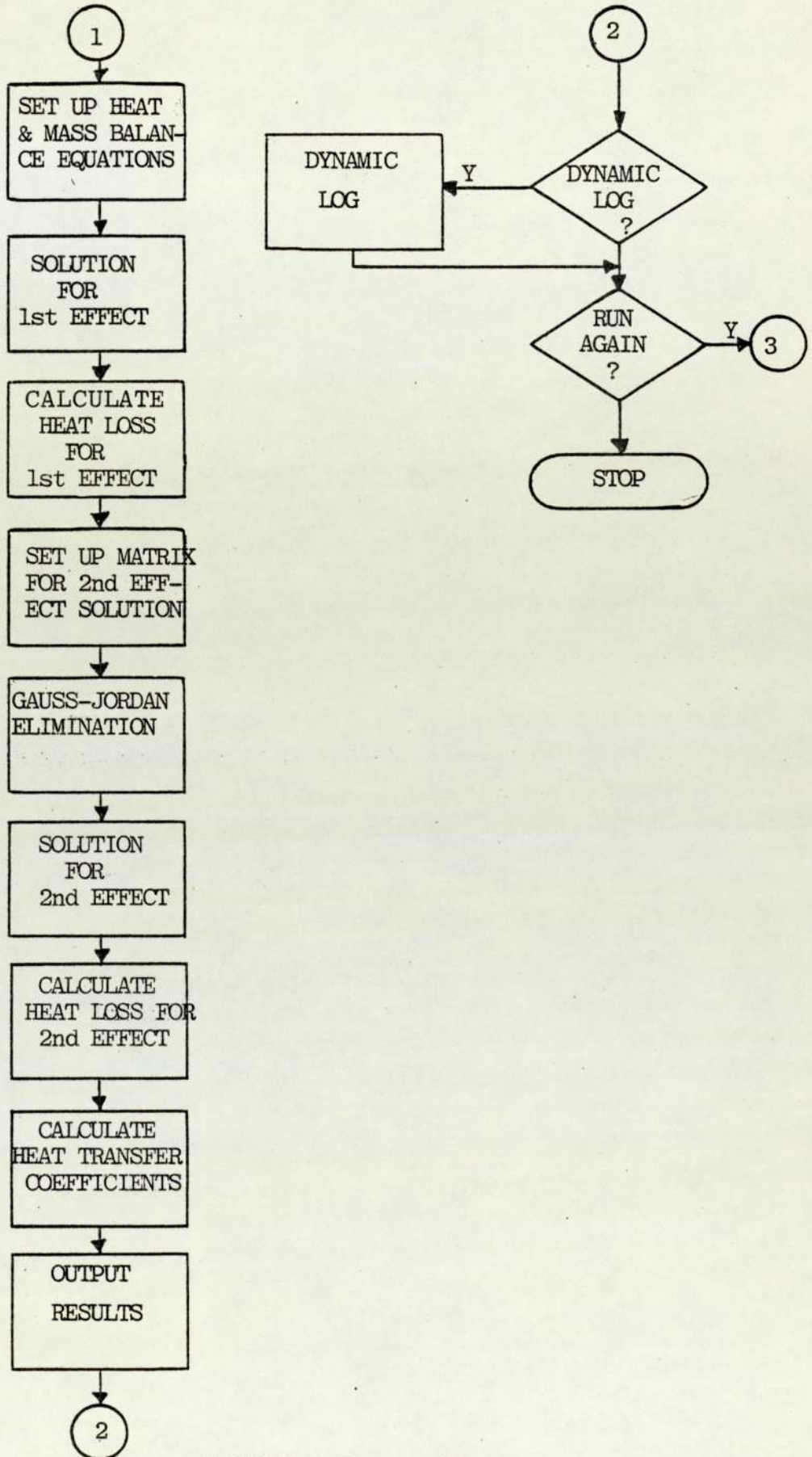


FIGURE 7.2 cont'd

condensate flowrate (M_5) are taken manually at the evaporator. The cooling water flowrate is measured using a calibrated rotameter and the condensate and steam condensate flowrates by collection and measurement over a known time interval. Three values of each measurement are taken and the average values used in the subsequent calculations.

When the required number of scans have taken place a message informing the user of this fact is printed out and the off-line measurements are then entered into the program via the VDU. The program then calculates the average values and the standard deviations of the measured variables and prints out a table of these values on the VDU. It is at this time that the validity of the steady state assumption is checked. The thermocouples installed at the evaporator for the purpose of temperature measurement are very accurate and if the standard deviations associated with these instruments are large then the evaporator is considered not to have been at steady state and the experiment is terminated.

If the standard deviations of the temperature are acceptable the program continues to solve the

heat and mass balances of the evaporator and then to print out the results. An example of the computer printout obtained during a steady state experiment is given in Table 7.3.

7.3.5. RESULTS AND DISCUSSION

Using the procedure described above a total of 70 experiments were carried out to enable an analysis of the steady state behaviour of the evaporator. These experiments were conducted at 35 different steady state situations to enable a check to be made on the repeatability of the results obtained. The results obtained in these 35 sets of experiments are summarised in Table 7.4 and shown for each individual heat transfer unit of the evaporator in Appendices H-1 to H-4.

7.3.5.1. HEAT TRANSFER CORRELATIONS

For the general heat exchanger with an isothermal condensing vapour in the shell and no change of phase in the fluid flowing through the tubes, the experimental overall heat transfer

TABLE 7.3 - EXAMPLE PRINTOUT FROM ON-LINE STEADY

STATE LOGGING PROGRAM

STEADY STATE LOGGING PROGRAM - PAGE 0
 =====

INPUT DATE (DAYS, MNTHS., YRS.) ! 22, 2, 77
 INPUT TIME (HRS., MINS., SECS.) ! 14, 35, 0
 INPUT INTERVAL BETWEEN SCANS (SECS.) ! 3
 INPUT TOTAL NUMBER OF SCANS ! 100
 INPUT 1 IF HARDCOPIES REQUIRED ! 0

SCANNING HAS STARTED

SCANNING HAS FINISHED

INPUT COOLING WATER RATE (KG/MIN) ! 25.9
 INPUT CONDENSATE RATE (SECS/KG) ! 28
 INPUT STEAM CONDENSATE RATE (SECS/KG) ! 42

STEADY STATE LOGGING PROGRAM - PAGE 1
 =====

DATE 22 / 2 / 77
 TIME 14 : 35 : 0

CHANNEL	MEASUREMENT	CONV. VALUE	STD. DEV.
0	139.946	15.409	.718155E-01
1	279.822	32.091	.924388E-01
2	193.432	21.674	.382733E-01
3	525.421	61.2973	.911086E-01
4	502.715	58.6266	.382733E-01
5	534.716	62.5376	.382733E-01
6	648.033	75.9184	.159344
7	421.601	48.9855	0
8	645.257	75.5739	.143206
9	646.913	75.8945	.159344
10	218.091	.230549E-01	.225035E-03
11	647.735	1.6775	.103119E-01
12	211.347	44.2853	.111585
13	120.293	22.0109	.184878
14	321.558	142.547	.631219
15	137.026	30.8753	.653641E-01
16	613.613	.260785E-01	.494441E-04
17	663.75	.463806E-01	.340821E-02

TABLE 7.3 cont'd

STEADY STATE LOGGING PROGRAM - PAGE 2

DATE 22 / 2 / 77
 TIME 14 : 35 : 0

STREAM	LIQUID (KG/S)	VAPOUR (KG/S)	TEMP. (DEG)	ENTHALPY (KJ/S)
1	.463806E-01	0	21.674	4.2088
2	.463806E-01	0	61.2973	11.9031
3	0	.203021E-01	75.8945	53.5291
4	.33162E-02	.169859E-01	75.5739	45.8348
5	.200079E-01	0	100	8.37691
6	0	.200079E-01	109.854	53.9624
7	.260785E-01	.203021E-01	75.9065	57.4885
8	.260785E-01	0	75.9184	8.28919
9	0	.154122E-01	62.1703	40.26
10	.181492E-01	.215288E-02	73.4202	11.2459
11	.357143E-01	0	48.9855	7.32472
12	.431667	0	15.409	27.8486
13	.431667	0	32.091	57.9982
14	2.11234	0	62.5376	553.08
15	2.11234	0	58.6266	518.491
16	.106663E-01	0	58.6266	2.61813

STEADY STATE LOGGING PROGRAM - PAGE 3

DATE 22 / 2 / 77
 TIME 14 : 35 : 0

COMPARISON OF MEASURED & CALCULATED VALUES

	MEASURED	CALCULATED	
CONDENSATE	.357143E-01	.39814E-01	(KG/S)
STEAM	.230549E-01	.200079E-01	(KG/S)
STEAM COND.	.238095E-01	.200079E-01	(KG/S)
ACCUMULATION	.731064E-01	.106663E-01	(KG/S)

HEAT LOSSES

HEAT LOSS FROM FIRST EFFECT	1.02531	(KJ/S)
HEAT LOSS FROM SECOND EFFECT	.840813	(KJ/S)
TOTAL HEAT LOSS	1.86613	(KJ/S)
HEAT LOSS FROM EVAPORATOR	2.16942	(%)

HEAT TRANSFER COEFFICIENTS

PREHEATER	.678891	(KW/M ² *K)
1ST. EFFECT	1.74303	(KW/M ² *K)
2ND. EFFECT	2.68751	(KW/M ² *K)
CONDENSER	.253064	(KW/M ² *K)

TABLE 7.4 - SUMMARY OF ON-LINE STEADY STATE EXPERIMENTS

No	FEED RATE ($\text{kg}\cdot\text{s}^{-1}$)	STEAM RATE ($\text{kg}\cdot\text{s}^{-1}$)	COOLING WATER ($\text{kg}\cdot\text{s}^{-1}$)	2nd EFFECT CIRCULATION ($\text{kg}\cdot\text{s}^{-1}$)	ACCUMULATION ($\text{kg}\cdot\text{s}^{-1}$)	FEED TEMP ($^{\circ}\text{C}$)	CYCLONE TEMP ($^{\circ}\text{C}$)	PREHEATER SHELL TEMP ($^{\circ}\text{C}$)	2nd EFFECT SHELL TEMP ($^{\circ}\text{C}$)	2nd EFFECT SEPARATOR TEMP ($^{\circ}\text{C}$)	CONDENSER SHELL TEMP ($^{\circ}\text{C}$)
1	.0315	.0139	.4333	1.931	.0083	18.25	72.42	71.22	63.47	50.50	48.73
2	.0315	.0126	.4333	2.020	.0088	18.21	72.52	71.35	58.29	50.77	49.15
3	.0315	.0142	.4333	1.922	.0068	18.22	72.57	71.43	63.84	50.84	49.22
4	.0312	.0136	.4333	1.977	.0080	18.18	72.60	71.48	63.87	50.88	49.26
5	.0315	.0154	.4417	2.071	.0065	17.90	73.29	71.87	63.68	54.22	49.11
6	.0315	.0156	.4417	2.071	.0065	17.97	73.49	72.03	63.93	54.61	49.42
7	.0315	.0161	.4417	1.912	.0065	18.01	73.81	72.34	64.43	54.67	49.18
8	.0315	.0172	.4417	1.922	.0065	18.09	73.61	72.14	64.14	54.78	49.18
9	.0315	.0173	.4417	1.914	.0065	18.06	73.64	72.18	64.17	54.83	48.95
10	.0471	.0189	.4417	2.047	.0023	17.69	70.88	69.39	60.42	52.59	48.82
11	.0471	.0169	.4417	1.906	.0023	17.69	70.83	69.24	60.37	52.65	48.88
12	.0471	.0201	.4417	1.838	.0023	17.69	70.89	69.20	60.35	52.72	48.72
13	.0471	.0181	.4417	2.243	.0023	17.69	70.97	69.30	60.38	52.78	48.69
14	.0471	.0181	.4417	2.251	.0023	17.63	71.03	69.35	60.38	52.86	48.65
15	.0315	.0184	.4350	2.244	.0071	17.24	75.83	74.28	65.85	55.79	49.45
16	.0315	.0194	.4350	2.061	.0071	17.24	75.94	74.37	65.91	55.90	49.80
17	.0315	.0193	.4350	2.076	.0071	17.22	75.99	74.39	65.89	55.92	50.03

TABLE 7.4 - continued

18	.0315	.0197	.4350	1.998	.0071	17.22	76.06	74.43	65.96	55.92	50.01
19	.0315	.0191	.4350	1.992	.0071	17.23	76.03	74.38	65.93	55.98	50.06
20	.0471	.0194	.4537	1.994	.0021	16.98	75.50	73.79	64.27	55.78	49.83
21	.0471	.0195	.4537	1.995	.0021	16.97	75.33	73.71	64.19	55.87	49.74
22	.0471	.0192	.4537	1.966	.0021	16.99	75.31	73.68	64.17	55.87	49.77
23	.0471	.0210	.4537	1.941	.0021	16.87	75.39	73.85	64.30	56.09	49.85
24	.0471	.0205	.4537	1.948	.0021	16.85	75.36	73.74	64.21	56.10	49.93
25	.0463	.0253	.4367	1.991	.0020	15.33	78.57	75.46	66.04	57.04	50.22
26	.0463	.0246	.4367	1.951	.0019	15.36	79.07	76.01	66.42	57.81	50.56
27	.0463	.0231	.4367	1.949	.0019	15.39	79.72	76.75	67.17	58.82	50.91
28	.0463	.0253	.4367	1.980	.0020	15.42	80.05	77.18	67.68	59.23	50.93
29	.0463	.0259	.4367	1.963	.0020	15.44	80.30	77.53	68.03	59.58	51.09
30	.0463	.0231	.4467	2.034	.0019	15.44	80.56	77.94	68.37	59.90	51.18
31	.0315	.0213	.4467	2.065	.0083	15.97	69.41	67.52	59.19	52.49	48.68
32	.0315	.0187	.4467	1.944	.0083	16.04	68.90	67.13	58.86	52.09	48.86
33	.0315	.0182	.4467	1.935	.0083	16.08	68.63	66.94	58.64	51.96	48.89
34	.0315	.0176	.4467	1.891	.0083	16.06	68.50	66.86	58.59	51.89	48.61
35	.0315	.0166	.4467	2.004	.0083	16.09	68.41	66.76	58.50	51.76	48.52

coefficient is given by,

$$U = \frac{Q}{A \cdot \Delta t} \quad - (7.69)$$

where, U is the overall heat transfer coefficient
(kW.M⁻².K⁻¹)

A is the total surface area of the outside
of the tubes (M²),

Q is the rate of heat transfer (kW)

and ΔT is the temperature driving force (K).

The experimental overall heat transfer coefficients are calculated for each heat exchange unit of the double effect evaporator during the execution of the steady state computer program and the values determined are printed out after the steady solution, see Table 7.3. The surface area of the tubes, A, was calculated from engineering drawings, see Table 5.1, and the heat transferred, Q, and the temperature driving force, Δt, are calculated from the solution of the steady state model of the evaporator as follows:

PREHEATER (PH)

$$Q = M_1 * C_p * (T_2 - T_1) \quad - (7.70)$$

$$\Delta t = \frac{(T_2 - T_1)}{\text{LOG}\left(\frac{T_4 - T_1}{T_4 - T_2}\right)} \quad - (7.71)$$

FIRST EFFECT (FE)

$$Q = V_6 * (C_p * (T_6 - T_5) + \lambda (T_6)) \quad - (7.72)$$

$$\Delta t = \frac{(T_5 + T_6)}{2} - \frac{(T_7 + T_2)}{2} \quad - (7.73)$$

SECOND EFFECT (SE)

$$Q = M_{15} * C_p * (T_{14} - T_{15}) \quad - (7.74)$$

$$\Delta t = \frac{T_{14} - T_{15}}{\text{LOG}\left(\frac{T_{10} - T_{15}}{T_{10} - T_{14}}\right)} \quad - (7.75)$$

CONDENSER (CD)

$$Q = M_{12} * C_p * (T_{13} - T_{12}) \quad - (7.76)$$

$$\Delta t = \frac{T_{13} - T_{12}}{\text{LOG}\left(\frac{T_{11} - T_{12}}{T_{11} - T_{13}}\right)} \quad - (7.77)$$

The experimental heat transfer coefficients calculated in this way are tabulated in Appendix H. Theoretical heat transfer coefficients were calculated for the general heat exchanger by

the method of McAdams (7.6). For heat exchangers with an isothermal condensing vapour in the shell, the tubeside heat transfer coefficient, h_t , is given by the Seider and Tate equations and the shellside heat transfer coefficient, h_s , is given by the Nusselt equation for condensate films. The clean overall heat transfer coefficient, U , can then be calculated by,

$$U = \frac{h_s \cdot h_t}{h_s + h_t} \quad - (7.78)$$

For the preheater, the second effect calandria and the condenser the heat transfer coefficients calculated using the above approach differed from the corresponding experimental values by $\pm 20\%$. At the first effect the experimental results are consistently three to four times larger than those obtained theoretically. Since the first effect is a two phase climbing film type evaporator and the algorithm for the theoretical heat transfer coefficient is for a general heat exchanger with a single liquid phase in the tubes, the experimental results are very feasible and indeed an improvement in performance of this

magnitude for two phase flow conditions is well within the claims of Kestner's Patent (7.7). The above discussion clearly shows that the use of theoretical values for the heat transfer coefficients is not advisable if accurate results are required. Thus, the results of all the steady state experiments were correlated into linear equations of the type reported by Heidemann et al. (7.8) and Gallatig (7.9). The heat transfer coefficient is assumed to be a function of the arithmetic mean temperature driving force, the shellside vapour flow rate and tubeside liquid flow rate. The resulting equations are as follows:

PREHEATER (PH)

$$U_{PH} = a + b. \left(T_4 - \frac{(T_1 + T_2)}{2} \right) + c.M_1 + d.V_3$$

- (7.79)

FIRST EFFECT (FE)

$$U_{FE} = a + B. \left(T_5 - \frac{(T_2 + T_7)}{2} \right) + c.M_2 + d.V_6$$

- (7.80)

SECOND EFFECT (SE)

$$U_{SE} = a + b. \left(T_{10} - \frac{(T_{15} + T_{14})}{2} \right) + c.M_{15} + d.V_4$$

- (7.81)

CONDENSER (CD)

$$U_{CD} = a + b.(T_{11} - \frac{(T_{12}+T_{13})}{2}) + c.M_{12} + d. \\ (V_{10}+V_9) \quad \quad \quad - (7.82)$$

In the above equations a,b,c and d are constants which were determined from the results given in Appendices H-1 to H-4 by multivariable regression analysis. The theory of multivariable regression analysis is given by Davies (7.10). The results of the analysis, which was carried out using the MULREG computer program (7.1), are given in Table 7.5. Appendices H-5 to H-8 show the predicted values calculated using these correlations together with the percentage deviations from the experimental values. As can be seen from these results the deviations from the expected values are all within the range $\pm 4\%$ and this was felt by the author to represent a sufficient degree of accuracy for these correlations to be used in future experimentation with the double effect evaporator.

7.3.5.2. HEAT LOSSES FROM THE EVAPORATOR

As described in section 7.3.3. the results

TABLE 7.5 - RESULTS OF THE MULTIVARIABLE REGRESSION
ANALYSIS FOR OVERALL HEAT TRANSFER
COEFFICIENT CORRELATIONS

	a	b	c	d	MULTIPLE CORRELATION COEFFICIENT
PREHEATER	$7.21546 \cdot 10^{-2}$	$-1.68154 \cdot 10^{-4}$	8.23371	13.6061	0.98714
FIRST EFFECT	2.32106	$-9.52303 \cdot 10^{-2}$	7.11163	152.481	0.99627
SECOND EFFECT	1.29675	$4.34556 \cdot 10^{-2}$	0.814137	-92.922	0.98581
CONDENSER	1.01167	$-5.08808 \cdot 10^{-3}$	-1.64215	$-2.92726 \cdot 10^{-4}$	0.99391

of the steady state experiments include information on the heat losses from the evaporator system. Due to insufficient measurements of the relevant variables within the system being available, it is only possible to compute the heat losses from the first and second effects of the evaporator and not, as would be more desirable for each heat exchange unit. These two heat losses are then lumped together and expressed as a percentage heat loss of the total heat input to the evaporator. The results obtained are shown in Table 7.6 and as can be seen they show no obvious trends. Attempts to correlate the heat loss with the main process flows and temperatures proved unsuccessful. Since the average percentage heat loss is found to be 9.57% and due to the fact that it can not be defined either empirically or theoretically, errors will arise in any dynamic simulation of the evaporator system. However, as will be discussed in section 8.3.1 other means have been devised to minimise the effects of such errors.

7.3.5.3. MEASUREMENT STATISTICS

Section 7.2 described the calibration of

TABLE 7.6 - STEADY STATE HEAT LOSSES FROM THE
DOUBLE EFFECT EVAPORATOR

RUN NO	LOSS FROM 1st EFFECT (kJ/s)	LOSS FROM 2nd EFFECT (kJ/s)	TOTAL LOSS (%)
1	2.72	1.64	8.86
2	9.26	2.31	14.30
3	3.55	1.72	9.44
4	11.89	3.94	7.61
5	4.03	1.43	9.77
6	3.61	1.48	8.28
7	2.13	1.26	6.04
8	9.38	2.58	12.02
9	8.98	2.52	11.49
10	3.71	1.01	8.58
11	8.94	2.88	10.03
12	1.87	1.09	6.04
13	5.95	2.47	7.54
14	5.86	3.14	10.20
15	5.53	2.91	9.66
16	3.01	2.75	6.07
17	3.27	2.64	5.55
18	12.18	5.03	15.09
19	3.60	2.17	6.26
20	12.81	4.61	15.67
21	12.58	4.73	15.82
22	3.37	1.47	6.03
23	8.78	3.88	10.11
24	10.11	4.32	11.81
25	12.94	4.95	16.19
26	4.85	2.04	7.38

TABLE 7.6 continued

27	8.74	3.18	10.16
28	3.02	0.92	5.17
29	11.52	3.92	11.45
30	8.78	2.85	10.09
31	6.32	3.71	7.42
32	2.35	0.41	3.44
33	11.70	4.11	12.18
34	8.34	3.67	11.36
35	5.81	2.19	7.82

AVERAGE TOTAL HEAT LOSS = 9.57%

the various measuring instruments installed at the double effect evaporator. Although these calibrations go some way to accounting for the effects of noise both from external sources and from the instruments themselves, they do not include any form of compensation for noise due to the evaporation process itself. Thus, any measurement made includes an element of uncertainty which has to be accounted for.

To overcome this problem the results from the steady state experiments were used to provide the necessary statistics. The mean of the signals recorded during each experiment is taken as the true value of that process variable, and the noise, or error, to be represented by the calculated standard deviation from this value. These results were then averaged over all of the steady state experiments and used to produce a standard deviation for each of the measured variables. These results are shown in Table 7.7 and will be used later in the on-line filtering experiments.

7.4. CHAPTER REVIEW

The methods used in carrying out this steady

TABLE 7.7 - WEIGHTED STANDARD DEVIATIONS OF DOUBLE
EFFECT EVAPORATOR INSTRUMENTATION

CHANNEL* NO	MEAN STANDARD DEVIATION
0	0.1593
1	0.0988
2	0.1127
3	0.1768
4	0.0884
5	0.1624
6	0.2096
7	0.1531
8	0.2051
9	0.1251
10	$0.6056 \cdot 10^{-3}$
11	0.1087
12	0.6404
13	0.4646
14	1.9203
15	0.6658
16	0.0193
17	$0.931 \cdot 10^{-4}$

*For a full description of all of the evaporator instrumentation see Table 5.2 and Figure 5.4.

state analysis have been described and the results obtained presented and discussed. The experimental results have been used both to generate equations describing the overall heat transfer coefficients and to calculate the statistical uncertainty associated with each of the measuring instruments. An attempt to quantify the heat losses that exist within the evaporator system proved to be inconclusive.

CHAPTER 8

DYNAMIC ANALYSIS OF THE DOUBLE EFFECT EVAPORATOR

8.1. INTRODUCTION

The objective of this dynamic analysis is to produce a mathematical model which will describe the transient and steady state behaviour exhibited by the double effect evaporator. The main constraint on this modelling exercise is that the model developed must be suitable for inclusion within all of the forms of the Kalman Filter which are to be used in on-line experiments. This constraint highlights the following factors which need to be taken into account during model development:-

(i) The model must consist of a set of ordinary differential equations in order to be compatible with the Kalman Filter. Such an approach is a direct contrast to the majority of research currently being carried out into the dynamics of heat exchange equipment where distributed-parameter models are preferred. Simplification of these more exact models is achieved by space lumping the equations into a number of well-mixed regions.

(ii) The model developed must be readily implemented on a digital computer and be of an

order that makes a solution in real-time a feasible objective. Due to the size and complexity of the double effect evaporator a comprehensive mathematical model describing its dynamic behaviour will inevitably involve too many equations to be solved in real-time. Thus, it will be necessary to reduce the model to a more convenient order; this is a process that was found to be necessary by previous workers in this field, see for example Payne (8.1), Coleby (8.2) and Hamilton et al. (8.3).

(iii). The model should take into account the measurements that are available from the double effect evaporator. More explicitly, this statement should be taken to mean that all of the process variables included in the model should either be directly available as measurements, be calculated during the solution of the model, or both.

8.2. DEVELOPMENT OF A DYNAMIC MODEL OF THE DOUBLE EFFECT EVAPORATOR

The approach adopted in the development of this dynamic model was to derive lumped parameter equations for the general heat exchanger in which vapour condenses on the outside of tubes through which liquid is flowing without a change in phase. These equations are then applied, where possible, to the heat exchange units of the double effect evaporator in order to produce a comprehensive mathematical model. Simplifications are then made to produce a model which can be implemented in real time during the on-line filtering experiments.

8.2.1. ASSUMPTIONS

In order to derive the dynamic model of the evaporator as outlined above, it is necessary to make the following assumptions:

(i) The heat exchanger shells are well mixed regions such that the exit stream and shell temperatures are equal.

(ii) The temperature of the tube contents is taken to be that of the liquid stream leaving the tubes.

(iii) Where vapour and liquid phases exist together, the temperature of the mixture is taken to be that of the saturated vapour at the operating pressure.

(iv) The heat given up by the shell-side to the tube-side contents is considered to be by condensation of the vapour alone.

(v) The shell volumes are constant, i.e. there is no liquid hold up.

(vi) Since the cooling water flowing through the tubes of the condenser is obtained from the departmental supply it is assumed that this flow-rate is constant.

(vii) The liquid flowing through the tubes of the second effect calandria is circulated by a powerful pump and so this flowrate is also assumed to be constant.

(viii) The vapour pressure in the condenser shell is considered to be constant.

(ix) The steam pressure on the shell-side of the first effect is also considered to be constant.

(x) Boiling in the second effect separator is isothermal.

(xi) Separation in the cyclone separator is isothermal and there is no liquid hold-up at this point in the system.

(xii) The heat transfer coefficient correlations discussed in chapter 7 also apply to the dynamic state.

In addition to the above assumptions, the thermodynamic correlations discussed in section 7.3.2 are considered to be applicable to the dynamic state.

Two of the above assumptions were arrived at from experimental observations made during the operation of the double effect evaporator. Assumption (ix), i.e. constant steam pressure in the first effect shell, was decided upon after a study of how the steam pressure varied when step changes to the feed flowrate or the steam flowrate were made. Appendix J-1 shows two examples of the pressure variation when such step changes are made. As can be seen from Figure J-1.1 when the steam flowrates is changed the pressure does increase but because this change only lasts for around 20 seconds and since it has an amplitude of approximately 2 kPa, it can be considered to be insignificant. For the case of a change in feed flowrate the effects on the steam pressure have to be transmitted thermally from the tubeside contents and so one would expect the pressure change to be smaller. As can be seen from Figure J-1.2 this turns out to be so.

The other assumption which resulted from experimental observations was the one concerning the pressure in the condenser shell, i.e. assumption (viii). Since all step changes are introduced into the system at the first effect or preheater then the condenser is not only farthest removed from them but also nearest to the very powerful vacuum pump. These two factors suggest that pressure variations are unlikely to be of any significance in the condenser shell and this assumption is confirmed by Figures J-2.1 and J-2.2. As there is no pressure transducer installed at this point in the system, the dynamic responses shown are of the shell outlet temperature, T_{11} , which is of course directly related to the pressure in the condenser shell.

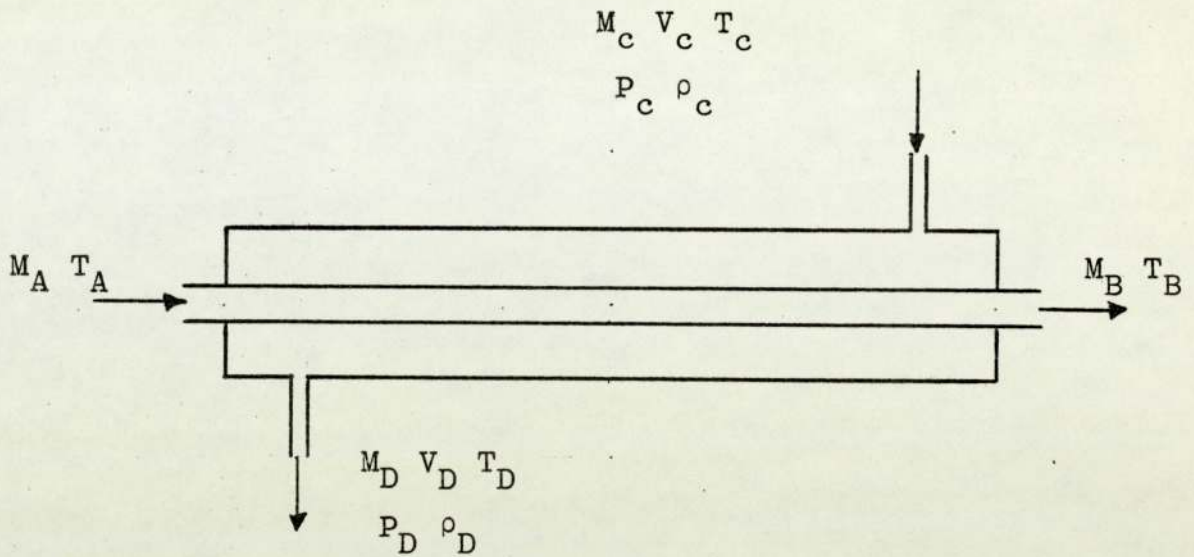
8.2.2. GENERAL APPROACH

Figure 8.1 shows a schematic diagram of the general heat exchanger used to develop general equations applicable to the dynamic state.

TUBESIDE

For a liquid flowing through the tubes without

THE GENERAL HEAT EXCHANGER



LIQUID FLOWRATE (KG.S ⁻¹)	- M _n
VAPOUR FLOWRATE (KG.S ⁻¹)	- V _n
TEMPERATURE (°C)	- T _n
VAPOUR DENSITY (KG.M ⁻³)	- ρ _n
LIQUID DENSITY (KG.M ⁻³)	- ρ _L
OVERALL HEAT TRANSFER COEFFICIENT (KW.M ⁻² .°C ⁻¹)	- U
HEAT TRANSFER AREA (M ²)	- A
SPECIFIC HEAT OF TUBES (KJ.KG ⁻¹ .°C ⁻¹)	- C _t
SPECIFIC HEAT OF SHELL (KJ.KG ⁻¹ .°C ⁻¹)	- C _s
MASS OF TUBES (KG)	- M _t
MASS OF SHELL (KG)	- M _s
VOLUME OF TUBES (M ³)	- V _t
VOLUME OF SHELL (M ³)	- V _s

FIGURE 8.1

a change of phase, the process can be modelled by the following mass and energy balances,

$$\underline{\text{MASS IN}} = \underline{\text{MASS OUT}} \quad - (8.1)$$

$$\underline{\text{HEAT IN}} - \underline{\text{HEAT OUT}} = \underline{\text{ACCUMULATION}} \quad - (8.2)$$

where, HEAT IN = HEAT CONTENT OF INPUT STREAM
+ HEAT TRANSFERRED FROM THE
SHELLSIDE

HEAT OUT = HEAT CONTENT OF OUTPUT STREAM

ACCUMULATION = CHANGE IN HEAT CONTENT OF
TUBES

+ CHANGE IN HEAT CONTENT OF THE
LIQUID IN THE TUBES.

Thus from equation 8.1,

$$\underline{M_A} = \underline{M_B} \quad - (8.3)$$

and from equation 8.2 and assumptions (ii) and (xii),

$$\frac{dT_B}{dt} (V_t \cdot P_L \cdot C_p + W_t \cdot C_t) = M_A \cdot C_p \cdot (T_A - T_B) + U \cdot A \cdot (T_D - \frac{T_A + T_B}{2}) \quad - (8.4)$$

SHELLSIDE

As shown by Figure 8.1, vapour and liquid enter the shell of the general heat exchanger where some of the vapour condenses on the outside of the tubes.

The latent heat released by this condensation is then transferred to the contents of the tubes. Finally, the resultant vapour and liquid mixture leaves the shell without any hold up. Quantitatively, this process can be described by the following mass and energy balances, which have been constructed in a similar way to that reported by Niemi and Koistinen (8.4).

$$\underline{\text{MASS IN} - \text{MASS OUT} = \text{ACCUMULATION}} \quad - (8.5)$$

where, ACCUMULATION = CHANGE IN MASS CONTENT OF SHELLSIDE. Since the shell volume and liquid density have been assumed to be constant, the accumulation of vapour is linked to changes in density. Thus,

$$\text{ACCUMULATION} = V_s \cdot \frac{d\rho_D}{dt} \quad - (8.6)$$

$$\underline{\text{HEAT IN} - \text{HEAT OUT} = \text{ACCUMULATION}} \quad - (8.7)$$

where, HEAT IN = HEAT CONTENT OF INPUT STREAM,

HEAT OUT = HEAT CONTENT OF OUTPUT STREAM

+ HEAT TRANSFERRED TO THE TUBESIDE

ACCUMULATION = CHANGE OF HEAT CONTENT OF SHELL
 + CHANGE OF HEAT CONTENT OF VAPOUR
 DUE TO TEMPERATURE VARIATIONS
 + CHANGE OF HEAT CONTENT OF VAPOUR
 DUE TO DENSITY VARIATIONS.

Thus, from equations 8.5, 8.6 and assumption (v),

$$\frac{d\rho_D}{dt} \cdot V_S = M_C + V_C - M_D - V_D \quad - (8.8)$$

Since vapour density is a function of pressure and temperature and because the vapour is assumed to be saturated,

$$\rho = f(T,P) = g(P) \quad - (8.9)$$

Over the normal operating range of the double effect evaporator, the above equation can be approximately represented by a linear relationship of the form,

$$\rho = b \cdot P + C \quad - (8.10)$$

where, b and c are constants.

Differentiation of the above equation gives,

$$\frac{d\rho}{dt} = b \cdot \frac{dP}{dt} \quad - (8.11)$$

Combining equations 8.11 and 8.8 the shellside mass balance becomes,

$$\frac{dP_D}{dt} \cdot V_S \cdot b = M_C + V_C - M_D - V_D \quad - (8.12)$$

From equation 8.7 and using assumptions (i), (iii), (iv), (v), (xii) the shellside energy balance can be written down as follows,

$$\frac{dT_D}{dt} (W_s \cdot C_s + V_s \cdot \rho_D \cdot C_{pv}) + \frac{dp_D}{dt} \cdot V_s \cdot (C_p \cdot T_D + \lambda(T_D)) = M_c \cdot C_p \cdot T_c + V_c \cdot$$

$$(T_c \cdot C_p + \lambda(T_c)) - M_D \cdot C_p \cdot T_B - V_D (T_D \cdot C_p + \lambda(T_D)) - U.A.$$

$$(T_D - \frac{T_A + T_B}{2})$$

- (8.13)

Combining equations 8.8 and 8.13, the shellside energy balance becomes,

$$\frac{dT_D}{dt} (W_s \cdot C_s + V_s \cdot \rho_D \cdot C_{pv}) = M_c \cdot C_p (T_c - T_D)$$

$$+ V_c \cdot C_p (T_c - T_D) + V_c (\lambda(T_c) - \lambda(T_D))$$

$$+ \lambda(T_D)(M_D - M_c) - U.A. (T_D - \frac{T_A + T_B}{2})$$

- (8.14)

Thus, for the purpose of this modelling exercise, the dynamic behaviour of the general heat exchanger shown in Figure 8.1 will be described by equations 8.3, 8.4, 8.12 and 8.14.

8.2.3. THE COMPREHENSIVE MODEL

Development of a comprehensive model of the

double effect evaporator can now proceed by applying, where possible, the equations obtained in section 8.2.2 to each of the heat exchange units.

8.2.3.1. THE PREHEATER (PH)

At the preheater liquid flows through the tubes without a change in phase and a vapour stream enters the shell, is partially condensed and leaves as a two phase stream. Thus, equations 8.3, 8.4, 8.12 and 8.14 can be applied once all terms involving M_C have been removed since this variable will always be zero. Adopting the process notation used in section 5.2.3 and Figure 8.1 the following mass and energy balances are obtained.

Tubeside Mass Balance,

$$M_1 = M_2 \quad - (8.15)$$

Tubeside Energy Balance,

$$\frac{dT_2}{dt} (V_{PHt} \cdot \rho_L \cdot C_p + W_{PHt} \cdot C_t) = M_1 \cdot C_p \cdot$$

$$(T_1 - T_2) + U_{PH} \cdot A_{PH} \left(T_4 - \frac{T_2 + T_1}{2} \right)$$

- (8.16)

Shellside Mass Balance,

$$\frac{dP_4}{dt} \cdot b \cdot V_{PHS} = V_3 - M_4 - V_4 \quad - (8.17)$$

Shellside Energy Balance,

$$\frac{dT_4}{dt} (W_{PHS} \cdot C_s + V_{PHS} \cdot \rho_4 \cdot C_V) = V_3 \cdot C_p \cdot$$

$$(T_3 - T_4) + V_3 (\lambda(T_3) - \lambda(T_4)) + M_4 \cdot \lambda(T_4)$$

$$+ U_{PH} \cdot A_{PH} (T_4 - \frac{T_2 - T_1}{2})$$

- (8.18)

8.2.3.2. THE FIRST EFFECT (FE)

In the tubes of the climbing film type first effect, the liquid is partially vapourised by the condensation of the shellside steam. This process is assumed to take place in the following three stages. Firstly, the liquid entering the tubes is heated to its boiling point. There then follows a region of nucleate boiling and slug flow. Finally comes the region where vapour flows up the centre of the tube dragging a

liquid film up the tube walls. Thus, for most of the tube there exists two phases which are mixed to varying degrees. After a consideration of the physical processes described above, it was decided to assume that on average the first effect tube contains two well mixed phases. Clearly this is an approximation to the actual situation but it was felt that the increased accuracy which would be obtained if more complex descriptions were used would be outweighed by the computational disadvantages which would accompany the resulting increase in model order. Having made the assumption that the tubeside is well mixed the general approach for shellside processes was applied by reversing the sign of the heat transferred term. Thus, the following mass and energy balances were obtained.

Tubeside Mass Balance,

$$\frac{dP_7}{dt} \cdot b \cdot V_{FEt} = M_2 - M_7 - V_7 \quad - (8.19)$$

Tubeside Energy Balance,

$$\frac{dT_7}{dt} (W_{FEt} \cdot C_t + V_{FEt} \cdot \rho_7 \cdot C_v) = M_2 \cdot C_p \cdot (T_2 - T_7) + \lambda (T_7)(M_7 - M_2) + U_{FE} A_{FE} (T_5 - \frac{T_7 + T_2}{2}) \quad - (8.20)$$

Applying assumptions (i), (iii), (iv), (v) and (ix) to the shellside of the first effect the following process description is obtained; steam enters the first effect shell at a constant pressure, condenses isothermally and then leaves the shell without hold up. Thus, the dynamics of the first effect shell can be described by the following equation,

$$T_5 = \text{constant} \quad - (8.21)$$

It should be noted, however, that the overall heat transfer coefficient, U_{FE} , is a function of steam flowrate and temperature and so the measurement of steam flowrate is not redundant.

8.2.3.3. THE CYCLONE SEPARATOR

As it has been assumed that the cyclone separator operates isothermally and also that there is no liquid hold-up at this point in the system (assumption (xi)) then the following algebraic equations can be written down.

$$T_7 = T_3 = T_8 \quad - (8.22)$$

$$M_7 = M_8 \quad - (8.23)$$

$$V_7 = V_3 \quad - (8.24)$$

8.2.3.4. THE SECOND EFFECT CALANDRIA (SE)

The process taking place at the second effect calandria is the same as that described in section 8.2.2 and so the following mass and energy balances are easily obtained by substituting the correct process variables into equations, 8.3, 8.4, 8.12, and 8.14.

Tubeside Mass Balance,

$$M_{14} = M_{15} \quad - (8.25)$$

Tubeside Energy Balance,

$$\begin{aligned} \frac{dT_{14}}{dt} (V_{SEt} \cdot \rho_L \cdot C_p + W_{SEt} \cdot C_t) = M_{14} \cdot C_p \cdot \\ (T_{15} - T_{14}) + U_{SE} \cdot A_{SE} \left(T_{10} - \frac{T_{14} + T_{15}}{2} \right) \end{aligned} \quad - (8.26)$$

Shellside Mass Balance,

$$\frac{dP_{10}}{dt} \cdot b \cdot V_{SEs} = M_4 + V_4 - M_{10} - V_{10} \quad - (8.27)$$

Shellside Energy Balance,

$$\begin{aligned} \frac{dT_{10}}{dt} (W_{SEs} \cdot C_s + V_{SEs} \cdot \rho_{10} \cdot C_v) = M_4 \cdot C_p \cdot \\ (T_4 - T_{10}) + V_4 \cdot C_p (T_4 - T_{10}) + V_4 (\lambda(T_4) - \lambda(T_{10})) \\ + \lambda(T_{10})(M_{10} - M_4) - U_{SE} A_{SE} \left(T_{10} - \frac{T_{14} + T_{15}}{2} \right) \end{aligned} \quad - (8.28)$$

8.2.3.5. THE SECOND EFFECT SEPARATOR (SP)

The second effect separator was modelled as a well stirred tank operating at the temperature of the liquid outlet stream (T_{15}). Hot liquid enters the second effect separator from the tubes of the second effect calandria and from the base of the cyclone separator. The liquid from the second effect calandria is in fact super heated and so on entry a certain proportion of this liquid flashes off instantaneously. As a result of this flash boiling the liquid level in the separator is nearly always changing.

Taking account of assumption (vii), the unsteady state mass balance for this unit can be written down as,

$$\frac{dH}{dt} \cdot A_{SP} \cdot \rho_L - M_8 - V_9 \quad - (8.29)$$

where, A_{SP} is the cross-sectional area of the second effect separator, and, H is the height of liquid in this unit. The unsteady state energy balance for the second effect separator is,

$$\begin{aligned} \frac{d}{dt} (H \cdot T_{15}) \cdot A_{SP} \cdot \rho_L \cdot C_p &= M_{14} \cdot C_p (T_{14} - T_{15}) \\ + M_8 \cdot C_p \cdot T_8 - V_9 (C_p \cdot T_9 + \lambda(T_9)) &- (8.30) \end{aligned}$$

Expanding the left hand side of the above equation, combining the resulting equation with 8.28 and taking note of assumption (x), the energy balance for the second effect separator becomes,

$$\begin{aligned} \frac{dT_{15}}{dt} A_{SP} \cdot \rho_L \cdot C_p \cdot H &= M_{14} \cdot C_p \cdot (T_{14} - T_{15}) \\ &+ M_8 \cdot C_p (T_8 - T_{15}) - V_9 \cdot \lambda (T_{15}) \end{aligned} \quad - (8.31)$$

8.2.3.6. THE CONDENSER (CD)

Since there is no change of phase in the condenser tubes equations 8.3 and 8.4 were used to describe the tubeside dynamics as follows,

Tubeside Mass Balance,

$$M_{13} = M_{12} \quad - (8.32)$$

Tubeside Energy Balance,

$$\begin{aligned} \frac{dT_{13}}{dt} (V_{CDt} \cdot \rho_L \cdot C_p + W_{CDt} \cdot C_t) &= M_{12} \cdot C_p \cdot \\ (T_{12} - T_{13}) + U_{CD} \cdot A_{CD} (T_{11} - \frac{T_{12} + T_{13}}{2}) \end{aligned} \quad - (8.33)$$

In the condenser shell, vapour from both the second effect separator and the second effect

calandria shell is condensed and the resulting liquid, together with the liquid from the second effect calandria shell, is drawn off without hold-up by the condensate pump. The vacuum pump connected to the condenser shell maintains a constant pressure at this point in the system (assumption (viii)) and so the temperature of the outlet stream also remains constant (see Appendix J-2 and Table 7.4). Thus, as the flow of vapour into the condenser shell varies, the change in the amount of heat to be removed is reflected by the variation in the tubeside outlet temperature (T_{13}). Inherent in this last statement is the assumption that since the temperature of the outlet stream is constant the condenser shell is also maintained at a constant temperature. The condenser shell algebraic mass and energy balances can now be written down as follows,

Shellside Mass Balance,

$$M_{11} = V_9 + M_{10} + V_{10} \quad - (8.34)$$

Shellside Energy Balance,

$$M_{11} \cdot T_{11} \cdot C_p = V_9 (C_p \cdot T_9 + \lambda(T_9)) + V_{10} (C_p \cdot T_{10} + \lambda(T_{10})) + M_{10} \cdot C_p \cdot T_{10} - U_{CD} \cdot A_{CD} (T_{11} - \frac{T_{13} + T_{12}}{2}) \quad - (8.35)$$

8.2.4. THE REDUCED MODEL

The comprehensive model developed in the previous section consists of eleven differential equations and nine algebraic equations. In this form the model can not be readily used in on-line filtering experiments for the following reasons:

(i) The order of the model (eleventh) is too large for a reasonable filter cycle time to be achieved. This is a fairly crucial consideration because of the time constants associated with the heat exchange units of the double effect evaporator.

(ii) The following process variables are neither calculated nor measured - V_3 , V_4 , V_9 , M_4 and M_{10} . In addition, the measuring instrument used to monitor the flowrate of liquid from the base of the cyclone separator (M_8) is rather slow in responding to variations in flowrate. The reasons for this measurement delay are two fold. Firstly the mechanical construction of this instrument does not lend itself to the rapid changes in flowrate which occur. Secondly a steam trap is installed in the pipeline between the cyclone and the flowmeter and this is bound to cause some delay.

As a result of the second of these reasons six relationships need to be developed so that the process variables V_3 , V_4 , V_9 , M_4 , M_8 and M_{10} can be calculated. These relationships were obtained by incorporating the simplifications described below into the comprehensive model.

(i) Previous research in the field of heat transfer dynamics (see for example Payne (8.1), Coleby (8.2), Andre (8.5) and Zavoroka et al. (8.6)), has shown that the time constants associated with the vapour phase dynamics are of a small enough magnitude for flowrates associated with the vapour phase to be approximated by algebraic equations with little loss in accuracy. Figure J-3.1 shows that the pressures associated with the shellside vapour spaces change rapidly for the first twenty seconds after a step change to the steam flowrate but thereafter the variation is much smaller and over a far greater time period. For a step change to the feed flowrate, Figure J-3.2 shows that there is very little change in the pressures associated with the shellside vapour spaces. Thus, the experimental evidence suggests that algebraic equations would be satisfactory as long as some term to

account for the change in pressure is included. The strain gauge pressure transducers used to measure the relevant pressures are shown by the standard deviations given in Table 7.7 (channels 12, 13 and 15) to be accurate instruments and so the following simplification was incorporated into the model,

$$\frac{\Delta P}{\Delta t} = \frac{dP}{dt} \quad - (8.36)$$

where, $\Delta P = P(t_{n+1}) - P(t_n)$,

and, $\Delta t = t_{n+1} - t_n$

As a result of this simplification, equations 8.17, 8.19 and 8.27 are modified.

(ii) The second simplification was made after an examination of Figures J-3.1 and J-3.2 which show that the pressure associated with the second effect separator is fairly constant when step changes are made to either the steam flowrate or the feed flowrate. Thus, it is reasonable to assume that the boiling point of the second effect separator is controlled at T_{15} . Therefore equation 8.31 becomes,

$$\begin{aligned} M_{14} \cdot C_p \cdot (T_{14} - T_{15}) + M_8 \cdot C_p (T_8 - T_{15}) - V_9 \cdot \lambda (T_{15}) \\ = 0 \end{aligned} \quad - (8.37)$$

(iii) From assumption (iv) the rate of heat lost from the shellside of the preheater, at any instant in time, to the tube contents can be expressed as,

$$M_4 \cdot \lambda (T_4) \quad - (8.38)$$

This is equivalent, discounting heat losses, to the rate of heat gained by the preheater tube contents, i.e.

$$M_1 \cdot C_p (T_2 - T_1) \quad - (8.39)$$

Thus, combining 8.38 and 8.39,

$$M_4 = \frac{M_1 \cdot C_p (T_2 - T_1)}{\lambda (T_4)} \quad - (8.40)$$

Similarly, for the first effect,

$$V_7 = \frac{V_6 \cdot (T_5) - M_2 \cdot C_p \cdot (T_7 - T_2)}{\lambda (T_7)} \quad - (8.41)$$

Finally, for the second effect,

$$M_{10} = \frac{M_{14} \cdot C_p \cdot (T_{14} - T_{15})}{\lambda (T_{10})} + M_4 \quad - (8.42)$$

8.2.5. SUMMARY AND DISCUSSION OF THE REDUCED MODEL

The dynamic model of the double effect evaporator developed above consists of the following thirteen

inputs, eleven algebraic equations and seven differential equations.

Inputs to the Model

$$M_1, M_{12}, M_{14}, V_6, T_1, T_3, T_5, T_{11}, T_{12}, T_{15}, P_4, P_7, P_{10}$$

Algebraic Equations

$$M_2 = M_1 \quad - \quad (8.43)$$

$$V_7 = \frac{V_6 \lambda(T_5) - M_2 \cdot C_p \cdot (T_7 - T_2)}{\lambda(T_7)} \quad - \quad (8.44)$$

$$V_3 = V_7 \quad - \quad (8.45)$$

$$M_7 = M_2 - V_7 - \frac{\Delta P_7}{\Delta t} \cdot b \cdot V_{FEt} \quad - \quad (8.46)$$

$$M_8 = M_7 \quad - \quad (8.47)$$

$$M_4 = \frac{M_1 \cdot C_p \cdot (T_2 - T_1)}{\lambda(T_4)} \quad - \quad (8.48)$$

$$V_4 = V_3 - M_4 - \frac{\Delta P_4}{\Delta t} \cdot b \cdot V_{PHt} \quad - \quad (8.49)$$

$$M_{10} = \frac{M_{14} \cdot C_p \cdot (T_{14} - T_{15})}{\lambda(T_{10})} \quad - \quad (8.50)$$

$$V_{10} = M_4 + V_4 - M_{10} - \frac{\Delta P_{10}}{\Delta t} \cdot b \cdot V_{SEt} \quad - \quad (8.51)$$

$$V_9 = \frac{M_{14} \cdot C_p \cdot (T_{14} - T_{15}) + M_8 \cdot C_p \cdot (T_8 - T_{15})}{\lambda(T_{15})} \quad - \quad (8.52)$$

$$M_{11} = V_9 + M_{10} + V_{10} \quad - (8.53)$$

Differential Equations

$$\begin{aligned} \frac{dT_2}{dt} (V_{PHt} \cdot \rho_L \cdot C_p + W_{PHt} \cdot C_t) &= M_1 \cdot C_p \cdot (T_1 - T_2) \\ + U_{PH} \cdot A_{PH} \cdot (T_4 - \frac{T_2 + T_1}{2}) & \quad - (8.54) \end{aligned}$$

$$\begin{aligned} \frac{dT_4}{dt} (W_{PHs} \cdot \rho_s + V_{PHs} \cdot C_4 \cdot C_v) &= V_3 \cdot C_p \cdot (T_3 - T_4) \\ + V_3 (\lambda(T_3) - \lambda(T_4)) + M_4 \cdot \lambda(T_4) + U_{PH} \cdot A_{PH} \cdot \\ (T_4 - \frac{T_2 + T_1}{2}) & \quad - (8.55) \end{aligned}$$

$$\begin{aligned} \frac{dT_7}{dt} (W_{FEt} \cdot C_t + V_{FEt} \cdot \rho_7 \cdot C_v) &= M_2 \cdot C_p (T_2 - T_7) \\ + \lambda(T_7)(M_7 - M_2) + U_{FE} \cdot A_{FE} (T_5 - \frac{T_7 + T_2}{2}) & \quad - (8.56) \end{aligned}$$

$$\begin{aligned} \frac{dT_{10}}{dt} (W_{SEs} C_s + V_{SEs} \cdot \rho_{10} \cdot C_v) &= M_4 \cdot C_p (T_4 - T_{10}) \\ + V_4 \cdot C_p \cdot (T_4 - T_{10}) + V_4 (\lambda(T_4) - \lambda(T_{10})) & \\ + \lambda(T_{10})(M_{10} - M_4) - U_{SE} \cdot A_{SE} (T_{10} - \frac{T_{14} + T_{15}}{2}) & \quad - (8.57) \end{aligned}$$

$$\frac{dT_{13}}{dt} (V_{CDt} \cdot \rho_L \cdot C_p + W_{CDt} \cdot C_t) = M_{12} \cdot C_p (T_{12} - T_{13})$$

$$+ U_{CD} \cdot A_{CD} (T_{11} - \frac{T_{12} + T_{13}}{2}) \quad - (8.58)$$

$$\frac{dT_{14}}{dt} (V_{SEt} \cdot \rho_L \cdot C_p + W_{SEt} \cdot C_t) = M_{14} \cdot C_p \cdot (T_{15} - T_{14})$$

$$+ U_{SE} \cdot A_{SE} (T_{10} - \frac{T_{14} + T_{15}}{2}) \quad - (8.59)$$

$$\frac{dH}{dt} \cdot A_{SP} \cdot \rho_L = M_8 - V_9 \quad - (8.60)$$

The above mathematical model now satisfies the objectives stated in the introduction to this chapter, namely, it is compatible with the Kalman Filter, it can be readily implemented on a digital computer, is of an order that makes a solution in real time feasible and takes account of the measurements available from the double evaporator. It may be thought at this stage that the process of model reduction has oversimplified the system, but it must be remembered that for the application of the Kalman Filter, the definition of a complex system model should be unnecessary. Huddle and Wismer (8.7), Payne (8.1) and Coleby (8.2) all indicate that unavoidable modelling errors and

simplifications can be considered so long as the development of any reduced model includes those process variables of dominant or special interest.

8.3. SIMULATION OF THE DYNAMIC MODEL

Having completed the development of the dynamic model the accuracy of the predictions made must be tested by comparing them with the measured responses obtained during on-line experiments with the double effect evaporator. The necessary data for this simulation exercise was obtained by using the steady state logging program shown in Appendix F-3 and the experimental procedure described in Section 7.3.4. Following the successful execution of the steady state logging program the dynamic logging program is requested and after it has started a step change is introduced into the evaporator system via either the steam flowrate or the feed flowrate.

8.3.1. DISCUSSION OF THE COMPUTER PROGRAM

The computer program which simulates the dynamic model of the double effect evaporator was written in the Basic language according to the program structure shown in Figure 6.2 so that it can be run with the ASP compiler (see section 6.4). A listing of this program is given in Appendix F-4.

Two features of this program which require some explanation are those concerning the procedure used to account for heat losses and the value used for the parameter 'b' (see section 8.2.2). Since the steady state analysis of the evaporator failed to produce a correlation to account for heat losses it was necessary to include some compensation strategy into the model. At the start of all of the on-line experiments, the double effect evaporator is at a steady state and so all of the dynamic model's differential equations should have zero derivatives. A series of preliminary experiments with the model simulation program showed that this was rarely the case and so it was decided to include in each differential equation a residual term such that at the start of all experiments involving the dynamic model the derivatives are zero. These residuals can justifiably be regarded as heat loss terms in view of the fact that the results shown in Table 7.6 indicate an average heat loss from the evaporator of 10%.

The 'b' parameter discussed in section 8.2.2 was evaluated by using the MULREG computer program (8.8) to perform a linear regression on pressure-density data obtained from steam tables (8.9).

8.4. RESULTS AND DISCUSSION

Using the computer program shown in Appendix F-4 a number of simulation experiments were carried out. The results of two of these experiments are shown in Appendix J. The first (see Appendix J-4) shows the predictions and, where applicable, the measurements obtained when a step change was made to the steam flowrate. The second (see Appendix J-5) shows the responses obtained when a step change is made to the feed flowrate. During all of these experiments, the inputs to the dynamic model are obtained from the results of the on-line dynamic logging experiments.

From the simulation results the following points arise for discussion;

(i) In general the predictions made by the model for tubeside temperatures are better than for shellside temperatures. Figures J-4.1, J-4.6, J-5.1, J-5.5 and J-5.6, show that the predictions follow the measurements quite closely as far as tubeside temperatures are concerned. There are a number of reasons which could be proposed to explain why shellside temperatures are not as well modelled but the most probable one is that

when the evaporator is in a dynamic state the liquid and vapour phases are not in equilibrium and consequently assumptions (i) and (iii) will not be strictly correct.

(ii) A further point which can be made about the shellside temperature predictions (see Figures J-4.2, J-4.3 and J-5.3) is that they change more rapidly and as a result reach steady state earlier than the measurements indicate to be the case. This could be caused by the use of algebraic equations to model the liquid and vapour flowrates. However, since there is considerable evidence that the time constants associated with the shellside liquid and vapour phase dynamics are small enough to be neglected then it could well be that the discrepancies are partially due to the sluggish responses by the thermocouples. In a number of cases, the thermocouples installed at the double effect evaporator are not ideally situated and as a result their distance from the heat exchange units will inevitably cause some measurement delay.

(iii) A comparison of the results obtained from a step change to the steam flowrate with

those obtained from a step change to the feed flowrate show that the changes in the state variables are much greater for the first of these two types of step change. This is entirely due to the limited range of feed flowrates which can be used when operating the evaporator. If the flowrate is too low, almost all of the liquid feed is vapourised at the first effect and if it is too high the cyclone separator fills with liquid because of the steam trap in the line between the cyclone and the second effect separator.

(iv) Figures J-4.7 and J-5.7 show that the predicted and measured heights in the second effect separator are in close agreement. This is a particularly pleasing result as it confirms some of the assumptions made during the derivation of the dynamic model. Inherent in the differential equation used to predict the height in the second effect separator (equation 8.60) is the assumption that there is no mass loss from the system. Although this assumption may not be exactly correct the results obtained do show it to be a reasonable approximation to the truth. The second assumption which is confirmed

by these results is the one concerning the use of algebraic equations to calculate the liquid and vapour flowrates occurring at the evaporator. Clearly the use of algebraic equations does not cause a significant loss in accuracy. The third point which is clarified by these results is the apparent contradiction between the results of the steady state analysis and the assumption of constant flowrate through the second effect calandria tubes. The steady state analysis (see Chapter 7) tends to indicate that this flowrate changes between different steady states due to the variation in height in the second effect separator causing the circulation pump to act against differing heads. However, the results obtained during these dynamic simulation experiments (see Figures J-4.7 and J-5.7) show that this variation in flowrate is not a significant one.

Figures J-4.7 and J-5.7 also show that on average the measurements of height in the second effect separator are approximately correct. However, one can not rely too much on a single measurement because of the noise it may contain;

this noise being caused by the boiling which takes place at the second effect separator.

(v) The discrepancy between the measured and predicted values of T_{13} (see Figure J-4.5) for a step change to the steam flowrate is quite an appreciable one. However, since T_{11} does not change very much the only way in which the derivative of T_{13} can increase significantly is if the heat transfer coefficient U_{CD} , increases. Equation 7.82 shows that the only way in which U_{CD} will change in this situation is if the vapour flowrate into the shell, $V_{10}+V_9$, increases and as the coefficient for the vapour flowrate term, d , is small, the predicted response of T_{13} may initially lag behind the measured value.

(vi) Figure J-5.4 shows that the prediction of the temperature of the exit stream from the second effect shell, T_{10} , is noisier than might be expected. This is caused by the complicated nature of the equation used to predict T_{10} . Since this equation contains a large number of terms which include flowrates calculated from measured inputs to the dynamic model, then if any one of these measurements is noisy the

the derivative could well be incorrect and on integration a noisy prediction will be generated.

(vii) Figures J-4.1, J-5.1 and J-5.2 indicate that some of the predicted temperatures associated with the first effect and the preheater are slow in reaching steady values following the introduction of a disturbance to the system. It is felt that this is caused by not updating the residuals used in the calculation of the derivative terms (see section 8.3.1). Clearly the heat loss from a heat exchange unit will change as the heat transferred varies but there is no way of knowing when the residual (or heat loss) terms should be modified.

(viii) The differential equations included in the dynamic model are solved using Runge-Kutta fourth order numerical integration which provides accurate predictions only if the step length is not too large. To determine the optimal integration interval the values of T_7 obtained after 30 seconds using different step lengths were compared. Table 8.1 shows that if a step length of 1 second is regarded as a standard then integration intervals between 5 and 15 seconds are acceptable.

TABLE 8.1. - COMPARISON OF THE ACCURACIES OF
DIFFERENT STEP LENGTHS

STEP LENGTH (SEC)	VALUE OF T_7 AFTER 30 SECS
1	75.4137
5	75.4561
10	75.5090
15	75.5133
20	75.9500
25	76.6298
30	77.6321

The overall conclusion made about the dynamic model after this simulation study was that although it is bound to be somewhat approximate due to the fact that it consists of ordinary differential equations, it does provide predictions which are sufficiently accurate for on-line filtering experiments.

8.5. CHAPTER REVIEW

A mathematical model describing both the dynamic and steady states of the double effect evaporator has been derived. The model is compatible with the Kalman Filter and can be readily solved on a digital computer in a time which makes on-line applications feasible.

The results of model simulation experiments have been reported and after a comparison of the predictions with measurement data the model was found to be acceptable.

CHAPTER 9

ON-LINE FILTERING EXPERIMENTS

9.1. INTRODUCTION

The culmination of this research project was the on-line application of the Kalman Filter. Every aspect of the work reported so far has been directed towards these on-line filtering experiments and during them the reliability and efficiency of all of the developments made was rigorously tested.

The discussion of the simulation studies of the Kalman Filter, see chapter four, revealed that the following three filters were the most suitable for on-line applications,

TYPE 2 - The Extended Kalman Filter,

TYPE 3 - The Extended Kalman Filter - using a state transition matrix calculated by the canonisation procedure discussed in chapter three.

TYPE 4 - The Adaptive Kalman Filter.

Thus, three on-line filtering programs were constructed using the above forms of the Kalman Filter and the dynamic model of the double effect evaporator discussed in chapter eight. These programs were written in Extended Honeywell Fortran so that they could be executed under the control of the OLDFP Executive, see chapter 6.

9.2. DISCUSSION OF THE ON-LINE FILTERING PROGRAMS

Each on-line filtering program to be run under the control of the OLDFP Executive consists of two master segments, INIT and FILTER.

Initialisation of all of the variables required for the filtering process is carried out by INIT. Following this initialisation FILTER is executed periodically to perform the task of filtering. The two main functions of FILTER are to handle all input/output and also to call on a number of other routines to perform the specialised tasks of prediction - subroutine PREDIC, estimation - subroutines KALMAN and ADAPT and calculation of the state transition matrix - subroutines TRANS(1) and TRANS(2).

Communication between all of the segments of these programs is by three named COMMON Blocks:-

(i) COMMON/KALM/ - all variables and arrays associated with the estimation segments

(ii) COMMON/MODEL/ - all variables and arrays associated with the prediction segment.

(iii) COMMON/SCAN/ - all variables and arrays associated with the data acquisition process. An explanation of all of the variables contained in the above COMMON Blocks is given in Table 9.1.

TABLE 9.1 - COMMON BLOCKS FOR ON-LINE FILTERING

PROGRAMS

COMMON BLOCK NAME	VARIABLE NAME	MEANING
KALM	XE(I)	VECTOR OF ESTIMATED STATE VARIABLES
	XP(I)	VECTOR OF PREDICTED STATE VARIABLES
	Y(I)	VECTOR OF MEASUREMENTS
	PE(I,J)	STATE VARIABLE ERROR COVARIANCE MATRIX
	R(I,J)	MEASUREMENT ERROR COVARIANCE MATRIX
	Q(I,J)	PROCESS NOISE COVARIANCE MATRIX
	THY(I,J)	STATE TRANSITION MATRIX, $\phi(k+1,k)$
	GAMMA(I,J)	INTEGRAL STATE TRANSITION MATRIX, $\Gamma(k+1,k)$
	RK(I,J)	FILTER GAIN MATRIX
	RM(I,J)	MEASUREMENT MATRIX, M i.e. $y=M.x$
	N	DEGREE OF STATE VECTOR
	M	DEGREE OF MEASUREMENT VECTOR
	P	SAMPLING INTERVAL, Δt .
	C(I,J)	STATE VARIABLE AND MODEL ERROR COVARIANCE MATRIX
	GAM(I)	VECTOR OF RESIDUALS, γ
	WW(I)	VECTOR OF MODEL ERRORS, \bar{w}
	F4(I,J)	RELATIONSHIP BETWEEN THE MODEL ERROR VECTOR, \bar{w} , AND THE STATE VECTOR, x , i.e. $\dot{x}(t)=F_1.x(t)+F_4.\bar{w}$.
	ALPHA	EXPONENTIAL FILTER CONSTANT, α
	BETA	EXPONENTIAL FILTER CONSTANT, β
	ITHETA	NUMBER OF FILTER CYCLES BETWEEN UPDATES OF \bar{w}
MF4	NUMBER OF COLUMNS IN MF4	
G	ESTIMATE OF THE TRACE OF RESIDUALS COVARIANCE MATRIX, g	
CC	DIAGONAL ELEMENT OF MODEL ERROR COVARIANCE MATRIX, \bar{c} , i.e. $P_w=I.\bar{c}$	
MODEL	V1toV6	VOLUMES OF EVAPORATOR TUBES AND SHELLS
	W1toW6	MASSES OF EVAPORATOR TUBES AND SHELLS
	CltoC4	SPECIFIC HEATS
	AltoA4	HEAT TRANSFER AREAS
	A5	CROSS SECTIONAL AREA OF SECOND EFFECT SEPARATOR

TABLE 9.1 - continued

	HALtoHA4 HBltoHB4 HC1toHC4 HD1toHD4 T(16) W(16) V(16) H DP1 DP2 DP3 P12 P15 U1toU4 R4, R7, R10	} COEFFICIENTS FOR HEAT TRANSFER CORRELATIONS } DOUBLE EFFECT EVAPORATOR TEMPERATURES } DOUBLE EFFECT EVAPORATOR LIQUID FLOWRATES } DOUBLE EFFECT EVAPORATOR VAPOUR FLOWRATES } HEIGHT IN SECOND EFFECT SEPARATOR } PRESSURE DERIVATIVES, $\frac{\Delta P}{\Delta t}$ } PRESSURE - CHANNEL 12 } PRESSURE - CHANNEL 15 } OVERALL HEAT TRANSFER COEFFICIENTS } VAPOUR DENSITIES
SCAN	INTR STEP NCYCLE NCOUNT IPRST IPRNT C(I) D(I)	SAMPLING INTERVAL INTEGRATION STEP LENGTH NUMBER OF FILTER CYCLES REQUIRED NUMBER OF FILTER CYCLES SO FAR COUNTER PRESET VALUE HIGH SPEED PUNCH PRINT OUT FLAG : 1 to PRINT INSTRUMENT CALIBRATIONS - INTER- CEPTS INSTRUMENT CALIBRATIONS - SLOPES

-ESTIMATED AND MEASURED STATE VARIABLES

ESTIMATE	PREDICTION	MEASUREMENT	VARIABLE NAME	VARIABLE
XE(1)	XP(1)	Y(1)	T ₂	PREHEATER TUBES EXIT TEMPERATURE
XE(2)	XP(2)	Y(2)	T ₄	PREHEATER SHELL EXIT TEMPERATURE

TABLE 9.1 - continued

XE(3)	XP(3)	Y(3)	T ₇	FIRST EFFECT TUBES EXIT TEMPERATURE
XE(4)	XP(4)	-	T ₁₀	SECOND EFFECT SHELL EXIT TEMPERATURE
XE(5)	XP(5)	Y(4)	T ₁₃	CONDENSER TUBES EXIT TEMPERATURE
XE(6)	XP(6)	Y(5)	T ₁₄	SECOND EFFECT TUBES EXIT TEMPERATURE
XE(7)	XP(7)	Y(6)	H	HEIGHT IN SECOND EFFECT SEPARATOR

9.2.1. THE INITIALISATION SEGMENT - INIT

In the initialisation segment the COMMON Blocks described above are set up either by arithmetic assignment statements or by reading in values input at the V.D.U. It will be noticed from the listing of INIT, See Appendix G-1, that many of the simple variables in COMMON/MODEL/ and the arrays in COMMON/SCAN/ are not initialised in this segment. This is because their values do not vary from experiment to experiment and so these variables and arrays are initialised when the program is loaded by the use of a Block Data Subprogram, see Appendix G-10.

Once the initialisation process is complete, a message informing the user that the program is ready to start is printed at the VDU and bit 1 of the digital output display is set. The digital input switches are now repeatedly sensed until bit 2 is set, at which time the filtering program FILTER, is connected to the clock for periodic execution and INIT is terminated.

9.2.2. THE FILTERING SEGMENT - FILTER

As shown by the flowchart in Figure 9.1 and

FLOWCHART OF ON-LINE FILTERING PROGRAM

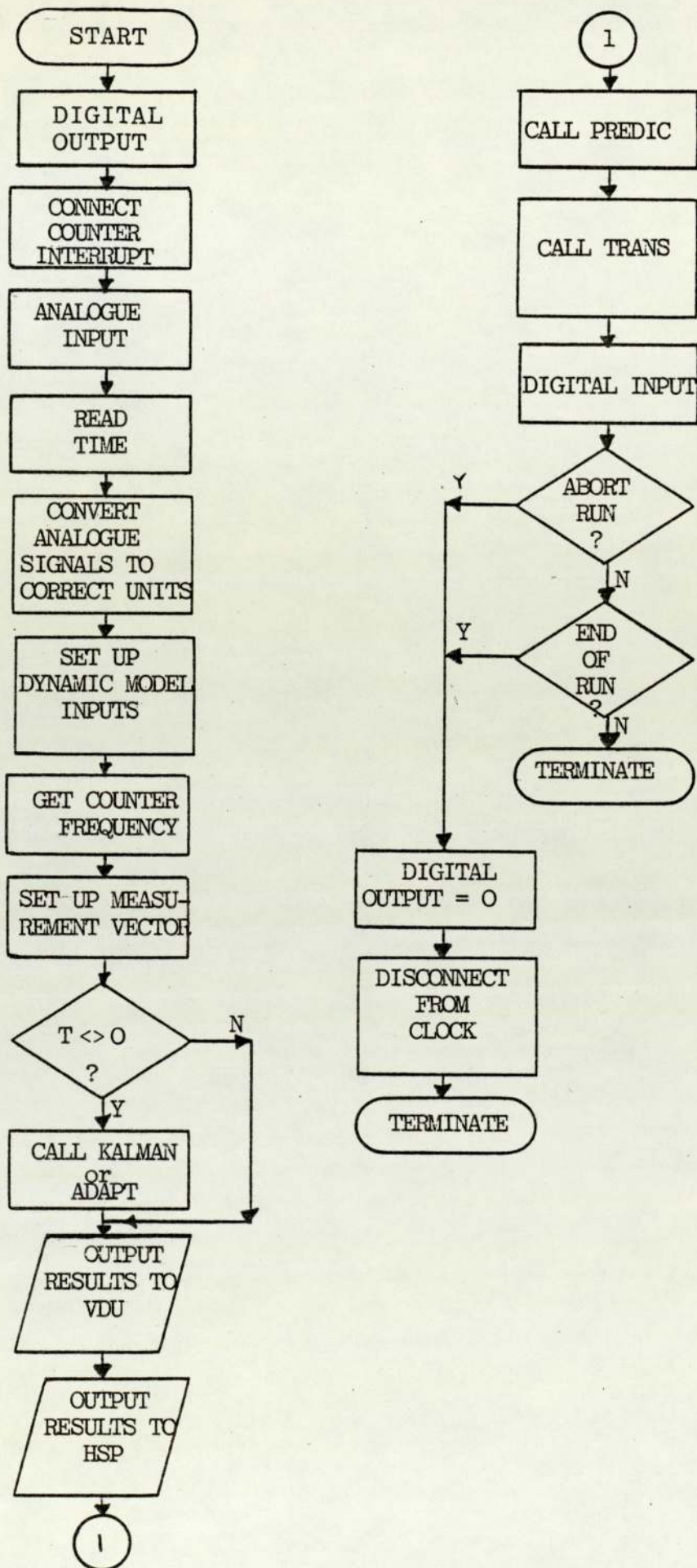


FIGURE 9.1

the listing given in Appendix G-2, the first action on entering FILTER is to signal the start of its execution by outputting the number of scans done to the digital output display. Interrupt is now connected to the counter input subinterface so that the flowrate to the preheater tubes can be obtained. The other instrument signals are then measured via the analogue inputs function, H6, of the HADIOS Supervisory program. The analogue measurements are then converted to the correct units and the inputs to the dynamic model set up. The counter frequency is now evaluated and converted into the correct units. The final step before filtering takes place is to set up the measurement vector, y .

Apart from during the first execution of this program, the next step is to call upon one of the estimation routines, KALMAN or ADAPT, to perform the task of filtering. Following this, the results from the filtering process are printed out at the VDU and then punched out by the high speed punch so that a permanent copy of the results can be obtained. Output to the high

speed punch is not via a WRITE statement because the OP-16 Fortran Package does not support this peripheral device. Instead the punch driver program is requested directly using the REQUEST statement (see line 152 of Table G-2.1). Due to the way in which the output to the punch is obtained the paper tape produced is in binary format and although these tapes have to be converted to Ascii format before they can be used in other programs, the use of binary format is quite an advantage because the output is greatly condensed and so less time is required by this operation.

The remainder of this program is devoted to the calculation of the predicted state from the estimates already obtained and the calculation of the state transition matrix. Finally, before terminating the program, a check is made to see whether the user wishes to abort the experiment. The user can bring about this action by setting bit 1 of the digital input switches.

9.2.3. SUBROUTINE PREDIC

The function of subroutine PREDIC is to solve

the dynamic model of the double effect evaporator discussed in chapter 8 by the fourth order Runge-Kutta method of numerical integration. The listing of this subroutine, see Appendix G-5, shows that the structure of this program is similar to that of the Basic program used for the dynamic model simulation studies, see Appendix F-4. The main differences are that there is no input/output and numerical integration is performed explicitly instead of by calling subroutines.

9.2.4. SUBROUTINE TRANS(1)

Subroutine TRANS(1) calculates the state transition matrix, $\phi(k+1,k)$, using a truncated Taylor series, see section 3.3. This method requires the calculation of the Jacobian of partial derivatives for the mathematical model used and so, as shown in section 4.4.2.4, the state transition matrix is calculated as follows,

$$\phi(k+1,k) = I + \left. \frac{\partial f(x)}{\partial x} \right|_{x(k,k)} \cdot \Delta t \quad - (9.1)$$

Thus, when using this method, $\phi(k+1,k)$ can be

calculated at each cycle of the filter by setting all elements of the matrix THY to zero except for those defined below:

$$\text{THY (1,1)} = (-0.5 \cdot U_{\text{PH}} \cdot A_{\text{PH}} - M_2 \cdot C_p) \cdot \frac{\Delta t}{\text{DIV1}} + 1 \quad - (9.2)$$

$$\text{THY (1,2)} = (U_{\text{PH}} \cdot A_{\text{PH}}) \cdot \frac{\Delta t}{\text{DIV1}} \quad - (9.3)$$

$$\text{THY (2,1)} = (0.5 \cdot U_{\text{PH}} \cdot A_{\text{PH}}) \cdot \frac{\Delta t}{\text{DIV2}} \quad - (9.4)$$

$$\text{THY (2,2)} = (-V_3 \cdot C_p - V_3 \cdot L + M_4 \cdot L - U_{\text{PH}} \cdot A_{\text{PH}}) \cdot \frac{\Delta t}{\text{DIV2}} + 1 \quad - (9.5)$$

$$\text{THY (2,3)} = (V_3 \cdot C_p + V_3 \cdot L) \cdot \frac{\Delta t}{\text{DIV2}} \quad - (9.6)$$

$$\text{THY (3,1)} = (M_2 \cdot C_p + 0.5 \cdot U_{\text{FE}} \cdot A_{\text{FE}}) \cdot \frac{\Delta t}{\text{DIV3}} \quad - (9.7)$$

$$\text{THY (3,3)} = (-M_2 \cdot C_p + L \cdot (M_8 - M_2) - 0.5 \cdot U_{\text{FE}} \cdot A_{\text{FE}}) \cdot \frac{\Delta t}{\text{DIV3}} \quad - (9.8)$$

$$\text{THY (4,2)} = (M_4 \cdot C_p + V_4 \cdot (C_p + L)) \cdot \frac{\Delta t}{\text{DIV4}} \quad - (9.9)$$

$$\text{THY (4,4)} = (-M_4 \cdot C_p - V_4 \cdot (C_p + L) + L \cdot (M_{10} - M_4) - U_{\text{SE}} \cdot A_{\text{SE}}) \cdot \frac{\Delta t}{\text{DIV4}} + 1 \quad - (9.10)$$

$$\text{THY (4,6)} = (0.5 \cdot U_{SE} \cdot A_{SE}) \cdot \frac{\Delta t}{\text{DIV4}} \quad - (9.11)$$

$$\text{THY (5,5)} = (-M_{12} \cdot C_p - 0.5 \cdot U_{CD} \cdot A_{CD}) \cdot \frac{\Delta t}{\text{DIV5}} + 1 \quad - (9.12)$$

$$\text{THY (6,4)} = (U_{SE} \cdot A_{SE}) \cdot \frac{\Delta t}{\text{DIV6}} + 1 \quad - (9.13)$$

$$\text{THY (7,7)} = 1 \quad - (9.14)$$

$$\text{where, DIV1} = V_{PH_t} \cdot \rho_L \cdot C_p + W_{PH_t} \cdot C_t \quad - (9.15)$$

$$\text{DIV2} = V_{PH_s} \cdot \rho_4 \cdot C_v + W_{PH_s} \cdot C_s \quad - (9.16)$$

$$\text{DIV3} = V_{FE_t} \cdot \rho_7 \cdot C_v + W_{FE_t} \cdot C_t \quad - (9.17)$$

$$\text{DIV4} = V_{SE_s} \cdot \rho_{10} \cdot C_v + W_{SE_s} \cdot C_s \quad - (9.18)$$

$$\text{DIV5} = V_{CD_t} \cdot \rho_L \cdot C_p + W_{CD_t} \cdot C_s \quad - (9.19)$$

$$\text{DIV6} = V_{SE_t} \cdot \rho_L \cdot C_p + W_{SE_t} \cdot C_t \quad - (9.20)$$

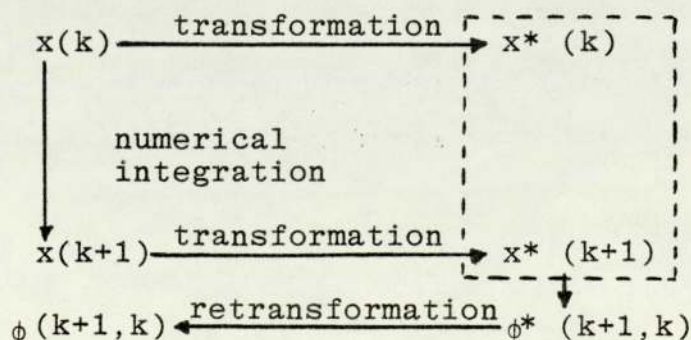
$$\begin{aligned} \text{and, } L &= \frac{d}{dT_i} (\lambda(T_i)) \\ &= \frac{d}{dT_i} (-2.4068 \cdot T_i + 2501.6) \\ &= -2.4068 \quad - (9.21) \end{aligned}$$

9.2.5. SUBROUTINES TRANS(2) AND RUTIS

One of the disadvantages of using the canonisation procedure described in section 3.3.

to calculate the state transition matrix is that this method increases the computational requirements of the Kalman Filter. The discussion of the results obtained during the simulation studies of the Kalman Filter, see chapter 4, pointed out that a considerable saving in the required computational power could be achieved by assuming that the eigenvector matrix required by this procedure was constant. For the on-line filtering experiments the fact that the computer to be used had only 16K words of memory and a cycle time of $1.6\mu\text{s}$ prompted an investigation into the possibility of using the simplified canonisation procedure described in section 4.5.4.4.

The simplified canonisation procedure which is being proposed for the on-line filtering programs can be best explained by reference to the diagram shown below.



SIMPLIFIED CANONISATION PROCEDURE

In the above diagram the process of transformation is carried out by using equation 3.77, calculation of $\phi^*(k+1,k)$ is done using equation 3.79, and retransformation is carried out by using equation 3.81. In all of the equations quoted above the eigenvector and inverse eigenvector matrices $V(k)$ and $V(k)^{-1}$, have constant values when this simplified procedure is used. The use of the above procedure greatly reduces the computational requirements of this form of the Kalman Filter and, as was discussed in section 4.5.4.4, does so without a significant loss in accuracy.

Clearly, before deciding to use the simplified canonisation procedure it is necessary to try and determine eigenvector and inverse eigenvector matrices which can be used throughout the on-line filtering experiments. The investigation to find these matrices was carried out off-line and is described below.

The first requirement of this investigation was to develop a means of calculating the coefficients matrix, $A(k)$, which will be used to determine the eigenvalues and eigenvectors of the process occurring at the double effect

evaporator. This was done by rearranging the differential equations of the dynamic model see equations 8.54 to 8.60, into the form,

$$\dot{x}(k) = A(k) \cdot x(k) \quad - (9.22)$$

Thus, when the matrix A has to be evaluated, it is obtained by setting all of the elements of the matrix to zero except for those defined below,

$$A(1,1) = (-M_2 \cdot C_p - 0.5 \cdot U_{PH} \cdot A_{PH}) \cdot \frac{1}{DIV1} \quad - (9.23)$$

$$A(1,2) = (U_{PH} \cdot A_{PH}) \cdot \frac{1}{DIV1} \quad - (9.24)$$

$$A(2,1) = (0.5 \cdot U_{PH} \cdot A_{PH}) \cdot \frac{1}{DIV2} \quad - (9.25)$$

$$A(2,2) = (-V_3 \cdot (C_p + L) + M_4 \cdot L - U_{PH} \cdot A_{PH}) \cdot \frac{1}{DIV2} \quad - (9.26)$$

$$A(2,3) = (V_3 \cdot (C_p + L)) \cdot \frac{1}{DIV2} \quad - (9.27)$$

$$A(3,1) = (M_2 \cdot C_p + 0.5 \cdot U_{FE} \cdot A_{FE}) \cdot \frac{1}{DIV3} \quad - (9.28)$$

$$A(3,3) = (-M_2 \cdot C_p + L \cdot (M_8 - M_2) - 0.5 \cdot U_{FE} \cdot A_{FE}) \cdot \frac{1}{DIV3} \quad - (9.29)$$

$$A(4,2) = (M_4 \cdot C_p + V_4 \cdot (C_p + L)) \cdot \frac{1}{DIV4} \quad - (9.30)$$

$$A(4,4) = (-M_4 \cdot C_p - V_4 \cdot (C_p + L) + L(M_{10} - M_4) - U_{SE} \cdot A_{SE}) \cdot \frac{1}{DIV4} \quad - (9.31)$$

$$A(4,6) = (0.5 \cdot U_{SE} \cdot A_{SE}) \cdot \frac{1}{DIV4} \quad - (9.32)$$

$$A(5,5) = (-M_{12} \cdot C_p - 0.5 \cdot U_{CD} \cdot A_{CD}) \cdot \frac{1}{DIV5} \quad - (9.33)$$

$$A(6,4) = (U_{SE} \cdot A_{SE}) \cdot \frac{1}{DIV6} \quad - (9.34)$$

$$A(6,6) = (-M_{14} \cdot C_p - 0.5 \cdot U_{SE} \cdot A_{SE}) \cdot \frac{1}{DIV6} \quad - (9.35)$$

$$A(7,7) = 1 \quad - (9.36)$$

where, DIV1 to DIV6 and L are as defined by equations 9.14 to 9.21.

To determine the eigenvalues and eigenvectors of the above system a Basic program was written, see Appendix F-6, to be executed with a package called ASPEIG which is based on the BASIC-16 Interpreter. This package includes two Fortran subroutines, the first (subroutine number 5) to calculate the eigenvalues and eigenvectors of the system and the second (subroutine number 6) to invert a given matrix by the method of Gauss-Jordan.

The program used to determine the eigenvectors and eigenvalues of the coefficients matrix is based on Rutishauser's LR transformation (9.1) and is a modified form of the program first reported by Carnahan et al. (9.2).

The final requirement of this investigation is a source of values for the process variables which are used to calculate the coefficients matrix, $A(k)$, see equations 9.28 to 9.36. These values were obtained by solving the algebraic equations of the dynamic model, equations 8.43 to 8.53, using the measurements obtained during dynamic logging experiments as the inputs to the model equations. During these dynamic logging experiments disturbances were introduced into the system by a step change to either the feed or the steam flowrates.

Since the interval between successive measurements was ten seconds during the dynamic logging experiments the Basic program shown in Appendix F-6 was used to calculate and printout the coefficients matrix, the eigenvalues and the eigenvector and inverse eigenvector matrices every ten seconds for the duration of the experiment (normally ten minutes).

From the results of this investigation the following observations were made,

(i) The eigenvalues of the system changed by less than 5% for a step change to the feed flowrate.

(ii) The eigenvalues of the system changed by approximately 10% for a step change to the steam flowrate.

(iii) The eigenvalues changed most noticeably during the first two minutes following the introduction of a step change.

(iv) The eigenvalues obtained during different runs of the program never differed by more than 30%.

From these observations and since there is no way known to the author whereby the average eigenvector matrix can be calculated from these results, it was decided to choose an eigenvector matrix from a period of response following a step change to the steam flowrate during the most representative run. The most representative run was chosen as the one which on average has eigenvalues in the middle of the range of all of the experiments.

As a result of this investigation subroutine TRANS(2), see Appendix G-7, was written to calculate the state transition matrix at each cycle of the filter using the simplified canonisation procedure described above and subroutine RUTIS was written to set up the constant eigenvector and inverse eigenvector matrices when this is required by subroutine TRANS(2).

9.2.6. CONSTRUCTION OF THE ON-LINE FILTERING PROGRAMS

The remaining subroutines required to construct the on-line filtering programs are the estimation segments, KALMAN and ADAPT, and the matrix manipulation routines, MATINV, MATMUL, MATADD and MATRAP. All of these subroutines are the same as those used in the simulation studies (see chapter 4) and so they will not be described here. The listings of these subroutines are given in Appendices G-3, G-4 and G-9 respectively.

Using the routines shown in Table 9.2 the three on-line filtering programs, FILTER2, FILTER3 and FILTER4 were now constructed. In order to

TABLE 9.2 - COMPONENTS OF ON-LINE FILTERING PROGRAMS

FILTER2	FILTER3	FILTER4
INIT	INIT	INIT
FILTER	FILTER	FILTER
KALMAN	KALMAN	ADAPT
PREDIC	PREDIC	PREDIC
TRANS(1)	TRANS(2)	TRANS(1)
-	RUTIS	-
MATLIB	MATLIB	MATLIB
LINKS*	LINKS*	LINKS*
PROGRAM	PROGRAM	PROGRAM

*See Appendix G-11

make these programs compatible with the OLDFP Executive a special version of the Fortran Libraries tape, see Appendix G-15, was used when loading these programs. The loading procedures for these programs are given in Appendices G-12, G-13 and G-14.

The Executive errors which may be generated by the on-line filtering programs at run time are tabulated in Appendix G-16.

9.3. EXPERIMENTAL PROCEDURE

Before on-line experiments can commence both the sampling interval (or filter cycle time) and the step length used in the prediction routine, subroutine PREDIC, need to be determined. Due to the limited range of step lengths which can be used with subroutine PREDIC, see section 8.4 and Table 8.1, it was decided to use the smallest practical step length, i.e. five seconds, and determine the minimum sampling interval which this requires. If this step length had caused the sampling interval to be too large then the step length could have been increased but this proved unnecessary as the following results show that a satisfactory sampling interval could be achieved when using a step length of five seconds.

Type 2 Sampling Interval = 23 seconds

Type 3 Sampling Interval = 24 seconds

Type 4 Sampling Interval = 29 seconds

No precise breakdown of the above times could be obtained but it was determined that the execution of subroutine PREDIC took approximately 12 seconds. In order to standardise the results obtained a

sampling interval of 30 seconds was used during the on-line experiments and it was felt that this compared favourably with the sampling intervals of 60 seconds and 120 seconds used by Coleby (9.3) and Payne (9.4).

Following the start up of the double effect evaporator as described in section 5.2.2, the procedure used for the on-line filtering experiments was,

(i) Load the HADIOS Executive Package Mk.II and the steady state logging program into the H316 computer.

(ii) Execute the steady state logging program to check that the evaporator is at steady state and to obtain initial values for the filtering programs.

(iii) Load the OLDFP Executive and one of the on-line filtering programs (FILTER2, 3 or 4) into the H316 computer.

(iv) Enter the Utility Program (ONLCUP) by typing a '\$' character at the VDU.

(v) Set the system time using the replace time (RT) function.

(vi) Use the replace core (RC) function to

store the start address of the initialisation program, INIT, in location 3246_8 .

(vii) Request the initialisation program and then terminate the utility program.

(viii) Enter the initial values, statistics and parameters at the VDU as required by INIT.

(ix) Following the printing of the "Ready to Start" message at the VDU, a final check is made to ensure that everything is ready for the experiment to begin and then the filtering program is connected to the RTX-16 Real Time Clock program by setting bit 2 of the digital input switches.

(x) During the first five minutes of the filtering experiment the results output to the VDU are examined carefully to ensure that the filter is operating correctly. The filter is deemed to be operating correctly if the following conditions are met during this initial period.

(a) The estimates are converging towards steady values which are in close agreement with the measurements.

(b) The predictions calculated by the dynamic model display the sort of behaviour which is consistent with the evaporator being at steady state.

(c) The diagonal elements of the error covariance matrix are converging to steady values which are of the order of 1.

Once the filtering programs had been debugged the above checks were only really necessary for the Extended Kalman Filter (Type 2), the other types of filter always displaying good convergence. This is probably the first observation made during the on-line filtering experiments which points to the instability of the classical Kalman Filter when it is applied to the estimation of chemical processes.

If the filter is operating correctly the system is then disturbed by making a step change to the steam flowrate. During certain on-line experiments it was decided at this time that the filter was not operating correctly and so the run was aborted by setting bit 1 of the digital input switches.

(xi) After the experiment had been proceeding

for 15 to 16 minutes a further step change to the steam flowrate was made.

(xii) Once the experiment had finished the evaporator was left for forty minutes so that it could again reach steady state and then steps (i) to (xi) were repeated for as many experiments as were required.

When a sufficient number of experiments had been completed the binary format tapes containing the filtering results were converted to Ascii format tapes using the Fortran program shown in Appendix G-17. The results were then plotted at the VDU using the graph plotting program shown in Appendix F-5 and, when required, permanent copies of these graphs were obtained using the Hard Copy Unit (for examples of the graphs obtained see Appendix K).

9.3.1. FILTER TUNING EXPERIMENTS

Before the experiments described above could be carried out it was necessary to optimise the performance of each filter. These filter tuning

experiments were carried out using the procedure outlined in the previous section with the exception that no step changes were introduced into the system. Unlike similar experiments carried out under simulated conditions, see chapter 4, there is no way of determining the errors contained in the state estimates generated during on-line experiments and so a more qualitative approach had to be adopted. Ideally, the estimates obtained should initially rely on the noisy measurements so that they converge on the true state of the plant and thereafter they should display the influence of both the predictions and the measurements such that the errors introduced by process noise and measurement inaccuracies are eliminated without divergence or bias being caused. Fortunately, only the process noise covariance matrix (the Q matrix) needs to be determined during these filter tuning experiments as the measurement noise covariance matrix has already been determined from the steady state analysis of the evaporator, see chapter 7.

From the results obtained during these experiments, see Table 9.3, the following points were noted,

TABLE 9.3 - RESULTS OF ON-LINE FILTER TUNING
EXPERIMENTS

VARIABLE	VALUE
P(0,0)	OFF DIAGONAL ELEMENTS=0 DIAGONAL ELEMENTS=5
R(k)	OFF DIAGONAL ELEMENTS=0 DIAGONAL ELEMENTS= 0.1768, 0.2051, 0.1674, 0.0988, 0.1624, 0.1087
Q(k)	OFF DIAGONAL ELEMENTS=0 DIAGONAL ELEMENTS =0.1 to 1.0 (TYPE2) =0.1 (TYPE3) =0.1 (TYPE4)
F ₄	ALL ELEMENTS ZERO EXCEPT FOR THE FOLLOWING:- (2,3), (3,1), (4,2), (5,4), (6,5), (7,6)
θ	4
α	0.2
β	0.3

(i) To ensure convergence of the estimates, the values of $P(0,0)$ need to be approximately equal to $I*5$. If the values specified for $P(0,0)$ are much less than $I*5$ then the estimates tend to become biased and if they are much greater than $I*5$ then the rate of convergence is poor.

(ii) A satisfactory value for the Q matrix of the second (or Extended) type of Kalman Filter could not be found. All that can be definitely stated is that it is somewhere in the region between $0.1*I$ and $1.0*I$.

(iii) When determining the F_4 matrix for the Adaptive (type 4) filter, the procedure adopted was the same as that used in the simulated study of the Kalman Filter (see section 4.5.2): this procedure is described in detail in section 3.2.2.1. A total of ten trials were conducted before the matrix given in Table 9.3 was chosen and it is interesting to note that the first state variable, i.e. the temperature of the liquid leaving the preheater tubes, is not related to any of the elements in the model error vector, \bar{w} . From this we can conclude that this state variable is well modelled.

(iv) The values assigned to the parameters of the Adaptive Filter, i.e. α , β and θ , were the same as those used during the simulated studies of the Kalman Filter (see section 4.5.2.) i.e.

$$\theta = 4$$

$$\beta = 0.3$$

$$\alpha = \frac{1}{k} , \quad \frac{1}{k} > 0.2$$

$$\alpha = 0.2 , \quad \frac{1}{k} < 0.2$$

where k = the number of filter cycles.

During the on-line filter tuning experiments with the Adaptive Filter the above parameters were varied within the ranges given in sections 3.2.2.2 and 3.2.2.3. but this had only a slight effect on the filter's performance. This observation confirms the findings of Kilbride-Newman who states that the Adaptive Filter is robust with respect to its parameters.

The above points will be discussed in more detail in the next section of this chapter.

9.4. RESULTS AND DISCUSSION

The final stage in this research project was to carry out on-line filtering experiments with the following three forms of the Kalman Filter,

TYPE 2 (FILTER2) - The Extended Kalman Filter

TYPE 3 (FILTER3) - The Extended Kalman Filter using a state transition matrix calculated by the canonisation procedure.

TYPE 4 (FILTER4) - The Adaptive Kalman Filter.

The object of this study was to attempt to show whether or not the theoretical developments proposed in chapter 3 will stand up to the rigorous tests of an on-line application, i.e. will they promote and ensure the convergence of the Kalman Filter. The reasons for including the Extended Kalman Filter (TYPE 2) in this set of experiments are two fold. Firstly it is hoped that some of the faults which the other types of filter are attempting to cure will be displayed and secondly the results obtained using this type of filter can be used as a standard for assessing the performance of the other filters.

In all thirty three on-line filtering experiments were carried out, but since each

experiment generates at least seven graphs, only those results which highlight the points to be made during this discussion are included in this thesis. These results are to be found in Appendix K as follows,

Appendix K-1 - Results obtained using the second type of filter.

Appendix K-2 - Results obtained using the third type of filter.

Appendix K-3 - Results obtained using the fourth type of filter.

One further point which should be made before the discussion of the results commences is that during these experiments a certain amount of numerical filtering is carried out when the measurements are taken. This is done by specifying an ensemble of 10 for the analogue inputs. Thus, at every sampling time each analogue input is measured ten times and the average value transferred to the master segment of the filtering programs, FILTER (see Appendix G-2). This averaging was done to try and remove some of the more spurious measurements obtained in

the hope that this would not only ease the task of filtering but also ensure the accuracy of the inputs to the dynamic model.

The first point to note from the results shown in Appendix K is that all of the filters initially produce estimates which converge rapidly towards the measurements. This feature is most noticeable after the first cycle of the filter when, in almost every case, the estimates change sharply from the initial guess to a value near to the measured state. These observations would seem to confirm the findings of other researchers in this area (see for example (9.5), (9.6), (9.7), (9.8)) who, as was discussed in chapter 2, state that as long as the initial error covariance matrix, $P(0,0)$ is large enough then the estimates will converge on the true state no matter how poor the initial guesses are. Specifying a large value for $P(0,0)$ means that initially the filter relies almost entirely on the measurements and so, inherent in the above statement regarding the convergence of the estimates, is the assumption that during the first few cycles of the filter, the measurements

are accurate. In this particular application this assumption is not an unreasonable one because the double effect evaporator is initially at steady state.

One further observation which can be made about this initial period is that convergence is a lot slower for unmeasured state variables than it is for those which are measured. Figures K-1.2, K-2.4 and K-3.4 show that the estimates of the second effect calandria shell outlet temperature do not settle down to steady values until after the fourth or fifth cycle of the filter whereas measured state variables converge after the first or second cycle of the filter: Figures K-1.1, K-1.4, K-2.1, K-2.3, K-2.5, K-2.7, K-3.1, K-3.2, K-3.3, K-3.5 and K-3.7 are all particularly good examples of the converge of measured state estimates during the first two cycles of the filter. This type of behaviour was also observed during the simulation studies of the Kalman Filter, see chapter 4.

RESULTS OBTAINED USING THE EXTENDED KALMAN FILTER

(TYPE 2) - See Appendix K-1.

The many difficulties which are encountered when applying the Extended Kalman Filter to the estimation of the state of chemical processes are highlighted by the results shown in Appendix K-1. It has already been pointed out that the Filter Tuning experiments failed to determine suitable values for the Q matrix for this filter and the following discussion will show why this was so.

In experiment number 19, see Figures K-1.1, K-1.2, K-1.3 and K-1.4, the Q matrix was assigned the values $1.0 \cdot I$. As can be seen from the results of this experiment this caused the predictions generated by the dynamic model to be more or less completely ignored, in fact the graphs of the estimates and measurements are very nearly coincident. Figure K-1.4 is a particularly good example of this behaviour and indeed shows that in this case the predictions are tracking the measurements. This proves quite conclusively that the estimate generated by the filter are being influenced almost

exclusively by the measurements and as a result very little filtering is taking place.

In an attempt to cure the above problems the Q matrix was decreased to $0.1 \cdot I$ but as can be seen from the results of experiment number 21, see Figures K-1.5, K-1.6, K-1.7 and K-1.8, this caused many of the measurements to be disregarded after the initial period of convergence. Figure K-1.8 shows a classic example of divergence due to poor modelling. This result shows that the estimates generated by the filter are depending almost entirely on the predictions and because, as was discussed in chapter eight, the modelling of the temperature of the liquid leaving the condenser tubes is suspect in certain situations the estimates have diverged from the true state. When analysing this particular set of results one must be careful not to confuse the measured state with the true state because the measurements are almost certainly lagging behind the true state in this case. However, the fact that in Figure K-1.8 the measured and estimated states bear no relationship to one another leads to the conclusion about the divergence of the estimates.

A further problem often encountered when applying the Extended Kalman Filter to chemical processes is that of the observability of unmeasured state variables. Figure K-1.7 shows the estimated temperature for the outlet stream from the second effect calandria shell and it can be seen that after about 16 minutes there is a very sharp change in the curve. This may be an indication of unobservability and if it is then the fact that this behaviour is not displayed by Figure K-1.2 leads to the possible conclusion that the observability of a system is affected by the values assigned to the Q matrix.

The most acceptable of all of the results obtained using the Extended Kalman Filter is shown by Figure K-1.9. A comparison of this figure with the corresponding results obtained using the other two filters, see Figures K-2.6 and K-3.6, shows that the estimates generated by the Extended Kalman Filter (Type 2) are quite acceptable even though they lag behind the measurements, a fact which contradicts the assumption that the measurements lag behind the true state. This result led to a series of experiments where each diagonal element of the Q matrix was assigned a

different value from the range 0.1 to 1.0. Unfortunately, it was not possible to determine a combination of values which produced satisfactory results for all of the estimates generated during one experiment. The failure of these experiments tends to contradict the results of other researchers in this area, see for example Schlee (9.9), Jazwinski (9.10) and Wells (9.11), who concluded that the Extended Kalman Filter (Type 2) can be successfully applied to uncertain processes if the Q matrix has a large value.

As a result of the experiments discussed above the following overall conclusions can be made about the application of the Extended Kalman Filter (Type 2) to chemical processes which are poorly understood.

(i) Increasing the value of the Q matrix does not lead to satisfactory estimates being generated.

(ii) Divergence and bias are present to such a degree that the estimates generated are in general unreliable.

RESULTS OBTAINED USING THE EXTENDED KALMAN FILTER
WITH A STATE TRANSITION MATRIX CALCULATED BY THE
SIMPLIFIED CANONISATION PROCEDURE (TYPE 3) - See
Appendix K-2.

The mathematical model of the double effect evaporator developed in chapter 8, is essentially linear in nature but the process which it models is a non-linear one. Thus, in this application, the benefits of the third type of filter are entirely due to the way in which the state transition matrix is fitted to the change in state during the sampling interval. This fitting process ensures that the estimates and related statistics are consistent and consequently the filtering results are much better for this type of filter (Type 3), see Appendix K-2, than for the standard Extended Kalman Filter (Type 2) see Appendix K-1.

The two results which clearly display the superior performance of this type of filter are Figures K-2.1 and K-2.7. Figure K-2.1 shows the way in which both the predictions and measurements are being combined to produce good

estimates of the preheater tubes outlet temperature. When this result is compared with the results obtained by the Extended Kalman Filter it can be seen by a comparison of Figure K-2.1 and Figure K-1.1 that the estimates generated by the third type of filter are not dependent exclusively on the measurements nor, as shown by a comparison of Figure K-2.1 with Figure K-1.5, are they dominated by the predictions.

Figure K-2.7 is perhaps one of the best examples of how a filter should work when applied to a chemical process. The results show that the measurements and the predictions are both influencing the estimates but at the same time the process noise present in the measurements is being filtered off. This result is particularly pleasing when it is compared with the results given by the Extended Kalman Filter, see Figure K-1.4, and the Adaptive Filter, see Figure K-3.7, where there is considerable evidence that the estimates are dependent on the measurements.

Confirmation of the above findings is obtained by comparing the results shown in Figures K-2.2. and K-2.3. with corresponding results given by

the Extended Kalman Filter (Type 2), see Figure K-1.6. Note there is no evidence of bias or divergence in Figures K-2.2. or K-2.3, whereas there are serious doubts as to the convergence of the estimates shown in Figure K-1.6.

A further point which can be made about the results given in Appendix K-2 is that there is no evidence of unobservability, in particular see Figure K-2.4 and compare this with Figure K-1.7.

The only drawback of this type of filter is that it appears unable to completely compensate for gross modelling errors. A comparison of the results obtained for the condenser tubes outlet temperature, see Figure K-2.5, with the corresponding results produced by the Extended Kalman Filter, see Figures K-1.3 and K-1.8, shows that although the measurements and predictions are being acknowledged in the case of the third type of filter, the poor modelling of this state variable causes the estimates to be biased.

The overall conclusions which can be made about this type of filter (Type 3) following

the on-line experiments discussed above are,

(i) The estimates generated by this filter in general show good convergence and are influenced by both the measurements and the predictions.

(ii) There is no evidence of unobservability

(iii) The problems encountered with the Extended Kalman Filter (Type 2) when choosing the Q matrix have been overcome by this filter (Type 3).

(iv) The main drawback with this filter is that it does not completely compensate for gross modelling errors.

The final comment which can be made about this filter is concerned with its somewhat empirical nature which is due to the use of constant eigenvector and inverse eigenvector matrices for the simplified canonisation procedure. This simplification does not appear to be detrimental to the performance of the filter in any way and indeed because of the way in which the eigenvector and inverse eigenvector matrices are determined the filter is essentially fitted to the process under study. This is thought to be

one of the features of this filter which causes its performance to be greatly superior to the Extended Kalman Filter (Type 2).

RESULTS OBTAINED USING THE ADAPTIVE KALMAN FILTER

(TYPE 4) - See Appendix K-3.

With the exception of Figures K-3.6 and K-3.7, which will be discussed later, the results shown in Appendix K-3 indicate that the Adaptive Filter (Type 4) is also capable of overcoming the problems encountered when applying the Extended Kalman Filter (Type 2) to chemical processes which are poorly understood. A comparison of Figures K-3.1, K-3.2 and K-3.3, with the corresponding results for the Extended Kalman Filter (Type 2), see Figures K-1.1, K-1.5 and K-1.6, shows that the Adaptive Filter is capable of compensating for the non-linearity and uncertainty of the process. Figure K-3.2 in particular shows the way in which the Adaptive Filter uses both the predictions and the measurements to calculate the estimates. It is also worth noting here that Figure K-3.4 does not exhibit any of the signs of unobservability shown by the Extended Kalman Filter, see Figure K-1.7.

The most important of the results shown in Appendix K-3 is Figure K-3.5. This result when compared with the results of both of the other two filters - see Figures K-1.3, K-1.8 and K-2.5, shows quite clearly that the model error compensation strategy incorporated within the Adaptive Filter, greatly improves the estimates generated for a state variable which is poorly modelled. At the end of the experiment (Run 23) during which the results shown in Figure K-3.5 were produced the model error vector, \bar{w} -see equation 3.34, was found to be,

0.013, 0.021, 0.019, 0.413, 0.032, 0.029

Due to the nature of the F_4 matrix this means that the derivative of the state variable describing the condenser tubes outlet temperature is being compensated by 0.413 as against 0.013 to 0.032 for the other derivatives. This shows quite clearly the way in which the Adaptive Filter is compensating for this modelling error.

The only results the Adaptive Filter produced which were not as good as had been expected are those shown by Figures K-3.6 and K-3.7. These results show that the estimates are relying

almost entirely on the measurements. This is not so surprising for the estimates of the second effect tubes outlet temperature, see Figure K-3.6, because it has already been shown see Figures K-1.9 and K-2.6, that this variable is not modelled as well as the dynamic analysis discussed in chapter 8 indicated. However, both the dynamic analysis, see Figures J-4.7 and J-5.7, and the results generated by the third type of filter, see Figure K-2.7, show that the predictions of height in the second effect separator are accurate and so it is surprising that the estimates generated by the Adaptive Filter rely on the measurements. In an attempt to cure this problem column 6 of the F_4 matrix was removed, i.e. the second effect separator is now assumed to be correctly modelled. The results obtained thereafter were very similar to those shown by Figure K-2.7. From this it can be concluded that in cases where the model is known to be correct no model error compensation should be attempted.

As a result of the experiments discussed above the following overall conclusions can be made about the application of the Adaptive Filter

(Type 4) to chemical processes which are poorly modelled.

(i) The estimates generated by the Adaptive Filter generally show good convergence and are influenced by both the measurements and the predictions.

(ii) There is no evidence of unobservability.

(iii) The problems concerning the choice of the Q matrix have been overcome by the Adaptive Filter.

(iv) The model error compensation strategy is capable of overcoming gross modelling errors.

(v) For variables which the model is known to describe accurately no error compensation should be attempted.

The final comment which can be made about the Adaptive Filter is that despite its increased computational requirements it is the most effective of the filters tested during these on-line experiments

9.5. CHAPTER REVIEW

The theoretical developments proposed in chapter 3 were combined with the dynamic model developed in chapter 8 to produce two modified forms of the Extended Kalman Filter. These filters were used to carry out on-line filtering experiments under the control of the OLDFP Executive and the results compared with the standard Extended Kalman Filter. The results obtained from the on-line experiments led to the following general conclusions,

(i) The Extended Kalman Filter (TYPE 2) does not perform satisfactorily during on-line applications to non-linear processes.

(ii) The Extended Kalman Filter using a state transition matrix calculated using a canonisation procedure (TYPE 3) overcomes many of the problems reported previously for non-linear applications.

(iii) The Adaptive Kalman Filter produced results during the on-line experiments which show it to be capable of overcoming the effects of poor models.

(iv) The dynamic model of the double effect evaporator developed in chapter 8 can be improved particularly with respect to the modelling of the condenser tubes outlet temperature.

CHAPTER 10

CONCLUSIONS AND RECOMMENDATIONS

FOR FUTURE WORK

10.1. CONCLUSIONS

The main aim of the research reported in this thesis is to overcome some of the problems associated with the application of the Kalman Filter to chemical process plant. These problems were discussed at length in chapter 2 and those areas requiring further study were found to be the application of the Kalman Filter to (i) processes whose dynamic characteristics are poorly understood and (ii) to highly non-linear processes. When either of the above conditions is true then the estimates generated by the filter, in particular those variables that are not available as measurements, may exhibit bias and in extreme cases diverge from the true state.

Two theoretical developments which have been proposed to combat these problems are the Adaptive Kalman Filter first reported by Kilbride-Newman (10.1) and a form of the Extended Kalman Filter which uses a canonisation procedure to calculate the state transition matrix. The results of simulation studies of various forms of the Kalman Filter led to the general conclusion that the modified forms of the filter displayed an

improvement in performance over the classical Kalman Filter in that they ensured and promoted the convergence of the estimates generated.

Following the success of these off-line experiments it was decided to apply the modified filters to the estimation of the state of a pilot plant scale double effect evaporator whose dynamic behaviour is poorly understood. In order to do this it was necessary to develop two major software packages. The HADIOS EXECUTIVE PACKAGE is an interactive data acquisition program based on the BASIC-16 language and was used in the early stages of this research. The On-line Digital Filtering Package (OLDFP) is a more sophisticated data acquisition system which is based around the OP-16 real time operating system. Both of these packages have proved to be extremely successful and because of their general nature their use is not confined to applications involving the double effect evaporator.

The next stage in this research was the development of a mathematical model to describe both the steady and dynamic states of the double effect evaporator. As a result of this

modelling exercise a seventh order mathematical model together with suitable heat transfer correlations and compatible with the Kalman Filter was derived and then tested by comparing the simulated responses with plant data. Although offering scope for improvement in accuracy, the predictions produced by integration of the dynamic model equations were close enough to the experimental responses obtained following two types of step change for the model to be adopted for on-line estimation.

The on-line filtering experiments carried out with the double effect evaporator proved to be rigorous and exacting tests for the theoretical developments which have been proposed in this research. This is mainly because the process taking place at the double effect evaporator is not controlled and so due to their dynamic nature these experiments constitute a more realistic test of the Kalman Filter than do those reported for closed loop experiments. After a comparison of the results obtained the following general conclusions were made:-

(i) The Extended Kalman Filter requires some strategy to eliminate bias and divergence when it

is to be applied to this sort of process.

(ii) The use of the canonisation procedure to calculate the state transition matrix improves the performance of the Kalman Filter because the estimates and error covariances generated by the filter are now consistent. When using the simplified canonisation procedure, i.e. the eigenvector matrix is assumed to be constant, this form of the Kalman Filter is somewhat empirical in nature and so, once the eigenvector matrix has been calculated, is process dependent. However, this is not a defect in the filter because it enables the fitting of the Kalman Filter to the process under study. In view of the increased computational requirements of more sophisticated filters and the uncertainty present in the modelling of most chemical processes this feature must be a considerable asset to this type of filter. One failing of this type of filter is that it is unable to cope with gross modelling errors and so it can only be applied with mathematical models of proven accuracy.

(iii) The main asset of the Adaptive Kalman Filter is its model error compensation strategy. The results obtained show that this filter

efficiently removes errors due to poor modelling. The main drawback with the Adaptive Filter is the time needed to determine the F_4 matrix.

(iv) When using either the Adaptive Filter or the modified Extended Kalman Filter the problems associated with the choice of Q matrix are overcome.

(v) The cycle time (30 seconds) achieved during the on-line filtering experiments shows that the application of these modified forms of the Kalman Filter is a feasible proposition.

10.2. RECOMMENDATIONS FOR FUTURE WORK

At the conclusion of any research project although a large proportion of the problems originally investigated have been solved, it is inevitable that further areas requiring some study are uncovered. As a result of this particular project it is apparent that further work is required in the following areas.

(i) There is considerable scope for further research in the area of dynamic modelling of the evaporator. In cases such as this, where

the dynamic behaviour is poorly understood, it would be interesting to see whether better models could be obtained by the use of a pseudo random binary sequence technique for calculating transfer functions - see Momen (10.2).

(ii) Further work needs to be done into the interpretation of the results generated by the Adaptive Filter regarding modelling errors. Ultimately it would be preferable to use the Adaptive Filter as a means of improving the model and then be able to revert to a simpler form of Kalman Filter.

(iii) The techniques which are now available for improving the performance of the Kalman Filter in chemical engineering applications are sufficiently sophisticated for this means of estimation to be regarded as reliable. Thus, further research could now be usefully carried out into the use of the Kalman Filter in on-line control applications.