University of London

IMPERIAL COLLEGE OF SCIENCE & TECHNOLOGY

Department of Electrical Engineering

THE DESIGN OF MICRO-PROCESSORS FOR DIGITAL PROTECTION OF POWER SYSTEMS

By

EDGAR HORNE

M.Sc. (Eng).

Thesis submitted for the degree of Doctor of Philosophy in the Faculty of Engineering.

LONDON, August 1978

ABSTRACT

The use of digital computers for power system protection has been widely studied during the past decade. In many respects it has been shown that digital techniques can offer an improvement on existing analogue protection methods. However, inter-related technical and economic problems, largely arising from the use of conventional computer hardware, present a serious obstacle to the practical application of these techniques. Most of the difficulties associated with a typical multi "mini" computer digital protection scheme were encountered during previous research at Imperial College.

In the work described here an alternative approach based on dedicated protection equipments employing microprocessors is presented. The design parameters for the data acquisition and processing functions of a dedicated digital relay are analysed. A prototype equipment featuring a high performance, bit-slice, bipolar, 16 bit microprocessor is described. The processor architecture is structured to optimise the performance of the relay and both the data acquisition interface and processor are designed to provide maximum flexibility of application.

To validate the design and to demonstrate the operation of the complete relay, two typical applications are included. The applications include generator negative sequence protection and a combined overcurrent - earth fault implementation which provides a range of optional characteristics and logical switching control.

ACKNOWLEDGEMENTS

The work in this thesis was carried out under the supervision of Dr.B.J. Cory, B.Sc(Eng), D.Sc, ACGI, C. Eng. FIEE, Reader in Electrical Engineering, Imperial College of Science and Technology, London. I wish to thank Dr.Cory for his constant guidance and support with both the technical and other aspects of this project.

I would also like to record my gratitude to my colleagues in the Power Systems Laboratory. I am particularly indebted to Mr. L. Petrou for his valuable assistance throughout the work, and to Dr. C.B. Giles, Mr. F.C. Chan and Dr. A. Traca for their constructive discussions.

Finally my thanks are due to Messrs E. Paddison, E. Walker and T. Lomas of G.E.C. Measurements Ltd. for the interest they have shown in the project. The equipment and constructional facilities supplied by G.E.C. Ltd. are greatly appreciated.

The project was supported by the Science Research Council.

To my parents.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 The Role of the Protective Relay.

An electric power system will inevitably suffer faults, brought about by either the gradual deterioration of the system components or by transient natural phenomena. Measures to minimise the effects of such faults are essential if massive supply disruption and castastrophic plant damage, are to be avoided. These measures form the basis of the "protection" task, a task which has assumed increasing importance and complexity as power systems have developed.

Protective "relays" capable of detecting fault conditions, fulfil a vital function in the overall protection task. The evolution of the modern interconnected power system, employing ever larger and more highly rated units of generation, has created a continuous demand for relays of improved performance.

The advantages of using new digital electronic methods for the construction of relaying equipment forms the subject of investigation of this thesis.

1.2 Relay Implementation

The operation of a protection relay may be defined as: i) the measurement of the primary power system parameters, ii) the evaluation of the measured data and iii) a decision process based upon the results of the evaluation. These functions must be performed rapidly and accurately, with

high reliability and security.

Initially, relaying equipment employed electro-mechanical or electrothermal techniques to fulfil the evaluation and decision tasks[1]. Subsequently to provide higher performance equipments, i.e. improved response times, accuracy and reliability, with more sophisticated characteristics to meet the increasing complexity of the protection function, later relaying devices have been based on "static" processing units. Analogue electronic relays now form the basis of current practice[2] and whilst representing a substantial improvement on earlier versions, several problems are still apparent.

The majority of the limitations arising in relays of this type, are those which are common to all systems employing electronic analogue processes. They are:

(i) Initial accuracy.

The construction of analogue equipment with signal processing errors of less than a few percent as required by the relaying function, demands the use in many areas of the circuitry of high quality, close tolerance components which in themselves are expensive. They also greatly increase the task of quality assurance prior to use. Even with such components, an initial calibration effort is still generally essential for each equipment, to nullify the parameter variations which remain, for example, gains and offsets.

(ii) Long term accuracy.

Although the initial accuracy of analogue equipment can be adjusted to meet given tolerance limits, the problems of long term accuracy still exist. Currently protection equipment service life is regarded as being in the order of 15 years. During this period the parameter drifts to which electronic components, especially passive components and connectors, are prone, may cause deterioration in the performance of analogue relays. The equipment user is thus faced with the need for preventative maintenance to combat these effects.

(iii) Reliability.

Performance studies[3] show that the reliability of static electronic relays is undoubtedly superior to that of previous equipment, providing component quality is assured. However, the nature of analogue circuits, entailing the use of many discrete parts, prevents the widescale application of recent advances in semiconductor large scale integration techniques. By reducing component interconnection these techniques can greatly improve the reliability of electronic equipment.

Further, the inclusion of automatic self test facilities within an analogue relay presents only limited possibilities for reliability improvement, since the considerable additional complexity involved will tend to offset the confidence which can be placed in this approach.

(iv) Flexibility

Despite the availability of static analogue relays for a wide range of applications, individual relay adjustment or modification will frequently be necessary to accommodate the characteristics of specific tasks. If the adjustments lie beyond the scope of permissable preset or configuration variations, analogue equipment will require costly circuit re-design and possible housing re-wiring. In these cases, much of the flexibility attributable to the modular construction methods usually employed, is lost.

During the past 10 years the availability of low cost digital computers has led to widescale interest in the possibilities of implementing the processing and decision functions of protection relays by digital methods. This could overcome many analogue equipment limitations, and also provide improvements in several areas.

The considerable research effort which has been directed to this end has demonstrated that, in principle, no insurmountable difficulties arise, and several advantages, discussed fully in Chapter 2 exist: These are,

(i) Improved absolute accuracy and long term stability of performance.

(ii) Enhanced reliability.

(iii) Greater flexibility.

(iv) Improved fault detection methods.

(v) Closer protection characteristic matching.

(vi) Capability for data recording, allowing post fault analysis.

(vii) Diagnostic software facility, simplifying maintenance procedures.

(viii) Off-line support aids, enabling rapid evaluation of relay performance during the design and commisioning stages of a protection scheme.

(ix) Standardisation of hardware.

Despite the considerable attractions which these listed benefits appear to offer relay manufacturers and users, digital methods have so far found little application for practical relaying purposes. The constraining factors, described in the following chapter, lie not in the fundamental digital techniques per se, but in the limitations imposed by currently available computing equipment.

Rapid progress which has occured in semiconductor technology, particularly in the field of integrated circuit digital devices, has now made feasible a new approach to protection relay design, based on dedicated microprocessor equipments.

These relays offer a solution to the majority of the application difficulties that have inhibited the acceptance

of previously proposed schemes, whilst retaining the many advantages which accrue from the use of a programmable digital computer.

1.3 Organisation of the Thesis.

This thesis describes the work carried out by the author on the design and construction of a dedicated digital protection relay, using micro-processor techniques.

To demonstrate the operation of the complete equipment two typical protection applications are described.

Chapter 2 reviews the history of digital power system protection, and analyses the disadvantages of the methods proposed which employ mini-computers as the processing element. The concept of a dedicated micro-processor based relay is developed and its advantages are discussed. Finally the implications of proposed hierarchical digital protection schemes are investigated

Chapters 3 and 4 deal at length with the requirements and design factors of the micro-processor relay sub-units. In Chapter 3, the relay data acquisition interface is described, and the performance parameters of the major components used within this unit are examined in detail. Chapter 4 describes the design of a specialised, 16 bit micro-processor, which forms the second unit of the relay. To highlight the design choices made, the requirements of the unit are discussed, and compared with a representative

survey of the many micro-processor types available.

The first of the application examples is described in Chapter 5. In this example, the relay is used to provide generator unbalanced load protection. The essential protection characteristics and performance of the relay are tested and compared with that of current static analogue equipment. In addition several improvements, made possible by digital techniques, are included and discussed.

The penultimate chapter, Chapter 6, deals with the second application, that of overcurrent and earth fault protection. This application is designed to demonstrate the flexibility of the micro-processor relay. A modular software structure is employed which provides several optional protection characteristics. Other functions, including a self test facility and simple auto-reclose are also programmed.

Finally, the conclusions summarise the overall work which has been carried out and indicate several areas which require further investigation and improvement.

Throughout the main chapters of this thesis descriptions of the dedicated relay design have been generalised and only functional block diagrams are included. Comprehensive hardware and software details of the equipment are provided by the appendices.

The contributions offered by this thesis are:

(i) identification of the application problems associated with the single computer "integrated" approach to substation protection. (Chapter 2).

(ii) the development of an alternative approach based on "dedicated" micro-processor relaying equipments (Chapter 2).

(iii) a study of the performance specification required for the data acquisition and computation units which form a dedicated relay. (Chapters 3, 4).

(iv) the design and construction of a digital relay comprising a self-contained data acquisition interface and an advanced high speed 16 bit micro-processor (Chapters 3, 4, Appendices).

(v) the development of software for the negative sequence protection of generating plant and on-line testing of the digital relay in this application. (Chapter 5).

(vi) a software structure is proposed which enhances the flexibility of the relay and provides a basis for the future development of a high level task oriented programming language. A specific example for overcurrent and earth fault protection is developed and tested on-line. (Chapter 6).

CHAPTER 2

DIGITAL PROTECTION OF POWER SYSTEMS

2.1 Digital Computer Applications in Power Systems

Digital computers are already well established in several areas of power system control, particularly where the rapid and reliable evaluation of data supplied from a multiplicity of sources is required. Typical examples of this type of application are found in network control and dispatch centres[4], and in power stations[5].

In a less centralised configuration, computers have recently been employed for tasks within transmission system sub-stations. Data acquisition for control and analysis purposes[6] is an increasingly common application. Also at sub-station level, automatic switching and reclosing routines for circuit breakers and isolators, have been implemented by programmable digital equipment[7].

The extension of digital methods to other substation functions, particularly those, concerned with system protection, is an obvious area of development.

Following an original assesment of the task by Rockefeller[8] in 1968, numerous studies of digital protection methods have been undertaken. The majority of these studies have been concerned with the development and

performance comparisons of algorithmic techniques for the detection of fault conditions in specific power system components. In particular, much attention has been devoted to methods of overhead transmission line protection [9,10,11]. Other topics including bus zone[12], transformer [13,14], generator[15] and back up protection[16,17], have also been examined from the digital viewpoint.

2.2 Advantages and Disadvantages of Digital Protection.

The studies mentioned above, and the work undertaken at Imperial College, have provided an insight into the potential benifits and shortcomings of digitally based protection schemes. The major advantages are:

i) Improved initial and long term accuracy.

With the exception of a small part of the computer/power-plant interface, processing operations are performed by digital methods. These methods provide drift free performance, with accuracies limited only by the word length and number representation methods of the computer.

ii) Enhanced reliability.

The reliability of a digital protection scheme is improved, in comparison to its analogue counterpart, by several contributory factors. From a hardware viewpoint, digital equipment benifits from the availability of a wide range of L.S.I devices. The use of these devices greatly reduces component interconnection which forms a

major source of failure in electronic systems. Additionally, digital circuits use fewer passive components, i.e. capacitors and resistors, the reliability of which is usually less than that of active semiconductor devices.

A further enhancement of digital equipment reliability is provided by its programmable features which facilitate the inclusion of automatic self test and error detection routines within the operating software. The possibility of undetected equipment failure is thus much reduced.

iii) Off-line support.

A feature, closely allied to reliability, is the availability of many off-line computing techniques which can provide support for the development and testing of on-line digital equipment. These techniques include software development and performance assesment aids, hardware functional emulation packages and complete system simulation.

iv) Improved protection performance.

The fault detection performance of a digital relaying scheme depends to a large extent on the algorithm used in the computation. As previously described, algorithm studies have formed the basis of the majority of digital protection work which has been undertaken. In consequence, methods are available which provide a detection accuracy, sensitivity and in some cases, a speed superior

to that achievable with analogue relays.

Digital methods also allow a closer matching of relay-plant characteristics, thereby improving the tripping performance and stability of the protection over a wide range of load and fault conditions. An example is, the provision of a rectangular or trapezoidal R-X characteristic for impedance relays, dependent upon prefault power flow.

Although not directly concerned with the primary fault clearance function, data recording and post fault analysis are valuable additions, made possible by the storage of a digital scheme.

v) Greater flexibility.

The stored program design of the digital computer offers greater flexibility in the relaying task. In the initial development stages of a protection scheme, specific relay functions can be defined for common hardware by the inclusion of appropriate software.

Subsequently, alterations in plant configurations or parameters can be readily accomodated by program modifications, thus avoiding protracted and costly equipment redesign, re-wiring and testing.

vi) Simplified manufacture and maintenance.

Construction of relaying equipment using standardised programmable digital hardware provides

considerable advantages to both the manufacturer and user. These are: i) limited range of components which simplifies the assembly process and reduces maintenance replacement requirements. ii) the replacement of a defective digital device restores equipment performance to its orginal specification without the need for re-calibration procedures. iii) hardware standardisation eases the burden of personnel training faced by the equipment user. iv) diagnostic software routines can be employed to locate and isolate faults.

vii) Improving technology.

Most of the advantages listed above are common to all digital equipment. They are responsible for its widescale acceptance in numerous tasks which have traditionally been the province of analogue methods. As a result the availability of digital components is excellent.

Developments in digital technologies continue apace , providing in many instances, direct device replacements with improved specifications. These factors eliminate many of the development and replacement constraints imposed by the unique nature of an analogue equipment.

Although digital techniques offer considerable scope for the improvement of protection equipment, their adoption introduces several disadvantages. The most significant of these are:

i) Increased complexity.

Power system protection relays are required to process continuous analogue signals. Before such signals can be processed by digital methods they must be converted to a suitable format. This conversion requirement inevitably increases the complexity and cost of a digital scheme.

Additionally, after conversion, parallel multi-bit signal paths are usually required in a digital processor to manipulate data. In consequence digital devices are more complex than their analogue counterparts in which single data paths are used.

ii) Testing difficulties.

To provide the multi-bit data paths described above, sophisticated circuitry and integrated devices have been developed. The complete testing of these devices for all combinations of input, output and control signals is virtually an impossible task. Undetected logical errors may thus remain in a digital processor. Clearly, such errors will be common to all equipment used in duplicated or triplicated high reliability systems. It is therefore necessary to develop different computation methods for each equipment to ensure that a single common failure mode does not totally disable the system.

iii) Interference.

High levels of electromagnetic radiation frequently occur in the power system relay environment. Digital equipment is particularly susceptible to external noise induced malfunction. The corruption of a single bit within, for example, a processor instruction word, may result in completely erroneous operation. To prevent this, digital relays will require more extensive isolation and screening than electromechanical or static analogue types.

In new installations these requirements can be easily catered for, but the modification of existing plant presents a more difficult task.

iv) Retraining of personnel.

The application of digital techniques involves many changes in established and proven practices for the design, operation and maintenance of protection schemes. To implement these changes, personnel re-training and documentation modification are essential, with consequent additional costs.

In addition to these general shortcomings, digital protection schemes employing a single processor, normally a mini-computer, programmed for a mutiplicity of relaying tasks, suffer from several specific drawbacks. These are discussed fully in the next section.

2.3 A Critique of Single-Processor Protection Schemes.

The applications of digital techniques to power system functions reviewed in section 2.1 have without exception employed a 'mini' or larger computer as the processing device. In many cases, for example, those of network control, data logging and monitoring, the use of a large computer is often essential. Those tasks usually have hardware and software requirements which are only supportable by the more powerful machines. Typical requirements are comprehensive bulk storage facilities in the form of magnetic tape or disc, and multiple input-output devices for data collection and operator communication. From the software standpoint the implementation of sophisticated techniques, for example, state estimation methods, are only economically achievable by the use of comprehensive high level languages. In general, these languages are most efficiently implemented on large processor installations.

For protection functions, however, the use of mini-computers imposes inter-related economic and technical constraints upon the schemes proposed, most of which were encountered when a system was constructed at Imperial College, as shown in Fig.2.1.

This equipment has provided a valuable tool for the development of basic digital protection philosophy.

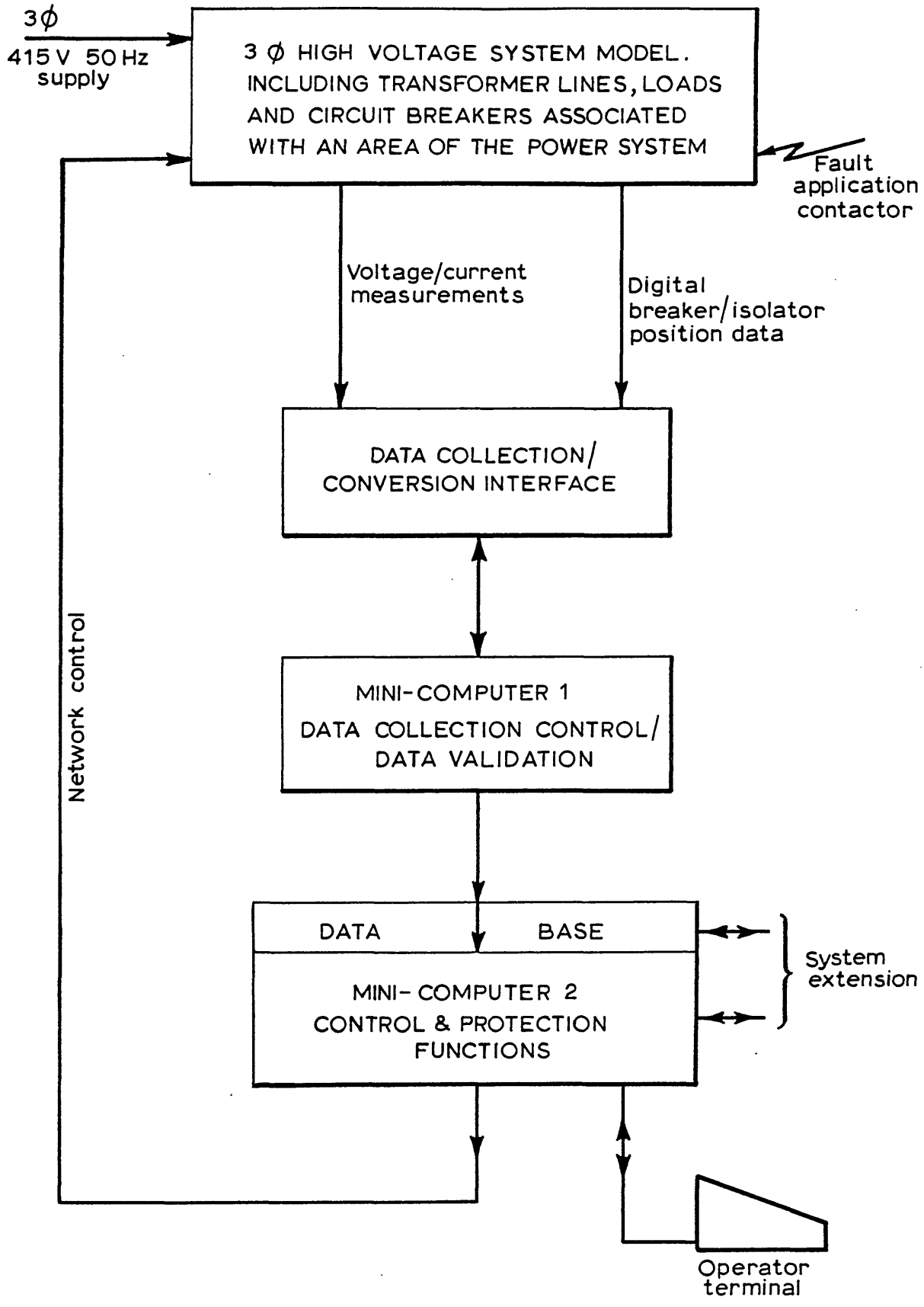


Fig. 2·1 The integrated protection scheme.

For example, overhead line protection by Ranjbar[9], transformer protection by Ong[13], substation switching control by Traca[18], and data validation by Gonzales[19]. The limitations of the configuration for practical relaying purposes became apparent [20], stemming from the need to satisfy the stringent reliability, security and availability criteria which characterise the protection task[21].

Inevitably in any type of main protection scheme, to meet these criteria, a duplication or even triplication of the relaying equipments involved is necessary. From an economic viewpoint, therefore, a multi-minicomputer protection scheme will represent a substantial investment in hardware, despite the considerable reductions in cost which the manufactures of these machines have achieved.

These economic considerations, have given rise to the concept of "integrated protection", evident in many of the published proposals. In such a scheme multiple relaying functions are embodied in a single computer which can be duplicated to produce the required reliability. Whilst this approach ensures maximum usage of the installed computing power, it also gives rise to several difficult technical problems.

The main areas of difficulty are:

i) Complex software.

The software required for the integrated protection system is classified into two distinct groups; (i) the algorithms capable of performing the required protective functions, and (ii) an"operating system" which is responsible for the organisation of these routines. As the extent of the integration increases, the complexity of the operating system software also increases and thus its operational reliability deteriorates.

ii) Initialisation.

The initalisation of protective routines is basically a function of the operating system mentioned above. In this context the correct identification of the appropriate plant protection routine at the onset of a fault condition entails a complex programming task.

iii) Response time.

Due to the sequential nature of the digital computer the time consumed by the organisational software during a fault may well result in a degredation of clearance speeds, particularly in cases of incorrect initalisation selection. This factor obviously assumes increasing importance as the power system load approaches its maximum.

iv) Reliability

A high level of hardware and software reliability

is required by the integrated system. Any major failure in either area will affect a substantial portion of the protection facilities. The provision of high reliability systems would impose further economic penalties.

v) Design, operation and maintenance.

The embodiment of the relaying functions for an area of a power system within a single computer software package, represents a more substantial departure from existing practice than simply a change of technology.

The new concepts involved intensify the problems discussed in section 2.2, which face users of protection equipment.

From the foregoing it is clear that, the limitations which are a result of the integration of protection processes within a single computer, greatly detract from the advantages of a digital relaying scheme. They present a major obstacle, and seriously restrict the practical benefits to power system operators, which a decade of research has shown to exist.

2.4 An Alternative Dedicated-Processor Approach.

To overcome the application difficulties of the integrated protection scheme, whilst retaining the advantages of a digital system, an alternative method for relay implementation has been developed.

Current electromechanical, or static analogue relaying practice, is based on a dedicated parallel unit organisation. Each function within a protection scheme is served by an individual relay. This type of configuration, which is a result of the historical development of relaying equipment, has several inherent advantages. By limiting the complexity of individual relays, the dedicated approach enables a high speed of operation to be achieved. Similarly, the reliability of the equipment is enhanced by the limited complexity and by the restricted influence which each relay has on the primary system plant. The system also provides good flexibility. It is readily expandable by the addition of further relays, although recent trends in pre-wired equipment racking are responsible for some loss of this flexibility[22].

Existing protection installations may thus be regarded as multi-processor systems, in which each processor is either an electromechanical or electronic relay. Recently, the value of the multi-processor solution, as a means of satisfying demanding task requirements, has aroused considerable interest in the digital computing field.

The important criteria for the majority of computer applications are reliability, operation speed, task complexity and economic considerations. Involved relationships exist between these factors. For example,

Fig.2.2 illustrates the areas of application for various processor types in respect of task complexity and reliability.

Complete protection of an area of a power system is a task of considerable complexity, which additionally places a premium on the speed and reliability criteria. A multi-processor approach, imitating existing relaying methods, is most applicable for a protection scheme employing digital equipments. The use of multiple task dedicated processors eliminates the need for software integration of protection functions, thereby avoiding the shortcomings of the single processor scheme.

Until recently, digital multi-processor networks have been prohibitively expensive for all but a few very high reliability or very high speed on-line control applications. However, major advances in semiconductor technology have radically altered the economics, design and capabilities of digital systems. These developments have greatly improved the speed-power performance ratio, shown in Fig.2.3, and integration density, in Fig.2.4, of digital integrated circuits.

As a result of this development, many of the components used within a digital computer, i.e. the central processor, memory and interfacing devices, are now available as single, low cost, integrated circuits. With these new components, the inclusion of specialised, task

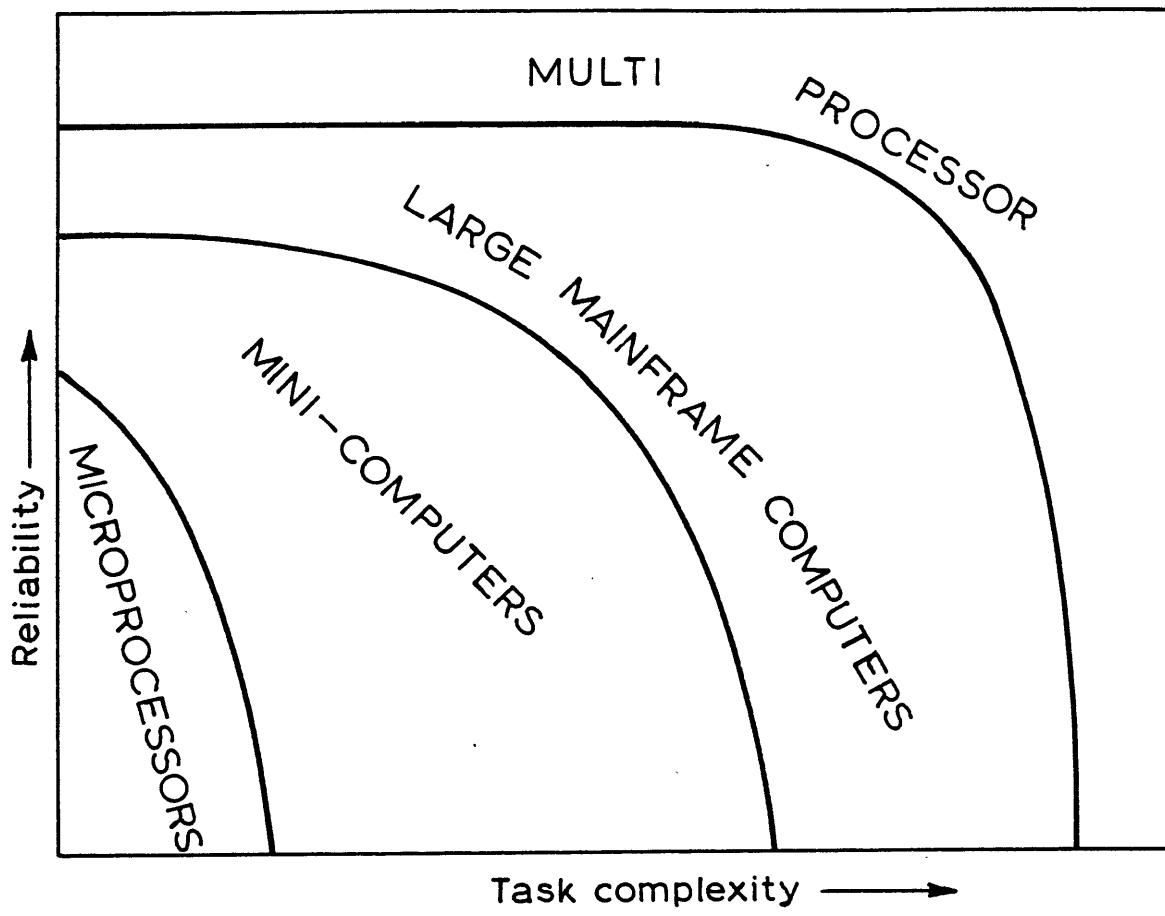


Fig. 2·2 Processor capabilities.

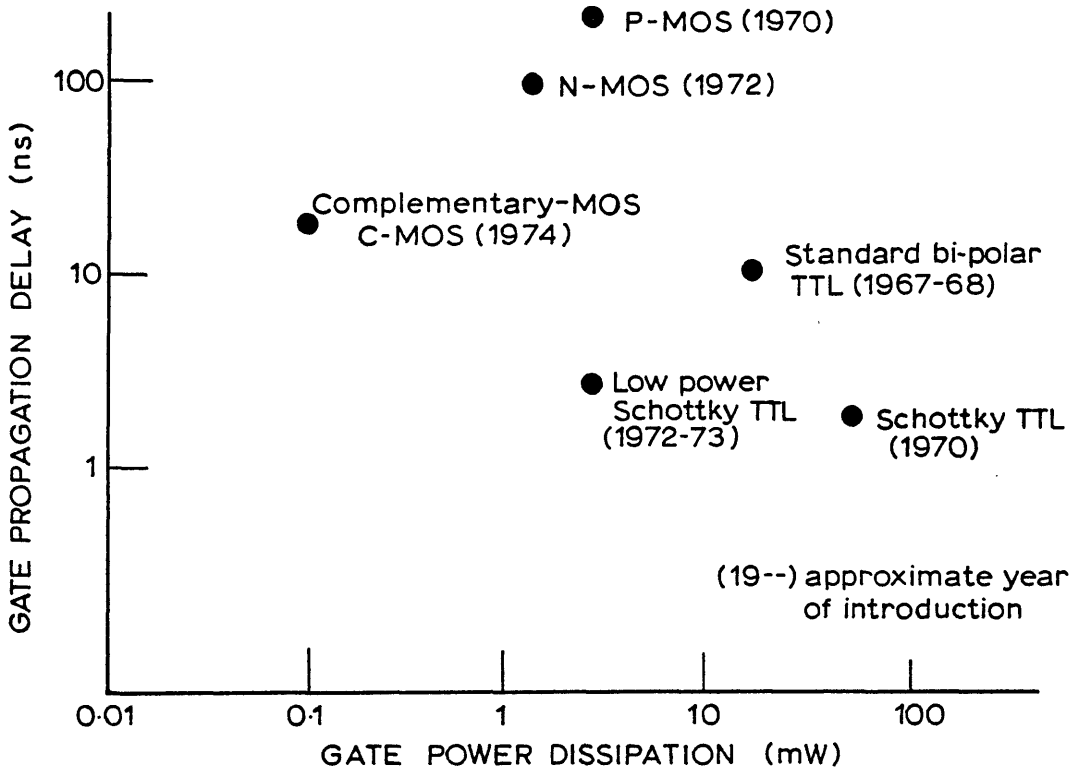


Fig. 2·3 Speed *versus* power for semi-conductor technologies

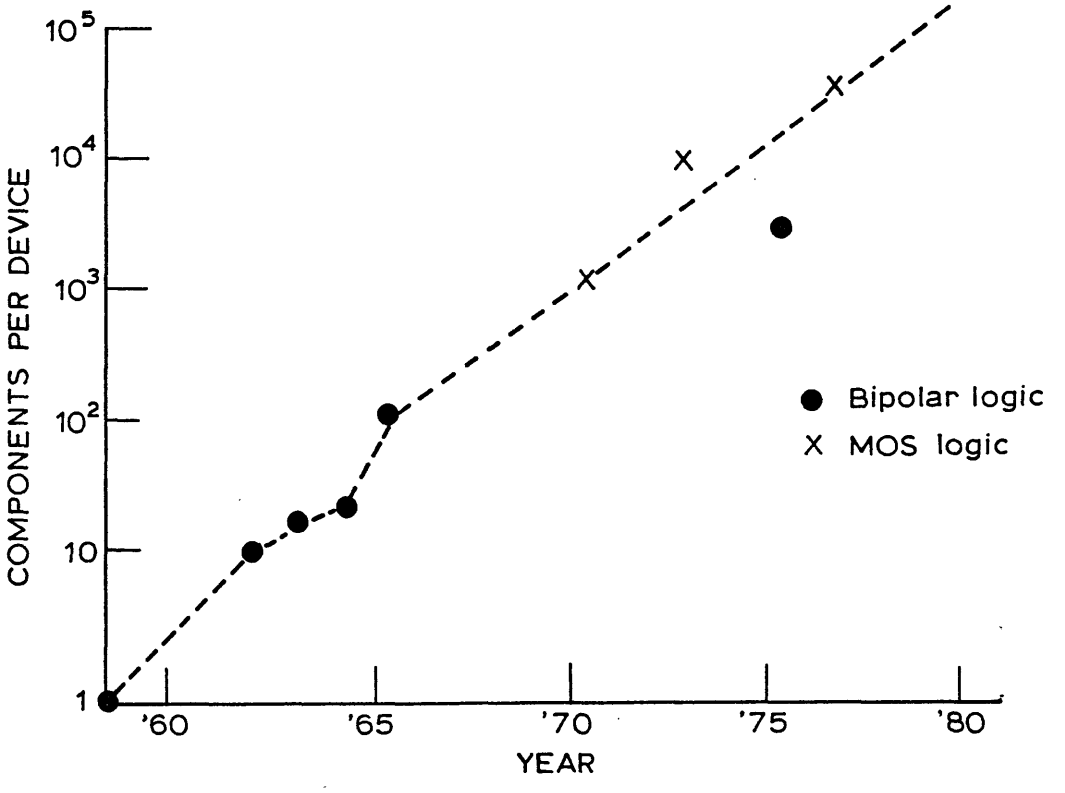


Fig. 2·4 The integration density of semi-conductor digital devices.

oriented processors, termed microprocessors, within a wide range of equipment has become economically feasible.

A microprocessor based relay has been designed, which enables the techniques of digital power system protection to be realistically exploited, and will be described.

2.5 The Dedicated Digital Relay.

To fully utilise the benifits which a stored program digital relay provides, a structure must be adopted which allows flexibility of application to be achieved with standardised hardware.

Many of the advantages of digital computing equipment stem from the ability to alter its function by program control. The implementation of protection relays by means of special purpose application hardware which must be tailored to each task is, therefore, clearly inapplicable for digital techniques.

For reasons discussed below, a structure for the dedicated relay which is broadly similar to that of integrated mini-computer system is considered to be most satisfactory. In this configuration, the data acquisition and processing functions are separated. A relay thus comprises two units. They are:

i) the data acquisition interface and ii) the protection

processor. This division of the relay components is dictated by the widely differing nature of the two functions.

The data acquisition interface is primarily concerned with data collection, and as such it must be able to fulfil many varying input requirements. By employing modular construction methods for this unit, its capabilities can be matched to specific applications in an economic manner.

Flexibility of the interface is further improved if its operations are controlled by a programmable device. The unit then becomes self contained and can be used for many acquisition tasks, for example, as the input interface of a mini-computer based monitoring or back-up protection system described in section 2.7. The internal interface controller also relieves the main protection processor of the simple, but time consuming data acquisition control overheads. These control actions do not require the computational power of the protection processor, however, they do involve relatively long waiting periods while analogue devices within the interface operate. To service these requirements efficiently the main processor would need a program interrupt facility which adds considerably to its hardware and software complexity.

The second unit of the relay, the protection processor, is essentially a high performance digital computer. It performs all the computational tasks required

by the relay which are determined by its application program.

Irrespective of the relay function, this unit can comprise common hardware, with the exception of memory capacity variations. Again modular construction can be used to advantage, in that it greatly simplifies maintenance and testing of the equipment.

To fully perform its function the processor must communicate with the acquisition interface, circuit breaker control equipment and possibly with a remote supervisory computer. Both internal and external communication channels are thus required by the relay. Their design must ensure that the flexibility of the equipment is not restricted by data transfer operations. In this respect the communication interfaces should function with a minimum of control intervention from either the protection processor or external sources. Equally, simple data transfer and data bus connection protocols are important to allow for expansion of the communication network. These considerations will be discussed further in later chapters.

Finally, since the processor forms a basic unit of every digital relay it could, in many applications, be usefully pre-programmed with basic software. For example, data input and output routines including circuit breaker control, diagnostics, monitoring facilities and alarm functions.

A digital relay hardware configuration must fulfil

the data acquisition and subsequent processing functions which are fundemental to power system protection. The double unit structure proposed here enables a specific relay to be assembled by the appropriate choice of interface input modules and processor program.

Greater hardware integration of these functions will cause a loss of flexibility. This gives rise to either uneconomic solutions for simple protection tasks, or the need for modification to accommodate tasks of high complexity.

2.6 Component Units of a Digital Protection Relay.

The basic operations of the two relay sub-units are defined as:

i) The data acquisition interface.

At regular intervals, the interface simultaneously samples the primary a.c. quantities required by the protection function. After sampling, the analogue values are stored and then sequentially converted to their digital equivalents. This digital data is deposited into an output buffer store to await transfer to the protection processor.

An input facility for direct digital plant information, for example, switch status indicators or intertrip signals is also provided by the interface.

For the reasons described in the previous section,

control of the unit is best achieved by a programmable device.

Unlike the data acquisition system of the integrated protection scheme, no provision is made in the dedicated relay interface for data pre-processing. Analogue manipulation of the primary transducer outputs, other than filtering, is avoided, as this introduces the problems of drift, offset and stability associated with analogue methods. Equally digital pre-processing, e.g. data validation, is performed, if required, by the second relay unit, the protection processor.

ii) The protection processor

From the sampled data provided by the acquisition unit, the protection processor must determine the condition of the protected power system plant. To achieve this the processor executes the particular algorithm(s) to which it is dedicated. The alogorithm will include fault detection routines, a protection characteristic, switching logic and possibly digital filtering and validation of the input data.

An emphasis must thus be placed in this unit upon the ability to perform logical and arithmetic operations. These considerations greatly influence the design of the processor. In particular, word length, operating speed, storage capacity and processor communication are important factors.

Figs.2.5 and 2.6 are outline illustrations of the data acquisition interface and protection processor respectively. The detailed design of these two units is described in the following chapters.

2.7 Monitoring and Back-Up Protection

The dedicated digital relay, described in 2.6, can be employed directly as a replacement for the electromechanical or static analogue equipments currently in use. In a conventional scheme, each relay performs a specific "main" protection task, with "back up" provided by duplication. This arrangement does not, however, take full advantage of the features provided by the digital system.

The addition of a monitoring computer, either within the relay installation, or at a remote location will greatly enhance the performance of the overall scheme. By enabling operator communication with the relay network, the monitoring computer provides an outlet for relay alarms, and abnormal power system condition messages. The ability to communicate with the main protection also allows rapid post fault analysis, and will simplify diagnostic and maintenance operations.

An extension of the monitoring function to include back up protection tasks provides further benefits. A proposal by Cory, Dromey and Murray[23] combines the largely

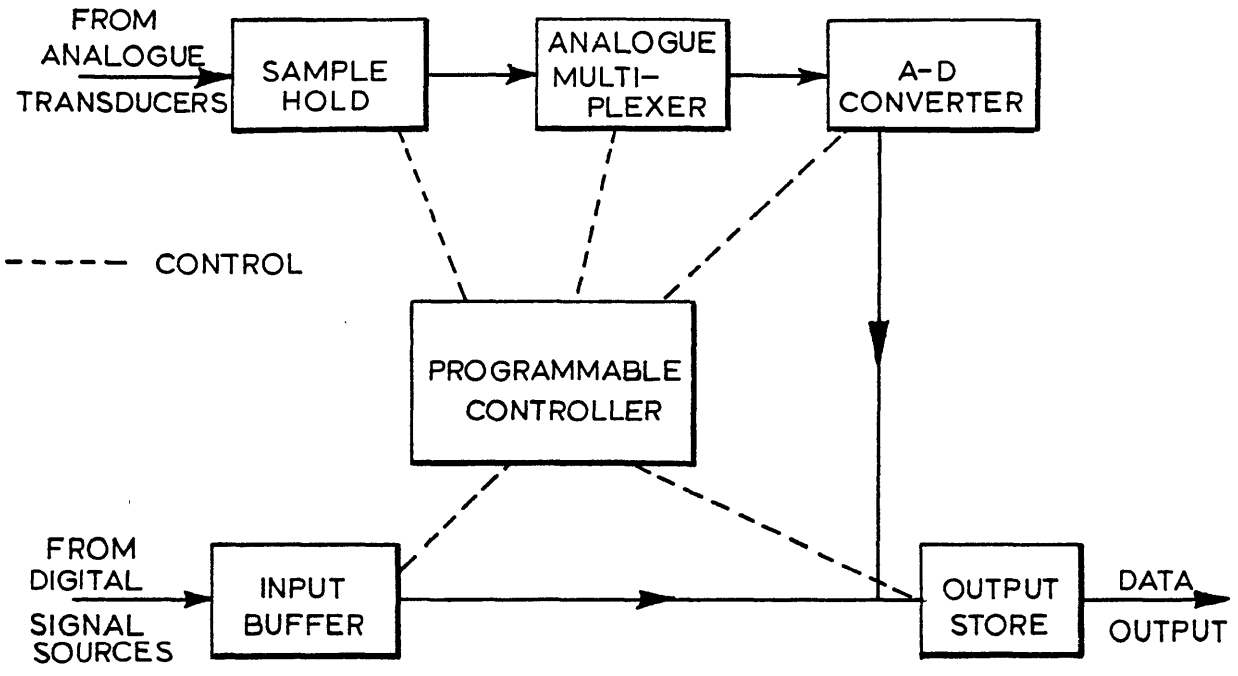


Fig. 2·5 Data acquisition unit outline configuration.

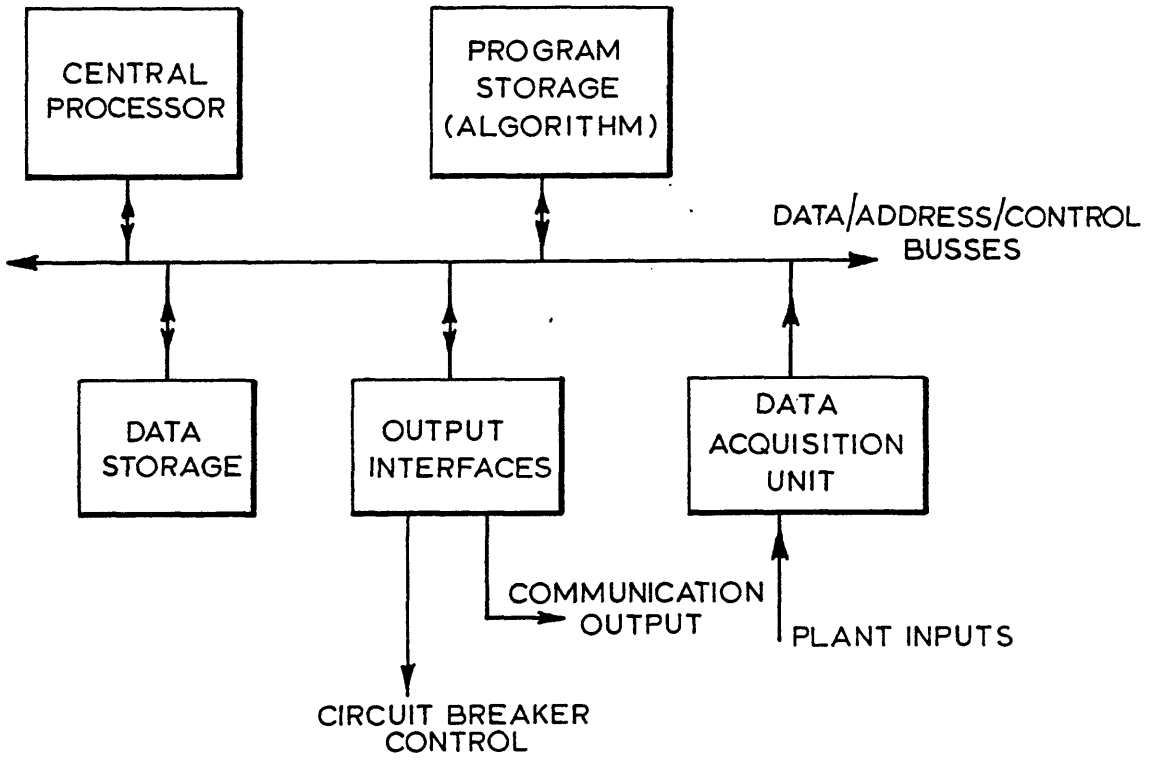


Fig. 2·6 Protection processor outline configuration.

complementary advantages of the dedicated and integrated approaches to produce a comprehensive digital protection scheme. The exact configuration employed for duplex schemes of this type, remains a subject of research. Fig. 2.7 shows a possible arrangement for a sub-station.

The provision of monitoring or back up/monitoring schemes has several important implications for the hardware and software design of the dedicated equipment. In particular, inter-processor data transfer methods require definition. Moreover, a scheme of this type will only be effective if the communication structure and interfaces of the relay equipment are efficient.

The relay design incorporates several of the features required by the combined protection scheme. They are experimentally demonstrated in the application examples, to be described in Chapters 5 and 6.

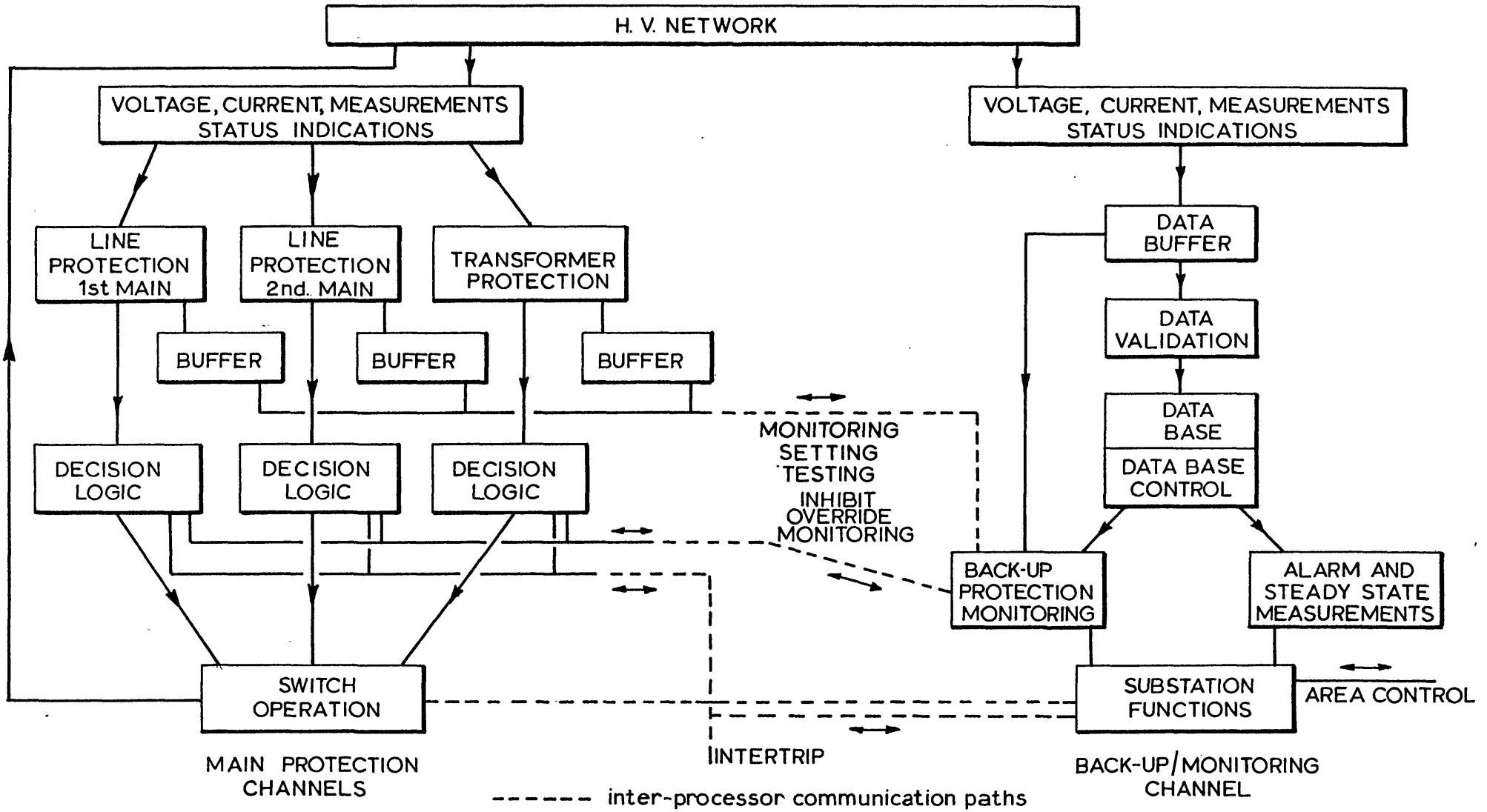


Fig. 2·7 A proposed configuration for the combined integrated/dedicated digital protection of a substation.

CHAPTER 3

THE DATA ACQUISITION INTERFACE

3.1 Introduction

A prerequisite for a digital protection or control scheme is, that the power system quantities used within the computation are measured by an acquisition and conversion interface. Several aspects of the data acquisition task require careful consideration if a satisfactory overall performance is to be achieved. In particular, the final interface structure must provide an adequate accuracy of measurement for the varying applications combined with flexibility to accommodate differing task specifications.

To produce this flexibility a modular approach has been adopted for each component of the system. This allows a wide range of interface requirements to be readily implemented. In dedicated protection relay applications, modular construction methods enable the data acquisition system to be closely matched to each specific task. From an economic viewpoint, this is most important since the interface will certainly represent a large portion of the total equipment cost.

For back-up protection or monitoring schemes, which require large numbers of inputs, the modular structure

offers simple interface expansion.

3.2 Data Sources and Characteristics.

In its minimum form the interface system must be able to serve as an input device for a single dedicated protection relay. Thus a definitive specification for the components of the unit requires examination of the basic quality and quantity of the data sources for several of the possible protection tasks for which a dedicated relay is to be employed.

Only two direct signal sources are of interest in the power network:

(i) the analogue quantities of voltage and current derived from their respective transducers, and,

(ii) the two state logic signals indicating plant status, e.g. circuit breaker and isolator positions; outputs from other protection equipment e.g. intertrip signals; or manually generated test data. In the combined digital protection scheme described in Chapter 2 additional digital data inputs are required by each dedicated relay, to facilitate communication with the supervisory computer.

The extent of the digital data inputs required is difficult to define for a specific task and in view of

their simplicity a maximum of 64 such inputs have been allocated to each interface unit, organised as four 16 bit modules.

Analogue voltage and current data requirements are more readily identified. A study of conventional practice[2,21] indicates the extent of these requirements for three important relaying areas, considered below:-

(i) Overhead feeder protection.

The analogue transducers employed in a typical single circuit, 3 phase overhead line, distance protection scheme, provide i) 3 line current and ii) 3 phase to phase voltage measurements. In some applications an out of balance current derived from a summation transformer is also available. Thus a maximum of 7 inputs are normally required, for which 2 four channel analogue input modules would suffice.

(ii) Differential and earth fault transformer protection.

For complete transformer protection, the unit dedicated digital relay must incorporate overall differential current protection and restricted earth fault protection for both windings. In this case, a maximum of 8 current signal inputs are required by the protection, catered for by 2 four channel modules.

(iii) Bus-zone protection.

The protection of sub-station bus-bar zones provides the most widespread range of input requirements. The data acquisition task for this application will be greatly influenced by the size and configuration of the sub-station.

Conventionally, the bus-zone protection of mesh type stations is organised on a corner by corner basis which results in 4 current measurements being required by the relay per phase. For the more commonly used double-bus station arrangement, the possible measurement sources are very variable. A typical example of an existing station associated with a modern generating plant[24] requires 14 current measurements per phase for bus-zone protection. One or more four channel input modules are thus required for this protection function.

The provision of 16 analogue input channels, organised in 4 channel sampling modules, within a relay interface unit will economically fulfil these data acquisition requirements.

Applications which require in excess of 16 analogue or 64 digital inputs can be accommodated by connecting several units to a common data bus. To facilitate this, a standardised output interface is an essential feature of the data acquisition unit.

Fig.3.1 shows the organisation of the data

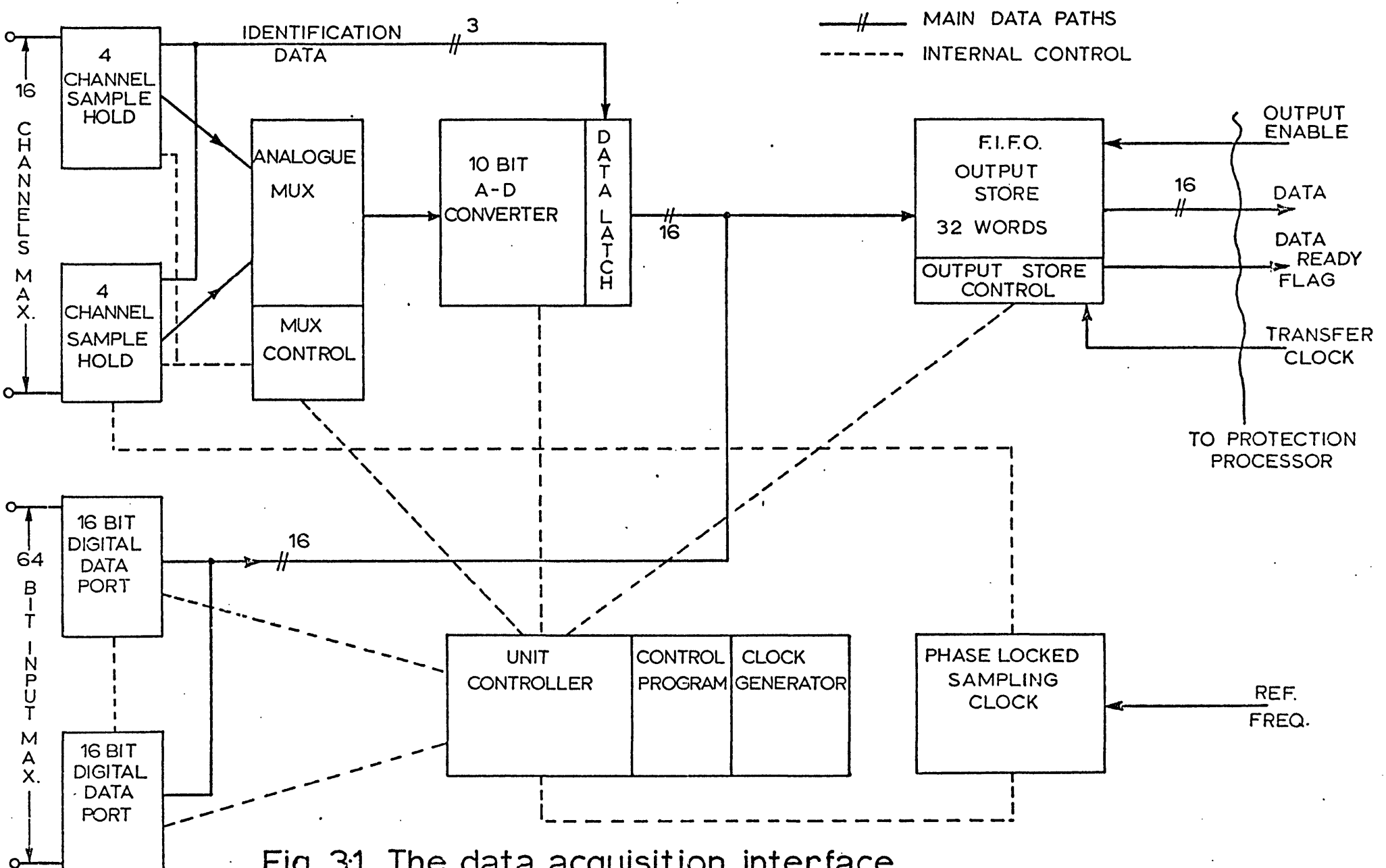


Fig. 3·1 The data acquisition interface.

acquisition unit. The design criteria and characteristics for each module are described fully in the following sections of this chapter.

3.3 Signal Isolation and Conditioning.

The dedicated digital relay design described in this work does not allow for the inclusion of signal isolation and conditioning modules within the data acquisition system of a specific relay. These processes are regarded as forming an additional function which provides a common service to several protective or monitoring equipments.

However, in many respects the isolation and conditioning functions influence the performance of the data acquisition task. A brief review of isolation and conditioning techniques is therefore included here.

The main functions involved may be listed as:

i) Digital signal isolation
ii) Analogue signal isolation
iii) Analogue signal amplitude control
iv) Analogue signal filtering

3.3.1 Signal isolation.

The electrical environment in which protection equipment must operate can be extremely hostile. Broad

band electromagnetic radiation and in severe cases, earth level potential shifts amounting to several kilovolts, can arise from isolator switching, lightning strikes, corona and fault conditions.

It is well appreciated that these conditions will cause noise induced malfunction or possibly catastrophic failure of digital equipment unless precautions are taken to isolate signal input and output lines. Considerable attention must also be paid to preventing interference entering the equipment via its own power supplies. Additionally elevated earth potentials represent a hazard to personnel involved in maintaining the protective equipment.

Recently the application of mini-computers and other digital equipment for monitoring, switching and experimental purposes in sub-stations, has provided an incentive for the development of suitable isolation techniques. Lomas[25], describes several important aspects of signal line and power supply isolation methods. Optical isolation of digital signal inputs is now a well proven technique and the use of multi-core light pipe "cables" is a highly attractive method of providing wide bandwidth noise immune inter-computer communication links.

For analogue signal inputs, the use of twisted pair lines, double screened isolation transformers and limiting filters have been reported[26,27] as providing

satisfactory protection.

Although the interference problems in the protection environment are severe, they can be overcome by careful design of both individual digital relays and the complete protection installation.

3.3.2 Analogue signal ranging.

Analogue signals derived from the power system transducers may at times be subject to a wide variation in magnitude. This is especially true of current measurements for which large fault current-normal load current ratios are possible.

The internal signal levels of the interface system are limited to an operating range of $\pm$ 10V. In many applications incoming signals will require scaling. For low level inputs amplification must be provided to maintain analogue-digital conversion accuracy (Section 3.6.2). Equally during peak conditions, saturation of the equipment, with a consequent loss of information, must be prevented. This is of great importance in protection applications, since saturation is most likely to occur during fault conditions when data acquisition is at a premium.

Three possible methods of signal amplitude control have been investigated and evaluated by the author:

a) Logrithmic amplifiers.

The use of amplifiers with a logarithmic input-output gain characteristic as shown in Fig.3.2, enables a

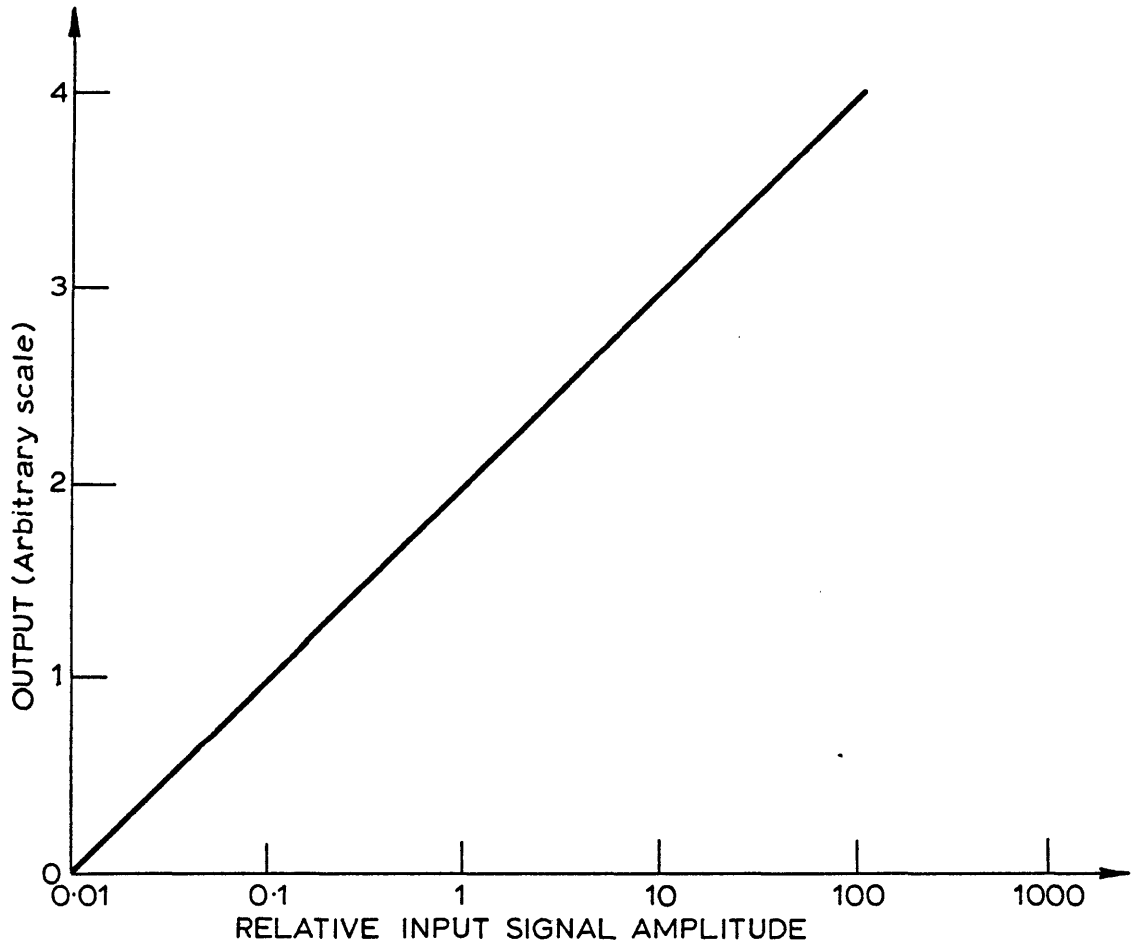


Fig. 3·2 Logarithmic amplifier gain characteristic

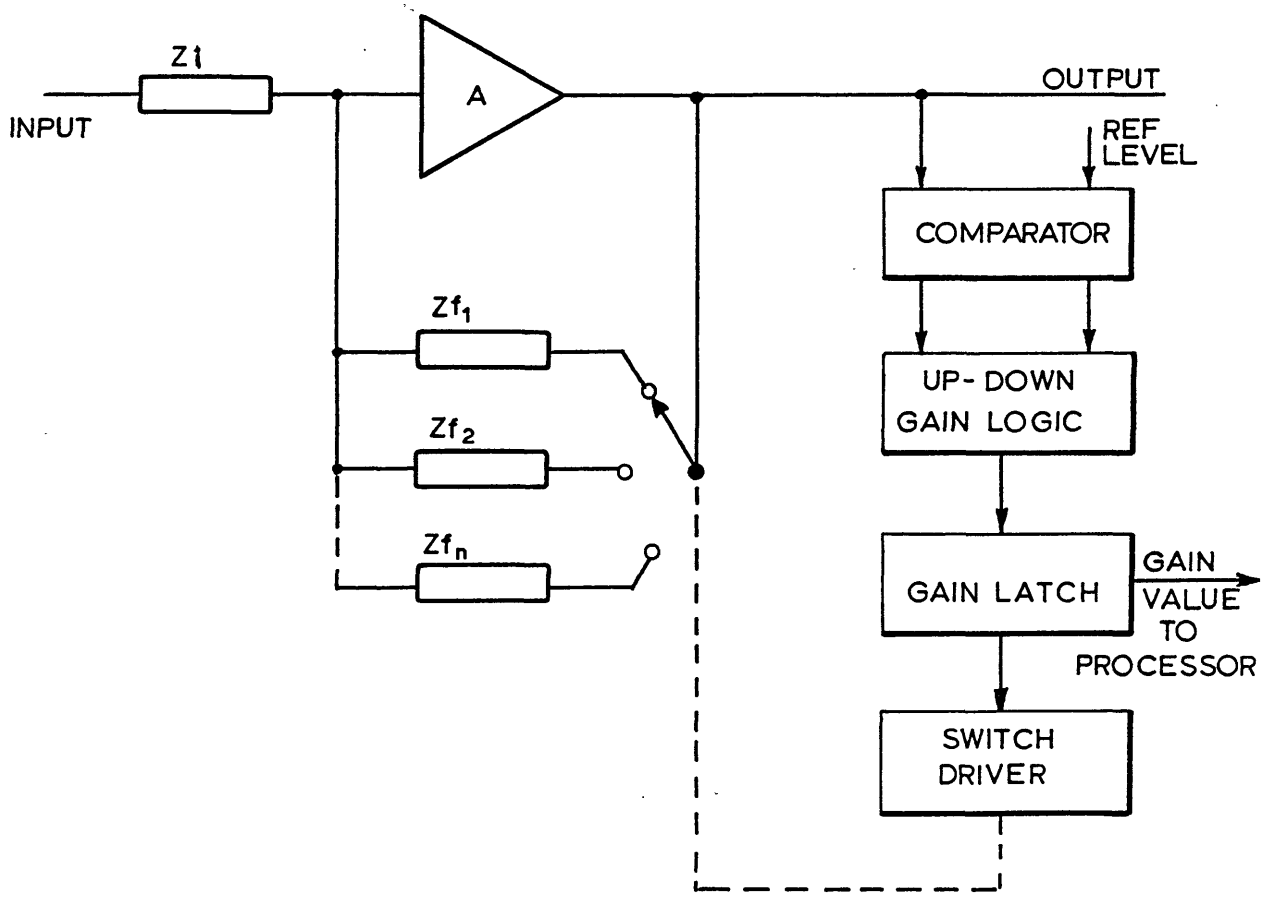


Fig. 3·3 Gain switching amplifier

wide signal amplitude control to be achieved. However, since the measured data becomes a logrithmic function of the true data, this approach was found to produce considerable additional complication in subsequent processing.

b) Gain switching.

In this method the feedback impedances of a linear operational amplifier are varied in discrete increments by control signals generated in response to level detection of the amplifier output, Fig.3.3. By adjustment of the impedances Zfi---Zfn it is possible to provide a corresponding range of gains $\frac{Zfi}{Zi}$ --- $\frac{Zfn}{Zi}$ to accommodate a wide range of input signals. If these gains are spaced at increments of 2^n the reduction of the measured data to its true value is easily achieved in later processing by n-bit left or right shifting. The gain switch control logic provides the "n" value output which is made available to the protection processor by the interface system.

c) Range switching.

The gain switching scheme described above suffers from prolonged (5-20μs) settling times following feedback alteration. This undesirable feature may be overcome by multi-amplifier range switching.

For each gain range this method employs a separate amplifier with a fixed feed-back network, shown in Fig.3.4. Each amplifier output is monitored by positive and negative voltage comparators set to operate at 80% of the maximum

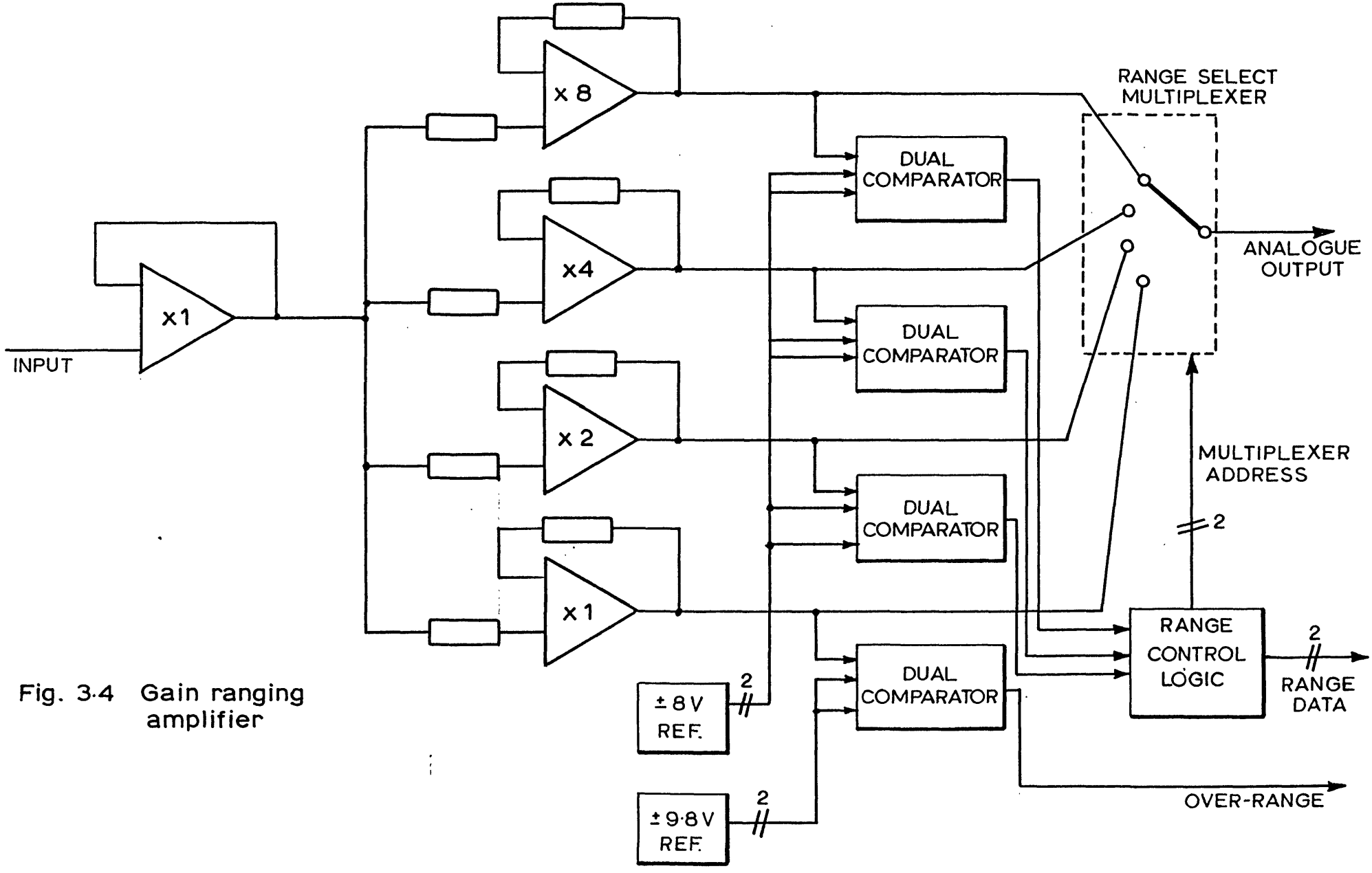


Fig. 3.4 Gain ranging amplifier

permissable amplitude, i.e. ± 8V. The comparator outputs provide data to control logic which in turn drives the output range selection switch to the highest gain channel operating within the 80% margin. This logic also generates the corresponding binary range identification data for use in the protection processor.

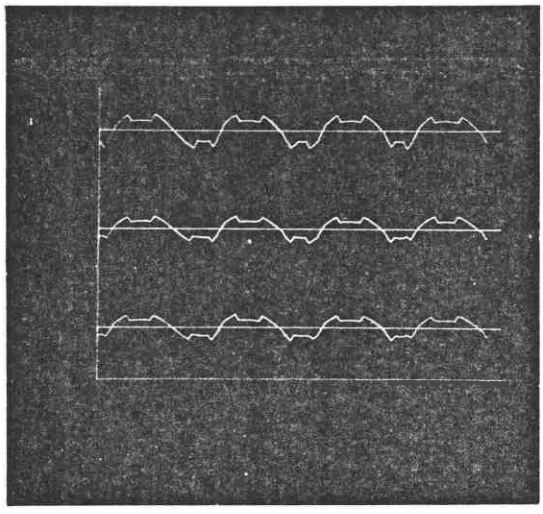
The lowest gain channel output is monitored by a comparator which operates when the output of this channel exceeds 98% of the maximum level, i.e. ± 9.8V. An indication is thereby provided to the protection processor of incipient saturation of the data source concerned.

Again for simplicity of processing the gain ranges are set at intervals of 2^n, although no restrictions exist in the choice of range steps. Range changes are achieved in 1.5μs with negligible overshoot or settling effects. The oscillograms of Fig.3.5 show the operation of the amplifier and subsequent reconstitution of the waveform by the protection processor.

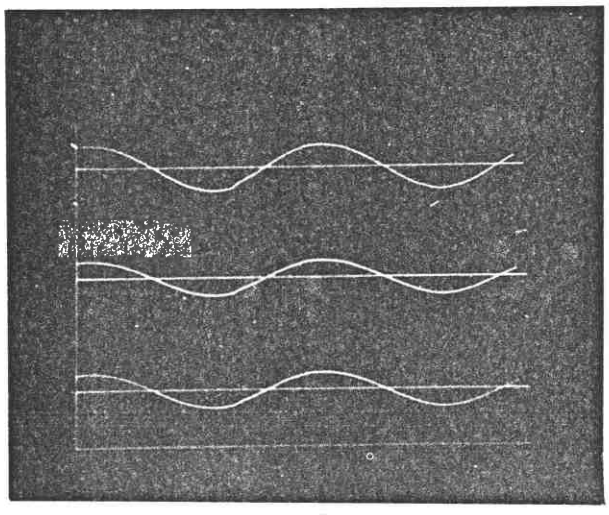
Range switching amplifiers of this type are recommended for signal amplitude control.

3.3.3 Analogue signal filtering.

The final stage of signal conditioning prior to measurement involves filtering. For each application and signal source, varying filter requirements exist. Ranjbar[9], provides a comprehensive study of the effects of analogue



(a)



(b)

Fig. 3·5 Operation of the gain ranging amplifier

(a) Amplifier output

(b) Waveform reconstructed by the protection processor

signal filtering on data processing in a protection function and concludes that for most applications a second order Butterworth filter with an appropriate cut-off frequency is adequate. It is essential that high frequency components in the input signal are removed by the filter, otherwise aliasing[28] will occur during the sampling process to the detriment of the measurement accuracy.

3.4 The Sample-Holds

3.4.1 The need for sample-hold devices.

Following isolation and ranging, analogue signal inputs to the interface are sampled at regular intervals and stored to await further processing. These functions are performed by the sample hold (S-H.) modules.

In most applications the analogue inputs will be alternating quantities with a frequency corresponding to that of the power system i.e. 50-60Hz. Conversion of these inputs to a digital form takes a finite time and the extent by which the input changes during this time may cause significant errors in the conversion process[10]. This may best be illustrated by considering the maximum frequency of input signal which can be tolerated by the converter used in the interface.

For an "n" bit analogue - digital converter having a maximum input range of Vfs volts, the resolution, Vr, is at best $\pm\frac{1}{2}$ least significant bit (L.S.B). i.e.

$$Vr = \pm \tfrac{1}{2}\left(\frac{Vfs}{2^n}\right) \quad \text{volts} \qquad (3.4.1)$$

Thus if the input signal changes by more than Vr volts during the conversion period, Δt, the conversion process will include an additional error greater than $\pm \frac{1}{2}$ L.S.B. The maximum acceptable rate of input change is then $\frac{Vr}{\Delta t}$ volts/sec. As described in section (3.6) the A-D converter used in the interface unit provides a 10 bit two's complement output with a conversion time of 6.4μs., and Vfs of $\pm$ 10 volts.

For this device then:-

$$Vr = \pm \tfrac{1}{2} \left(^{10}/_{2^9}\right) = 9.76\text{mV} \qquad (3.4.2)$$

and the corresponding rate of change:-

$$\frac{Vr}{\Delta t} = \frac{9.76 \times 10^{3}}{6.4 \times 10^{-6}} \text{ volts/sec} = 1600 \text{ V/s} \qquad (3.4.3)$$

For a sinusoidal input signal with peak values equal to the maximum interface system internal range of $\pm$ 10v, the rate of change at the zero crossing maxima is:-

$$\left.\frac{dV}{dt}\right|_{max} = 10\ (2\pi f) \qquad (3.4.4)$$

giving f, the frequency of the signal as:-

$$f = \frac{1}{20\pi} \frac{dV}{dt} \qquad (3.4.5)$$

Therefore, substituting the maximum acceptable rate of

change $\frac{dV}{dt}$ = 1600 V/s from 3.4.3

gives a frequency limit for conversion within $\pm$ ½ L.S.B, f max. of:-

$$f\ max. = \frac{1}{20\pi} \times 1600 = 25.5Hz \qquad (3.4.6)$$

This converter is therefore unable to provide its full accuracy for alternating inputs with frequencies in excess of 25 Hz. At power system frequencies the 10 bit resolution would be totally wasted.

By providing the converter with a static input during the conversion time from the S-H modules, this difficulty is overcome. An alternative method of solution involves the use of a converter with a reduced operation time. However, this is again generally unsatisfactory for a multiple signal source system. The period which elapses between the conversion of the first and last input channel data may result in later processing errors if these quantities are taken to be representative of a single time instant, for example, current and voltage in an impedance computation.

The severity of this error obviously increases as the number of inputs requiring conversion increases. A solution is only possible by the use of multiple converters, which is economically unacceptable, or by the provision of S-H buffers for each input.

3.4.2 Sample-hold characteristics.

Fig. 3.6 represents the basic configuration of a S-H. When the switch closes the capacitor, C, charges to the input voltage, Vs, and the charge remaining on the capacitor provides the held output when the switch opens.

The circuit of Fig. 3.6 suffers from severe limitations in many of the major characteristics which are essential for the application of a S-H. These characteristics may be listed as:-

i) Acquisition time. This is the period for which the switch must be closed to allow the capacitor voltage to rise to within a given accuracy of the input. It is an expontial function of the switch characteristics, particularly the "ON" state impedance, the value of the capacitor and the characteristics of the source to which the S-H is connected.

The inclusion of an input buffer amplifier, A1, Fig. 3.7 ensures that the capacitor charging source impedance is low and that adequate charging current is available. In the interface system this amplifier is not included within the S-H modules, but may be considered to exist as the output stage of a preceding active signal filter or gain ranging amplifier. The acquisition time of the modular S-H units[29] is 35μs which allows a satisfactory droop rate to be achieved without excessive economic penalties.

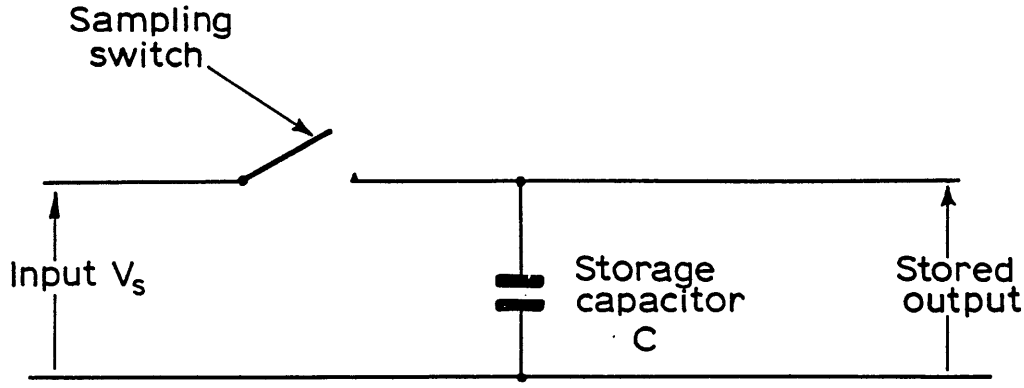


Fig. 3·6 Basic configuration of a sample-hold device

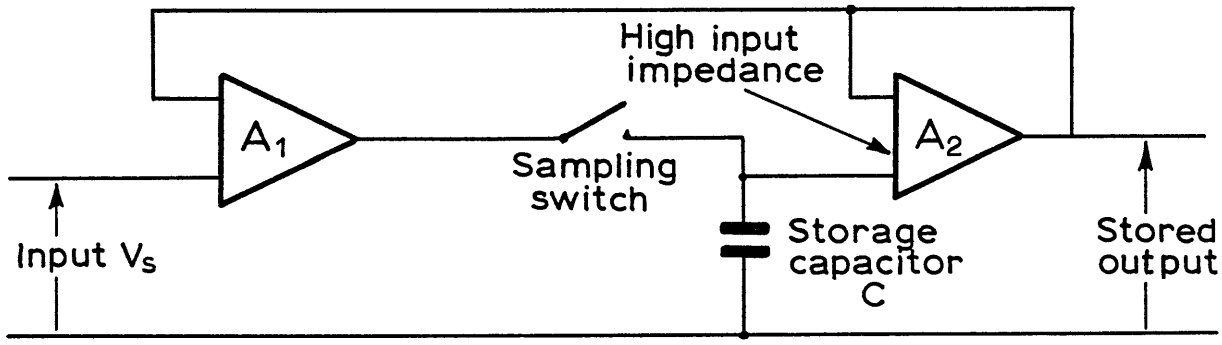


Fig. 3·7 Improved sample-hold with input and output buffer amplifiers

ii) Droop rate. When the sampling switch opens the stored capacitor voltage is subject to change, caused by leakage, both internally and through the "OFF" state switch resistance, and to a very large degree by the characteristics of the load to which the S-H is connected. The output amplifier A2, Fig. 3.7 , is included to ensure that the capacitor, C, is always provided with a high impedance load, and that the output impedance of the S-H module, which must be considered at a later stage of the system (section 3.5) is low.

The droop rate of the S-H modules must be defined such that; during a maximum multi-channel conversion, the value of the last channel to undergo conversion will not have altered by more than $\pm$ ½ L.S.B. of the conversion accuracy within the total conversion time of the preceeding channels.

For the interface system with a maximum of 16 input channels and a 10 bit conversion rate of 14μs per channel the total hold time required will be approximately 234μs including the possibility of digital data input. $\pm$ ½ L.S.B. for the 10 bit converter is $\pm$ 9.76 mV and therefore the maximum acceptable droop rate is 9.76mV/234μs = 41 V/s. The devices used in the modules have a droop rate of 2V/s at ambient temperature (25^{o}C) which is substantially less than the tolerable maximum. This characteristic is, however, highly temperature dependent and a doubling of the

rate for every 10°C rise in temperature must be allowed for[30].

iii) Gain, offset, aperture time. These parameters are of secondary importance since the overall gain and offset characteristics of the sample-hold are closely controlled and can be externally compensated to give unity gain, with offset values at least an order of magnitude less than the converter resolution.

Aperture time is a measure of the uncertainty in the acquisition time due to the dynamic characteristics of the sampling switch. For the devices used in the interface system the aperture time is specified as less than 100ns. Errors arising from this source will increase with increasing signal frequency.

Since the interface unit has a maximum sampling rate of 24 times per 50 Hz cycle, the highest frequency component for which the Nyquist criterion holds is: 12 x 50 Hz = 600 Hz. Frequencies in excess of this value must be removed from the input signal by filtering, prior to sampling.

If the input signal consists solely of a $\pm$ 10V, 600 Hz sine wave, the error caused by the 100ns aperture time will be a maximum for samples taken at the zero crossing points. At these points the rate of change of the signal is greatest and is given by:

$$\left|\frac{dV}{dt}\right|_{max} = 20\pi f \qquad (3.4.7)$$
$$= 20\pi \times 600 \text{ V/s}$$

The error voltage Δv in the sample is thus:-

$$\Delta v = \frac{dV}{dt} \Delta t \qquad (3.4.8)$$

where Δt is the aperture time.

Hence:

$$\Delta v = 20\pi \times 600 \times 10^{-7} V$$

which as a percentage of the full scale voltage, (10V) is:

$$\Delta v = \frac{20\pi \times 600 \times 10^{-7} \times 100}{10} \%$$

$$\Delta v = 0.038\%$$

At lower frequencies the error becomes negligible for example at 50 Hz, $\Delta v = 0.0031\%$

3.4.3 Sampling module operation modes

Two alternative modes of operation of the sampling modules are possible. In the sample-hold mode the input waveform is sampled for a period of 35μs at every sampling instant, as defined by the sampling clock (section 3.9), and stored during the conversion process until the following sampling instant. The sequence is illustrated in Fig. 3.8.

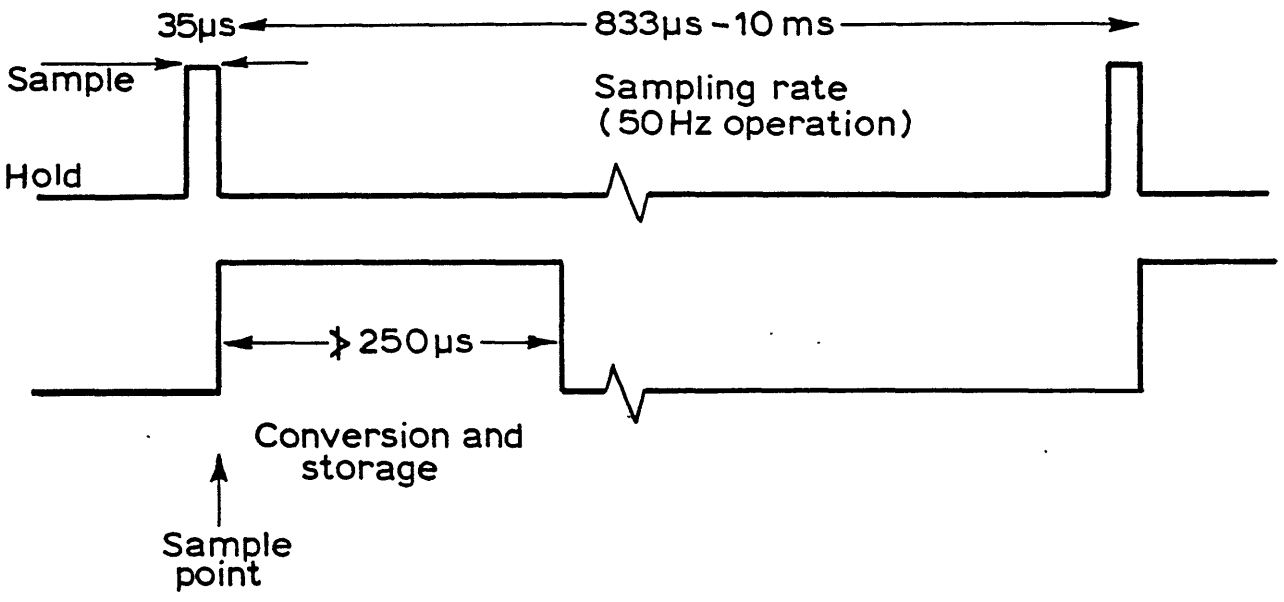


Fig. 3·8 Operation of sampling modules in the sample-hold mode.

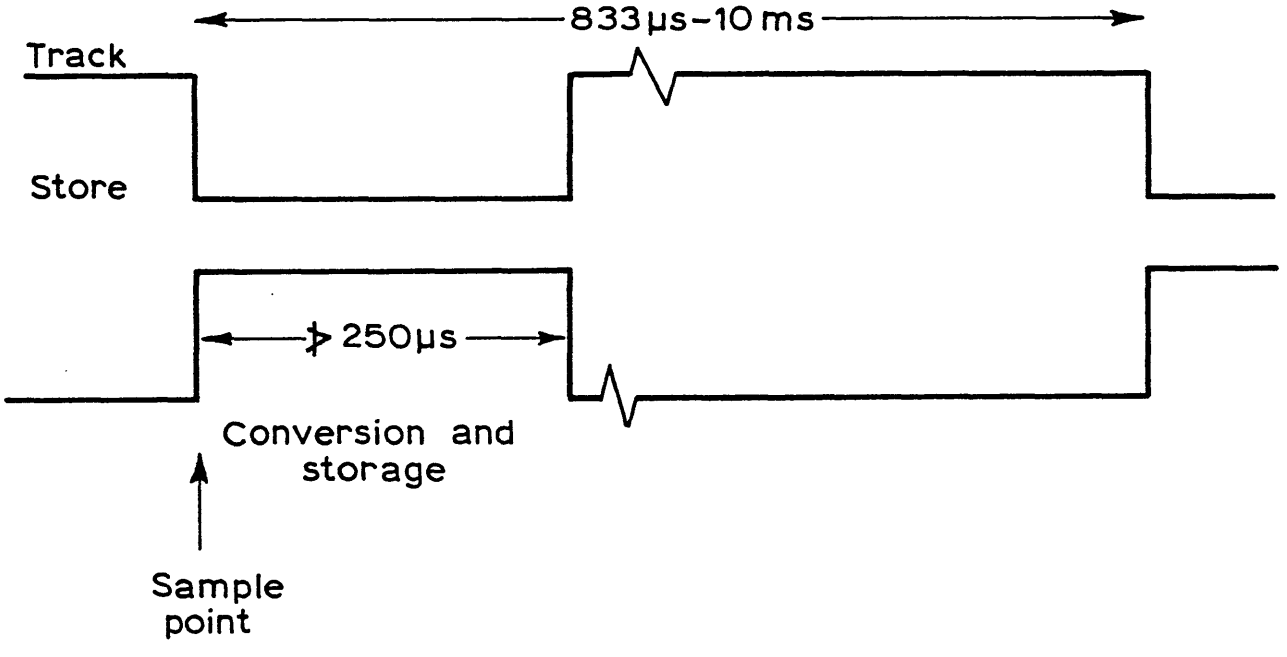


Fig. 3·9 Operation of sampling modules in the track-store mode.

For power system applications the sampling instants typically occur from 4 to 24 times per primary system cycle. At points of maximum rate of change in the case of peak value (i.e. $\pm$ 10V) inputs, these sampling instants will produce substantial step changes within the S-H's which aggrevate error sources due to amplifier overshoot and settling times.

It is possible to minimise the occurence of step change conditions by operating the sampling modules in a "track-store" mode.

In this mode the input signal is continuously tracked between sampling points, i.e. the sampling switch is closed. At each sampling point the switch is opened thereby storing the data value at that point until the end of the conversion process, when the track mode is again resumed. Since the worst case conversion period (24 samples/cycle, 16 input channels) is approximately 30% of the interval between samples, the sampling modules are in the track state for the remaining 70% of the interval, in this example 600μs, which allows ample time for transient effects to decay. Fig. 3.9 shows the timing sequence for the track-store mode. The sampling clock design facilitates either mode of operation.

3.4.4 Digital data storage

Each sampling module within the interface system comprises four analogue signal channels. Associated with each of these channels is a three bit digital storage latch. This digital word storage is provided to enable acquisition of additional information relating to the corresponding signal channel, e.g. identification or signal ranging amplifier gain data, (section 3.3.2). A data word presented to the latch inputs is stored upon termination of the sample/track signal, i.e. the point at which the analogue channels enter the store mode. During the subsequent conversion of each analogue data the corresponding digital data is multiplexed and combined with the converter word output prior to storage in the interface unit output buffer.

Fig. 3.10 is a block diagram of the four channel sampling module.

3.5 The Analogue Multiplexer

3.5.1 Analogue signal switching

The third module in the signal path of Fig.3.1 is the analogue multiplexer. (MUX). This device provides signal selection by successively connecting the A-D converter input to each of the sample hold outputs.

Analogue multiplexers to perform this task are

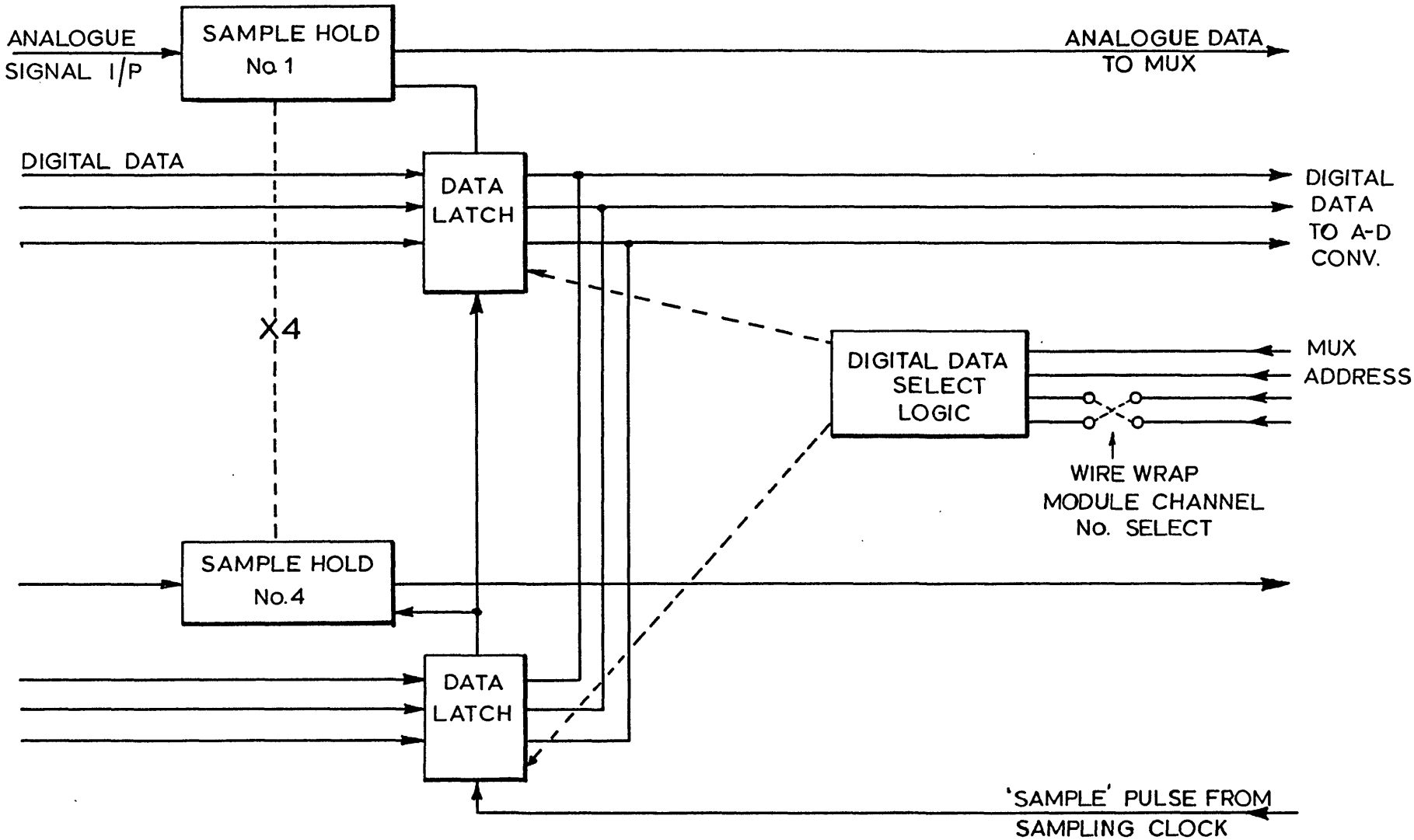


Fig. 3·10 Sample-hold module.

frequently constructed using miniature reed relays or discrete field-effect transistors as the switching elements. They are also readily available as integrated devices with commonly 8 or 16 channels and standard logic level control inputs.

This latter approach[31] was adopted for the interface system. However, several parameters of this basically simple device require careful consideration as they have an important bearing on the overall performance of the equipment. The major factors involved may be listed as:-

Signal range: the switching elements of the MUX must be capable of handling the maximum range of the analogue signals present. In this respect the relay has an ultimate performance which is superior to the solid state types presently available. However, the internal signals of the interface system are limited to $\pm$ 10 volts and the $\pm$ 15 volts range of the integrated device used is sufficient.

Static transfer accuracy: the static or DC transfer accuracy of transmitting the MUX input voltage to the output depends on the switching element "ON" state resistance (R_{on}), Fig. 3.11, the source and load resistances between which the MUX is connected, (R_S) and (R_L) respectively, the load bias current (I_b) and the leakage current from the "OFF", channels (I_ℓ). Fig. 3.11 is a generalised equivalent circuit for these error sources which is applicable to any type of switching device. From this figure it is evident

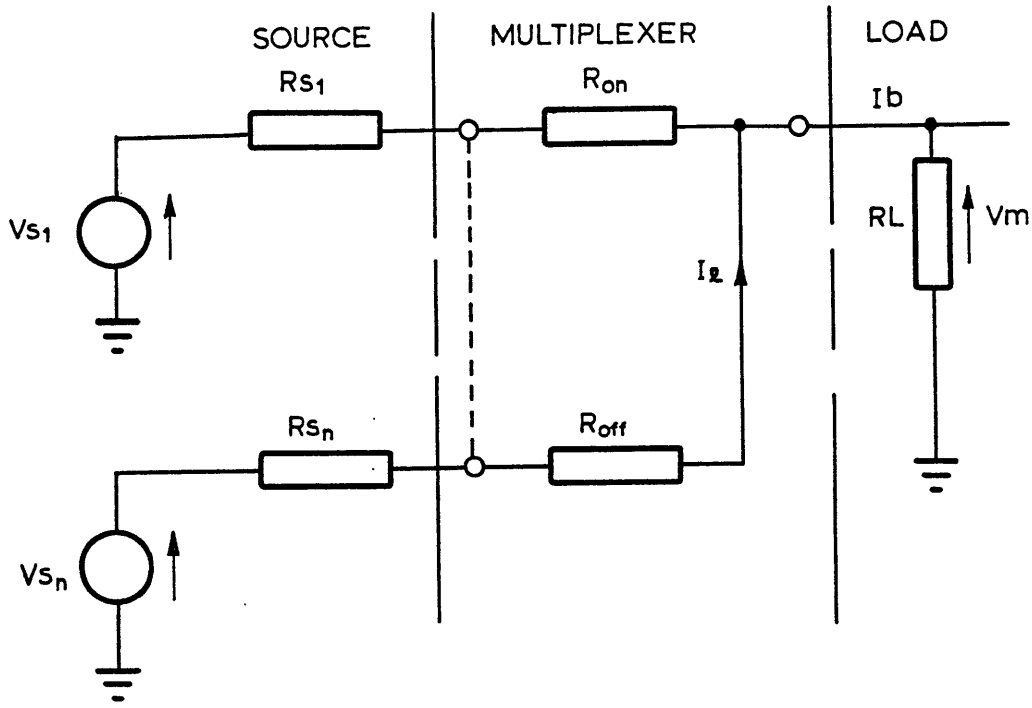


Fig. 3·11 Equivalent circuit for multiplexer static transfer error sources.

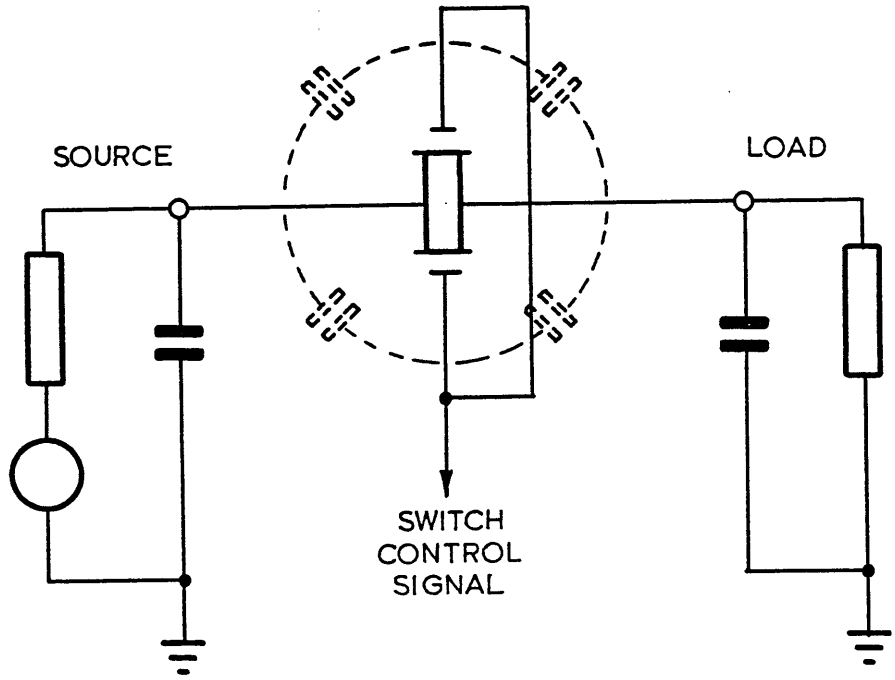


Fig. 3·12 Semiconductor switch settling time effects.

that the error introduced by the MUX is given by:-

$$E = \frac{R_S + R_{ON}}{R_S + R_{ON} + R_L} \times 100\% \qquad (3.5.1)$$

In the interface system the output impedance of the sample-hold units (R_S) is very low, in the order of 0.1Ω and can therefore be neglected, whilst the input impedance of the A-D converter (R_L) is $4.7 \times 10^6\Omega$, R_{ON} for the MUX is $1.8 \times 10^3\Omega$ in the worst case. From 3.5.1 these values give a transfer error of 0.04%. The offset voltage,

$$V \text{ offset} = (I_b + I_\ell)(R_S + R_{ON}) \qquad (3.5.2)$$

can be neglected since the combined currents I_b and I_ℓ are less than 80 nA, giving an error contribution of 0.0014%.

Dynamic characteristics: Several parameters describe the dynamic performance of the multiplexer. The first, and most important, is the operating time of the switching device used. The interface system application requires that, in the worst case, the MUX must perform 16 operations in approximately 800 μs. Semiconductor switches are therefore essential, although their inherent internal capacitance, Fig. 3.12 gives rise to switching transients. To maintain transfer accuracy, the settling time which must elapse while these effects decay considerably lengthens the total operation period. The integrated device used has a switching time of 0.5μs and a settling time of 2μs to 0.1%.

Finally, to ensure that signal sources connected

to the MUX inputs are not shorted together, the switching must occur in a "break before make" sequence, i.e. the turn-off time of the switching devices must be less than the turn-on time.

3.5.2 Multiplexer control logic

To control the MUX operation, it must be supplied with a binary "channel select" address which is generated by the control logic. The addressing is sequential and provided by a binary counter.

A MUX cycle consists of loading the control counter with the "first channel" switch data, after which it is successively incremented by commands from the interface controller. When the current address corresponds to that of the "final channel" switches the equivalence is detected by the comparator, and upon receipt of the next "increment" pulse the control counter reloads the "first channel" data and the cycle repeats.

To simplify the task of the controller the comparator output is provided as a "cycle complete" flag (MUX FLAG). This obviates the necessity for the controller to maintain a channel counter, which would require a programming alteration whenever a change was made to the number of analogue inputs required.

The address counter output is also used by the sample-hold modules to select the digital identification

data corresponding to the current analogue channel.

Fig. 3.13 is a block diagram of the multiplexer module.

3.6 The Analogue-Digital Converter

3.6.1 The conversion function

The modules and components of the interface system which have been discussed in the preceding sections are primarily concerned with the processing of the analogue quantities derived from the power system transducers. Before these quantities can be transferred to the digital system which the interface serves, it is necessary that an analogue to digital conversion is performed, by which each analogue value may be represented as a corresponding digital word.

This task is fulfilled by the analogue-digital (A-D) converter module, and the characteristics of this unit are the most important single factor within the interface. The module also represents a major portion of the system cost.

3.6.2 Converter performance requirements

Several frequently used techniques exist for converting an analogue value to its digital counterpart, and detailed appraisals of the relative merits for the various methods are fully documented by several sources[32,33].

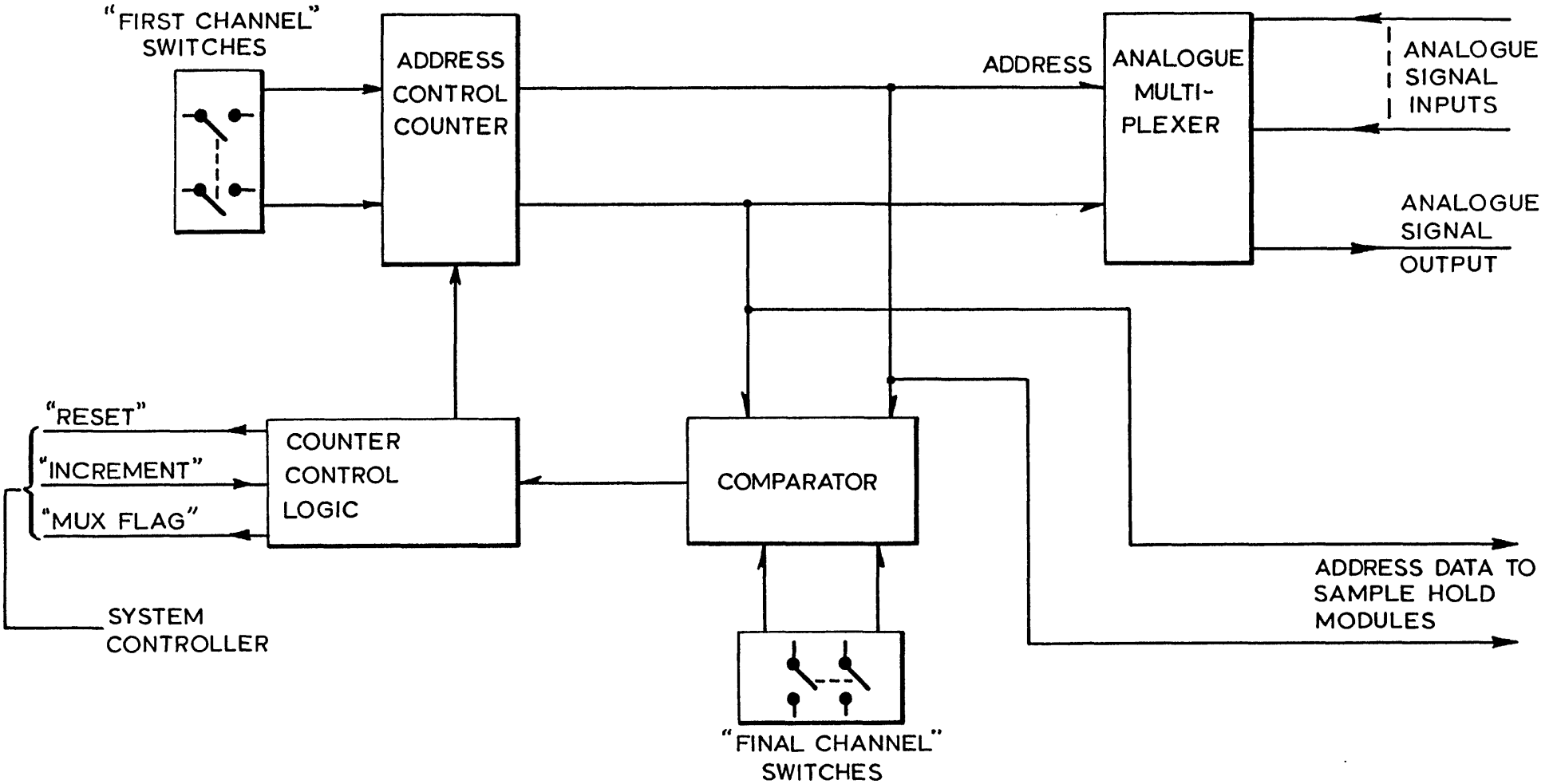


Fig. 3·13 Analogue multiplexer module

For the interface system application the important criteria are the time taken to perform a conversion, and the overall accuracy of the conversion, in which the number of bits used to represent the converted value is a factor.

i) Conversion time.

In the extreme case, which occurs at the highest data sampling rate, the interface must be able to convert and store a maximum of 16 analogue data values, and 4 direct digital input words between sampling instants. An allowance must also be made for the transfer of the data block to the protection processor.

These considerations indicate a limiting value of approximately 45 μs per conversion at 24 samples per 50Hz power system cycle, when allowance is made for the internal interface system control and data transfer operations. A further 5 μs must be allowed for analogue device settling times per conversion which implies that the A-D converter must operate in less than 40 μs.

This requirement precludes the use of several conversion techniques, but is easily fulfilled by devices using successive approximation methods, for which conversion times in the order of 10 μs to accuracies of 12 bits or less are readily available.

By employing a converter which provides a substantial speed margin, timing difficulties which may arise in

multiple input schemes involving the possibility of data validation are minimised. Savings in overall acquisition time may be used for pre-processing between sampling instants in these applications.

ii) Conversion accuracy.

The representation of an analogue value by a binary word will in most instances give rise to quantisation errors caused by the finite word length of the representation. Quantisation errors result from the truncation or rounding of a true value when accommodated in a given number of binary digits. Both error sources are non-linear in nature and, in truncation, are dependant on the method employed by the digital system to represent negative quantities.

The errors may be regarded as introducing an additional noise source to the data acquisition process. An analysis[34] of the influence of various conversion word lengths on protection algorithms using sinusoidal peak determination methods, concludes that unless the input quantities are maintained at levels close to the rated converter maxima, the quantisation errors will become excessive when 13 + sign, or less bits are used.

In designing the interface, however, several factors must be considered before a decision can be reached as to the quantisation word length of the A-D converter module.

For an n bit converter word length the limiting

accuracy with which an input quantity may be represented is $\pm$ ½ L.S.B.. The error Eq, thus attributable to the process of choosing the closest quantisation level is given by:

$$-\tfrac{1}{2}\,2^{-n} < Eq \leq \tfrac{1}{2}2^{-n} \qquad (3.6.1)$$

assuming that a value lying exactly halfway between two steps is always rounded up. For an n bit converter producing a two's complement output code and a full scale voltage input range of $\pm$ Vfs this error is:

$$Eq = \pm\tfrac{1}{2}\,\frac{Vfs}{(2^{(n-1)})Vi} \times 100\% \qquad (3.6.2)$$

Where Vi is the input voltage. Eq is thus a minimum when Vfs = Vi, thus:

$$Eq_{min} = \pm\,\frac{100\%}{2^n} \qquad (3.6.3)$$

Eq_{min} for 10, 12 and 14 bit converters is thus:

word length	Eq_{min} %
10 bits	0.097
12 bits	0.024
14 bits	0.006

To evaluate the significance of these errors, the other error sources to which analogue data is subject must be assessed, they are:-

i) Sampling module - sample to hold error ± 0.01%

ii) Static transfer error of multiplexer - 0.04%

iii) Settling time error of multiplexer ± 0. 1%

iv) Settling time error of converter buffer ± 0.01%

v) Converter linearity errors, reference voltage errors, and supply voltage drift ± 0.0122%

The mean error from these sources Em is thus:

$$+Em = -0.04 + \sqrt{0.01^2+0.1^2+0.01^2+0.0122^2} = +0.06\%$$

and

$$-Em = -0.04 - \sqrt{0.01^2+0.1^2+0.01^2+0.0122^2} = -0.14\%$$

These errors are thus of the same order as the quantisation error for a 10 bit converter and substantially larger than those of 12 or 14 bit devices.

The inclusion of the quantisation error for a 10 bit converter results in a total error of measurement for varying inputs as shown below:

Input voltage	Total error %
10v	+0.1 - 0.18
5v	+0.18 - 0.26
1v	+0.93 - 1.01

For 12 and 14 bit converters the total error including that due to quantisation is considerably less, rising to a maximum of -0.31% at a one volt input level for the 12 bit

device and -0.15% for 14 bits.

These errors relate solely to components within the interface. Even for the 10 bit conversion word length, they compare favourably with the basic transducer accuracies[35] and errors arising from signal isolation equipment and active analogue filters, which may reasonably be expected to have no better than a 1% tolerance.

Therefore to justify the use of an A-D converter with an output in excess of 10 bits, in the complete system, would require a much improved performance of the signal path components which precede it. The economic constraints of providing high stability analogue equipment with accuracies better than 1% would greatly detract from the advantages of dedicated digital protection equipment.

For these reasons a 10 bit converter is employed in the interface. By ensuring that signal levels are maintained, when required, by means of range switching amplifiers, the contribution of the quantisation error for this converter, to the total measurement process error, can be reduced to acceptable limits for protection tasks, for which primary transducer accuracies of 5-10% are normally specified.

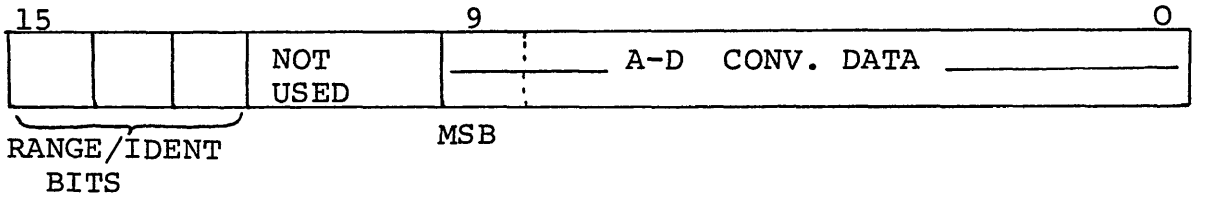
3.6.3 Analogue-digital converter module

Design of this module is based on a hybrid, successive approximation, 10 bit A-D converter[36], with a conversion time of 6.4μs. A buffer amplifier provided

within the device is employed to raise the input impedance of the module, thereby reducing multiplexer transfer errors. An additional 2μs allowance for the settling time of this amplifier results in a total conversion cycle of 8.4μs. The block diagram, Fig. 3.14, shows the control logic and storage associated with the converter.

A conversion cycle is initiated by the application of a start pulse (ADCV PLS), from the system controller, and the process continues under the control of the converter internal clock. An "end of conversion" (EOC) signal, generated by the device upon completion of a conversion, is used to load the two's complement output into the single word storage register. At the same time the converter ready flag (ADCV FLG) is set to provide a cycle termination indicator for the system controller.

The storage register allows the conversion and data transfer (to the interface system output buffer) operations to be overlapped, which reduces the conversion cycle timing by 18%. Converter output data and corresponding digital identification data, if any, from the sampling modules (section 3.4.4) are combined in the register to form a complete 16 bit word shown below:-



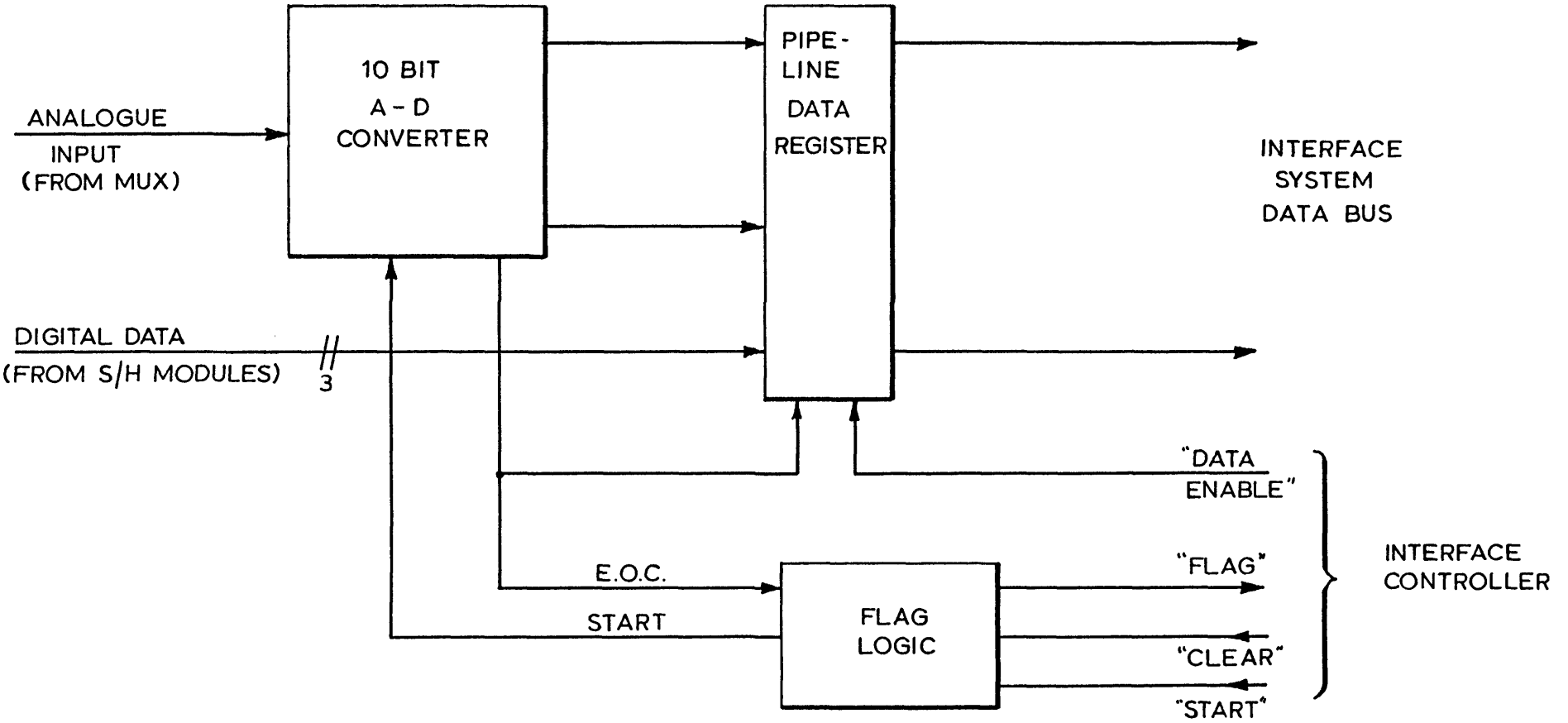


Fig. 3·14 A-D converter module.

3.7 Digital Input Ports

3.7.1 Digital data sources

In many relaying applications the requirement for digital data acquisition is apparent. For example, in bus zone protection schemes circuit breaker and isolator status information is required to enable zone determination to be carried out by the relay, and for an integrated back up protection configuration the breaker status data input forms an important part of the data validation process.

Further sources of digital data which may be required by the protection processor for specific tasks arise from the output signals of other protective equipment. Buchholz gas relays are a typical example in the case of transformer protection, whilst intertrip signals must be accommodated in the majority of unit protection applications. Additionally, in the combined integrated-dedicated digital protection scheme described in Chapter 2, each dedicated relay requires bi-directional communication channels with the supervisory-back up protection processor.

By entering data from the back up processor via a digital input port of the dedicated relay data acquisition interface, the range of supervisory control available is greatly extended. With data entry at this point, the testing of the data bus, storage functions, sampling clock

and interconnection links of the interface is directly possible, while still allowing the data entered to provide program modification commands in the protection processor.

3.7.2 Digital input modules.

Each module comprises a 16 bit wide parallel bus driver, and forms an input port for data at standard transistor-transistor logic positive logic levels i.e. $0 <$ Logical '0' ≤ 0.8, $2v \leq$ Logical '1' $\leq 5v$.

Provision for a maximum of four such ports is made in a single interface unit giving a total input capacity of 64 bits. No intermediate data storage is provided on the modules, and data present on the module input is transferred directly to the output buffer store by command pulses generated by the programmable interface system controller.

3.8 Data Output Buffer Store

3.8.1 The buffer function

At each sampling instant a set of data comprising converted analogue quantities and direct digital inputs is generated intermittently over a short period of time by the data acquisition system.

To simplify, the output interface of the unit and to reduce the input control task of the protection

processor(s), which would otherwise require considerable complexity to cope with the intermittent data stream in a word by word transfer mode, an output buffer store is provided. This facilitates the accumulation of a complete sampling instant data set prior to block transfer.

3.8.2 Storage implementation

To eliminate the need for address control logic associated with a random access type of store, an integrated device based on the First In First Out (FIFO) memory principle has been adopted for the buffer module of the interface system. The buffer is organised as 32 words each of 16 bits.

A full description of the operation of FIFO memories is given in reference 37. The FIFO mode of operation does not impose any limitation on the composition of the output data block, which can be adjusted by programming alterations of the system controller (section 3.10) and selection of the input data source connections.

3.8.3 Output transfer control

The presence of a valid buffer data block is indicated by the setting of the Data Ready Flag (DRDY) and the store is arranged such that this flag is cleared upon transfer of the final word, thereby eliminating the requirement for the protection processor to maintain a block length counter during the transfer operation.

In addition the data ready flag is cleared during the period following the transfer of each stored word, and remains clear until the next data word has shifted to the buffer store outputs. Hence DRDY is only set when valid stable data exists on the interface unit output. Under control of this flag a maximum transfer rate of 1MHz i.e. 16 M bits/s is possible.

To load the data block an external processor is only required to sense the state of the data ready flag and generate a series of transfer clock pulses (TFCL) until the flag remains in the reset state. The output enable signal (OE) is available to facilitate the connection of multiple interface units on a common output data bus, thereby readily expanding the data acquisition function. When this enable line is at a logical '0' state the buffer store outputs of the unit are in a high impedance "OFF" condition.

The timing sequence for the data transfer operation is shown in Fig. 3.15 and Fig. 3.16 is a block diagram of the overall store module.

3.9 The Phased-Locked Sampling Clock

3.9.1 Data acquisition control

The data acquisition and conversion processes performed by the interface system take place at pre-defined

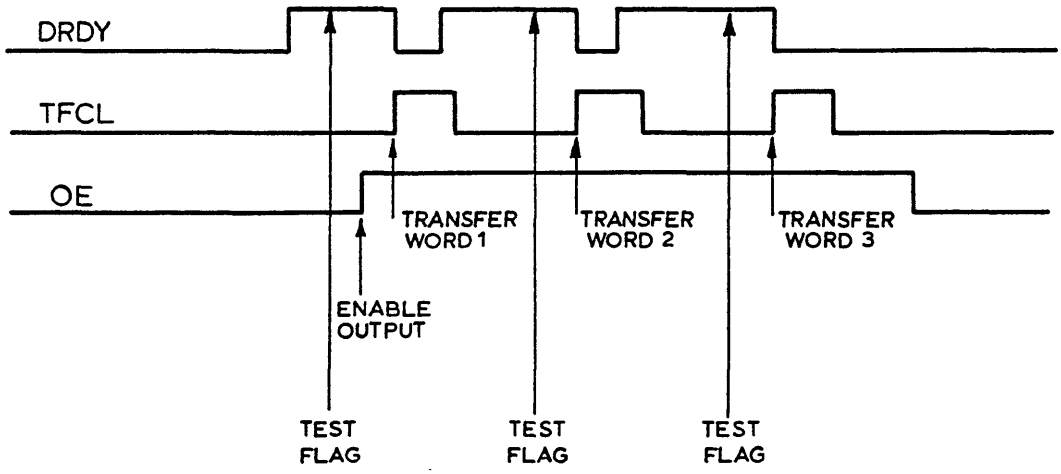


Fig. 3·15 Data transfer timing sequence, shown for a 3 word block.

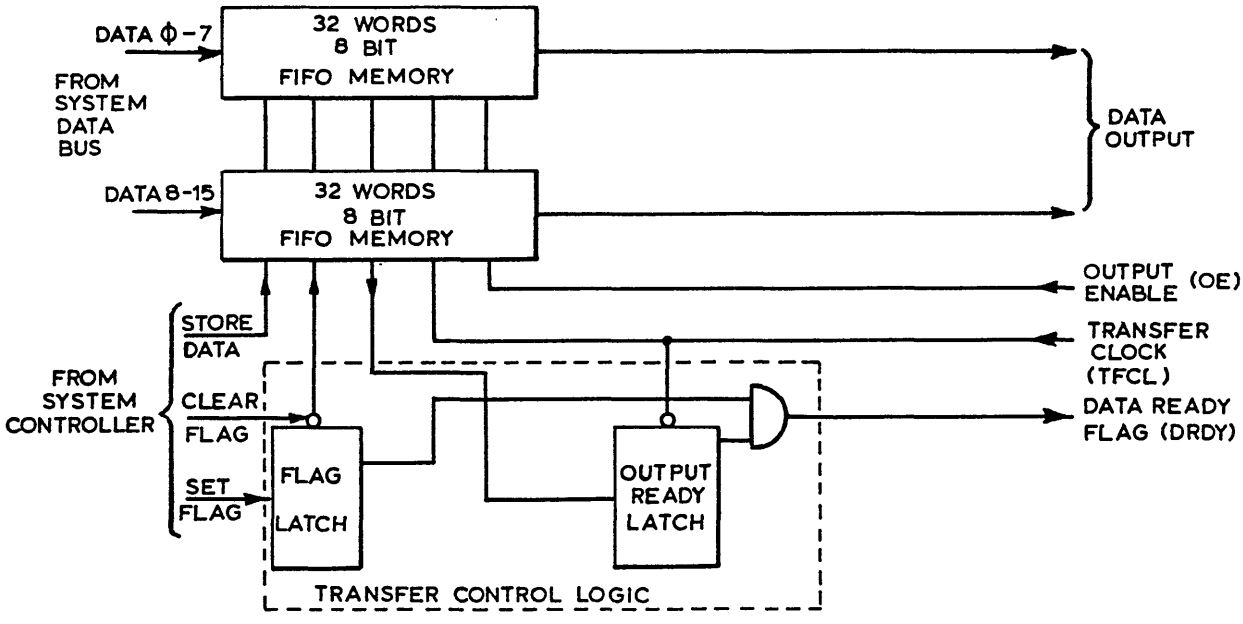


Fig. 3·16 Output store module block diagram.

intervals and are initiated by pulses generated in the sampling clock module.

The sampling clock must fulfil the following requirements:

i) To provide multiple samples of the power system quantities within a single cycle of the system frequency, a frequency multiplication method must be used to produce the necessary sampling control.

ii) If the data sampling instants are synchronised to the power system frequency, valuable phase and frequency data becomes available for use in subsequent processing. In using the interface as part of a larger integrated back-up protection scheme, the ability to synchronise the sampling instants throughout the network is essential if erroneous or missing data is to be replaced by redundancy, estimation or substitution techniques.

iii) The requirement of (ii) above implies that the sampling clock module must be capable of synchronising to an external reference signal, which may be derived from either the primary system frequency or a remote master "system time" clock[38].

iv) In the event of a loss of the reference signal the data sampling must continue independently with a minimum change in period.

3.9.2 Phase-locked oscillators

The criteria described in the preceding paragraphs can best be satisfied by a sampling clock module based on the phase-locked oscillator principle.

This technique has found widespread application in many fields, particularly those involving communications, and considerable documentation is available which provides a comprehensive analysis of the design parameters involved[39,40].

Fig. 3.17 shows the fundamental components of a phase-locked oscillator, and the similarity to a control system is immediately apparent. The phase detector compares the relative phases of the input reference and feed-back signals to produce an error signal, which is then modified by the loop compensation network before being applied as a control signal to the voltage controlled oscillator. In applications requiring frequency multiplication the voltage controlled oscillator operates at the desired multiple, n, of the input reference frequency and a division by n completes the feed-back path.

An addition to the loop compensation network is required in frequency multiplication applications to overcome frequency modulation effects which are a consequence of the phase detector characteristics. The phase detector is essentially a mixing device and its output contains both sum and difference frequency components, the latter being d.c and the required error signal. The sum component,

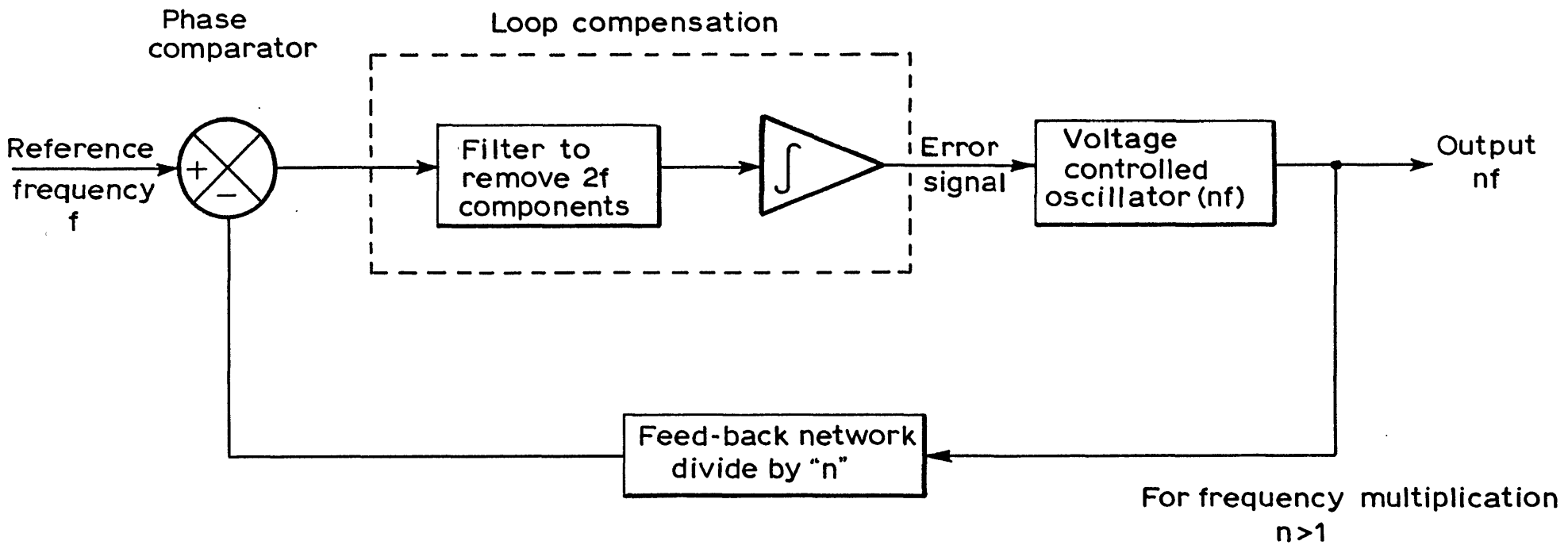


Fig. 3·17 Basic components of a phase locked oscillator

however, with a fundamental frequency of twice the reference signal must be removed by the loop filter to prevent frequency modulation of the voltage controlled oscillator and a resulting time jitter in the sampling pulse output.

3.9.3 The sampling clock module

The sampling clock used in the interface and shown diagramatically in Fig.3.18 was designed to operate in synchronism with input reference signals whose frequency lies within the capture range of 42-58 Hz. It is based on a single integrated device which incorporates the phase detector, voltage controlled oscillator, and loop compensation amplifier[41].

Two series of sampling rates are available from the clock, i) 2,4,8,16 times the reference frequency provides rates which Ranjbar[9] concludes are optimum in many respects for transmission line protection applications, and ii) 3,6,12,24 times the reference frequency which allows sampled data to be readily processed in applications requiring phase angle information, (Chapter 5).

To achieve these sampling rates the voltage controlled oscillator operates at 48 times the nominal input reference frequency and two division chains, modulo 2 and modulo 3, comprise the feedback network. The outputs of the dividers are used by the rate selection logic to provide the clock control logic with a pulse waveform at the

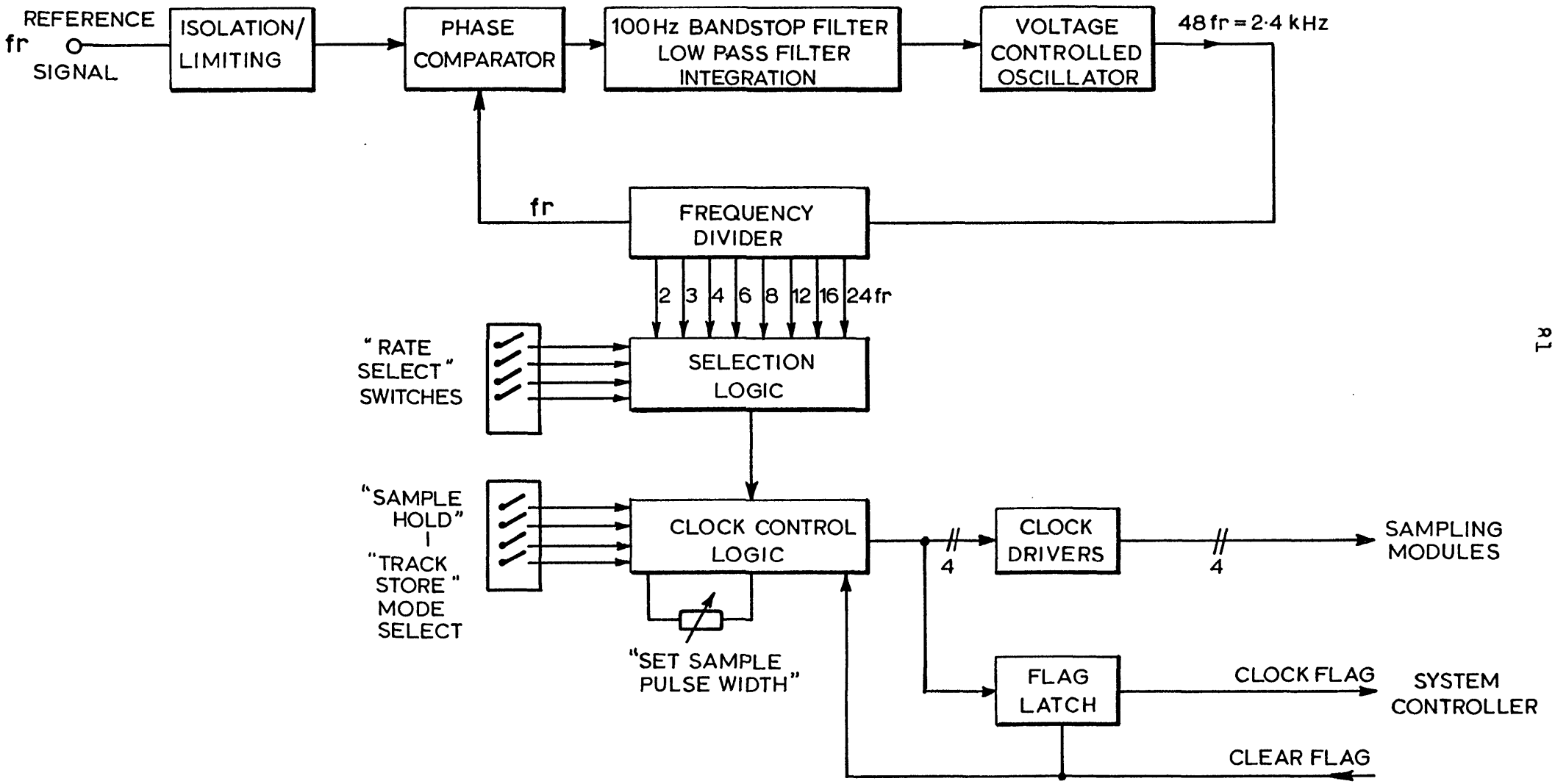


Fig. 3·18 Sampling clock module.

desired clock rate and depending upon the mode select input to this logic the output sampling clock will cause the sampling modules to operate in either the 'track store' or 'sample hold' mode.

As previously described the inclusion in the loop compensation network, of a filter to remove unwanted components from the phase comparator output, is essential in frequency multiplication tasks and is achieved in the clock module by a passive twin-tee 100 Hz bandstop arrangement.

Finally, the clock flag latch which is set by the sampling clock output provides the system controller with an indication that a data sample is present and thereby initiates the acquisition cycle.

3.10 The Interface System Controller

3.10.1 Controller functions

To realise the maximum benefit from common modular data acquisition equipment for all types of protection schemes, the interface controller must be flexible. Only a stored program device will meet these requirements.

The two major functions which the controller must provide are:-

(i) Configuration control.

The usage of the various elements within the interface must be subject to programmable variation to cater for varying input data sources.

(ii) Timing control.

The controller must also be able to regulate the timing of the interface components by providing control pulses and device status testing. Again this function must be programmable since configuration alterations will necessitate corresponding changes in timing.

3.10.2 Controller module

Since the interface is to operate solely in a data acquisition role without a pre-processing facility, the complexity of the system controller can be limited. Fig. 3.19 is a block diagram of the purpose designed interface controller, which used only a control program memory, sequencing logic and instruction decoding logic.

The controller memory comprises two 32 word 8 bit P.R.O.M's to provide a total storage capacity of 32-16 bit instructions. A control program stored in these devices is executed by the application of an address sequence generated by the store sequence control logic. Memory sequencing is based on a program counter provided with count modification inputs from the jump address (section 3.10.3) portion of the control instruction, and the device status

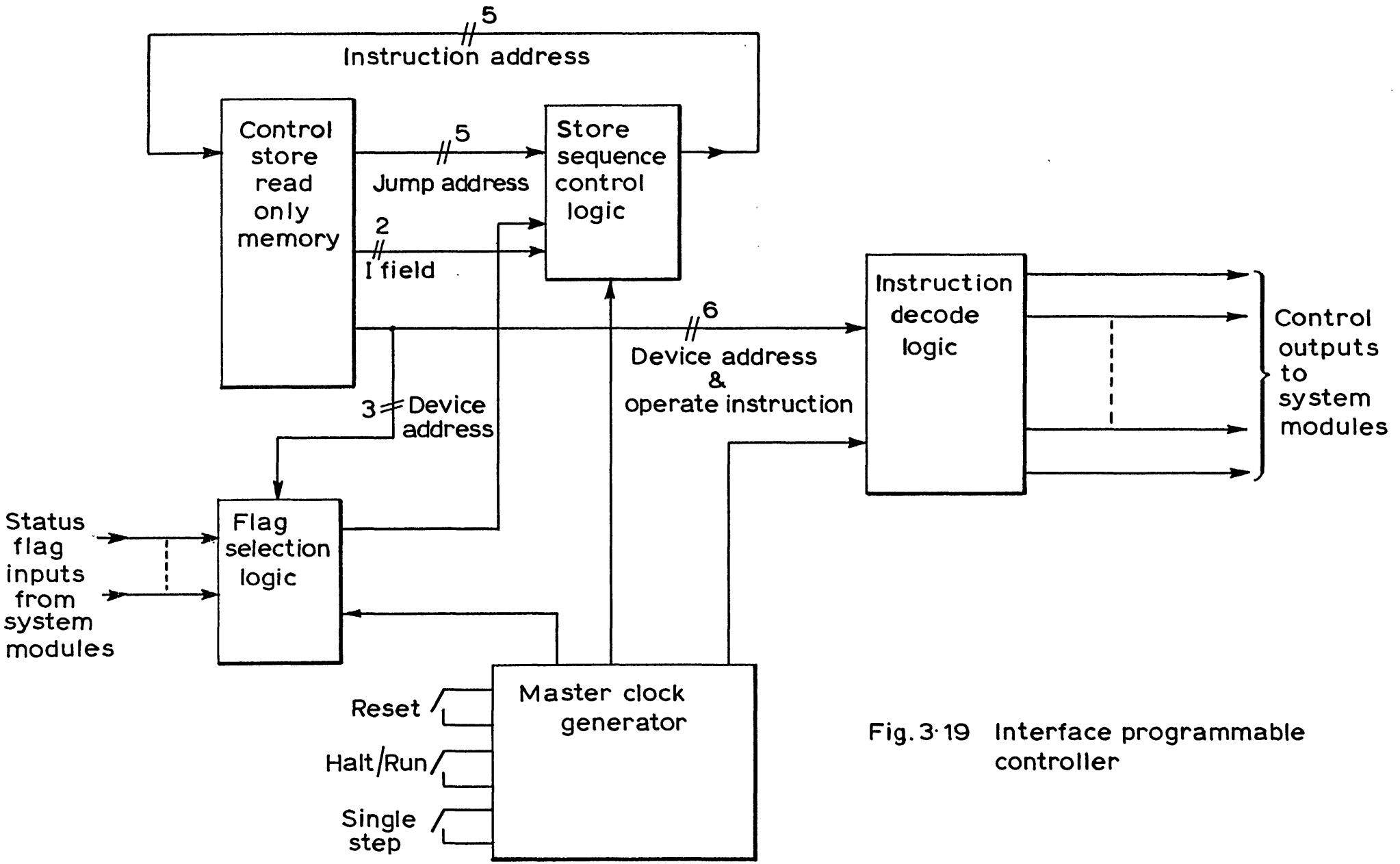


Fig. 3·19 Interface programmable controller

flag selection logic, which allows the controller to respond to the condition of the interface elements.

Control actions are derived from the instructions by the instruction decode logic which is largely implemented by a 256 word 8 bit P.R.O.M, thereby reducing the number of devices required for this combintorial task. The control outputs are unique signals directed to the individual elements of the interface. Although some flexibility is lost by this organisation of the control distribution, the possibilities of malfunction which can occur on common bus systems are much reduced.

Finally the overall operation of the controller is timed by clock pulses from the master clock generator. The 2μs instruction execution cycle determined by the clock generator is an optimum compromise, in that it allows sufficient time between operations to compensate for the settling and transient effects present in the analogue devices of the system, whilst ensuring that the performance of the unit is not limited by prolonged controller processing.

3.10.3 Controller instructions.

To fulfil the control functions outlined in 3.10.1 an instruction format must be employed which provides both configuration data i.e. an address group, and timing data, i.e. an operation group.

The address group:

This group must be able to define exclusively individual devices within the interface namely:

1-4) The digital input ports A-D (DIG A,B,C,D)

5) The analogue multiplexer (MUX)

6) The analogue-digital converter (ADCV)

7) The phase locked clock (RTC)

8) The output store "data ready" flag (DRDY)

To provide these 8 possible addresses the group must comprise at least 3 bits of an instruction word.

The operate group:

The control signals required by the devices differ. However, for simplicity of format they may all be classified as one of the following:

1) Clear/reset the device or its flag (CLR)

2) Clock the device or flag (PLS)

3) Determine the device status (FLG)

4) Transfer data between devices (STR)

These 4 basic operations will therefore require at least a 2 bit group for definition.

The program jump group:

In addition to the groups already detailed, information must be available in each instruction to provide the next instruction memory address. This information facilitates program jumps, either as the result of a direct instruction, (JMP), or upon receipt by the controller of

status data from a device within the interface. The flexibility and power of the controller are thus greatly enhanced.

The jump group consists of 2 parts:-

1) The jump address field which provides the post jump instructions address and is of a length determined by the direct jump range required.

2) The 2 bit wide I field which is employed to select the memory address counter mode. A program jump may be directly initiated by the inclusion of the appropriate I field code in an instruction. During the execution of a program jump the I field also prevents the generation of false address or operate signals.

For device status initiated jumps, the address and operate groups allow the device status data to modify the I field, thus causing a transfer of program control to the jump address specified.

Table 3.1 lists the complete controller instruction set.

Analogue Multiplexer:	
Reset Multiplexer to starting channel	CLR MUX (X)
Increment Multiplexer	PLS MUX (X)
Test Multiplexer for final channel	FLG MUX (JA)
Analogue - digital converter:	
Start conversion	PLS ADCV(X)
Test for end of conversion	FLG ADCV(JA)
Clear conversion complete flag	CLR ADCV(X)
Transfer data to output store	STR ADCV(X)
Digital input ports:	
Transfer data to output store	STR DIG A(BCD)(X)
Phased locked clock:	
Test clock flag status	FLG RTC (JA)
Clear clock flag	CLR RTC (X)
Output store data ready flag:	
Set output ready flag	PLS DRDY(X)
Clear output store and flag	CLR DRDY(X)
Program control transfer:	
Jump to location specified	JMP Ø (JA)

( ) Indicates the contents of the jump address field. X= irrelevant, JA = the location address to which the jump is directed. In the case of the FLG instruction the jump occurs when the flag condition is true e.g. FLG RTC (JA) will cause a jump to JA when the flag is set, otherwise the next successive instruction will be obeyed.

Table 3.1

3.10.4 Controller instruction format

Studies of possible data acquisition tasks indicate that a 32 word program capacity is adequate and consistent with currently available P.R.O.M's.

The controller instruction word must accommodate at least a 3 bit device address group, 2 bit operate group, 2 bit I field and the jump address field. A word length in excess of 7 bits is thus required, and a suitable memory option of 32 16 bit words enables a full 5 bit jump address to be included. 4 bits of the word are unused.

The 16 bit word is organised as:-

I_0 I_1	F_0 F_1	1 Bit	D_0 D_1 D_2	3 Bits	5 Bits
I field	Operate Group	Not Used	Device Address Group	Not Used	Jump address field

Appendix 1 contains a complete list of the controller instruction bit assignments and example programs for the interface.

CHAPTER 4

THE PROTECTION PROCESSOR

4.1 Introduction

4.1.1 Processor functions.

The second unit of a dedicated digital relay, the protection processor, must provide the computational power to evaluate the data samples collected by the acquisition interface. A major feature of this unit must therefore be an ability to perform arithmetic and logical operations upon the measured data. From the outcome of these data processing functions the unit must reach a decision regarding the state of the power system plant, thus fulfilling the final requirement of a protection relay.

To produce the flexibility necessary to meet various relaying tasks,the protection processor is implemented as a stored program digital computer. In Chapter 2 it was argued that microprocessor devices allow an economically realistic approach to the construction of the protection processor for a dedicated relay.

Microprocessor development has been rapid and many device families are now available for the design of digital systems. These families overlap each other in performance, design flexibility and ease of implementation.

A survey of the differing device capabilities and their suitability to the protection processor functional requirements forms an essential part of the design task.

4.1.2 Protection processor performance.

Digital power system protection research has produced techniques and algorithms which have been developed, refined and tested in the majority of examples, with the aid of a mini-computer. Mini-computers are now well established, typically offering word lengths of 12-18 bits, comprehensive instruction sets and instruction execution times of 1-3μs for basic operations.

A study of the protection software developed at Imperial College and details given in published papers, indicate that mini-computer performance is adequate for digital relaying. However, for the more complex applications, notably transmission line protection, "starting" techniques[42] are frequently used to reduce the computational load to manageable proportions. Several workers also comment that an improvement in computer speed would provide additional advantages[43].

It is concluded that the performance of the protection processor should at least be comparable with that of a mini-computer.

4.1.3 Microprocessor catergories.

The protection processor performance criteria must then be compared with those of a range of microprocessor devices. For simplicity the profusion of microprocessors which are available can be classified into four major system design catergories. These are:

i) 4 bit MOS.
ii) 8 bit MOS.
iii) 16 bit MOS.
} Single integrated circuit devices

iv) Bipolar bit-slice.

Whilst devices which fall into catergories (i) and (ii) are able to perform most of the functions associated with a digital computer they suffer from several drawbacks. The limited word length requires multi-byte processing to provide usable instruction formats and to handle data words in excess of 4 or 8 bits. (The A-D converter output of the relay acquisition interface is 10 bits, plus range or identification data, Chapter 3) This limitation, and the inherently low speed of the MOS fabrication technology, reduces the performance of these devices to an unacceptable degree for high complexity tasks.

The extended word length of the 16 bit microprocessor produces a substantial improvement in performance when compared with the 4 or 8 bit types. Several 16 bit single package MOS microprocessors are available, some of

which have architectures identical to mini-computers. However, the fundamental speed limitation of MOS technology and constraints imposed by the number of pins in the device package,prevent these microprocessors from reaching the performance of 16 bit mini-computers.

More recent versions of 16 bit microprocessors employing bipolar fabrication methods offer true mini-computer performance. Currently these devices remain at a development stage and are not readily obtainable.

Bit-slice microprocessor families consist of basic processor components implemented with low power Schottky bipolar logic. A complete processor constructed with these components can have any desired word length and features fast operation. The pipelined architecture and micro-programmable software capability of the bit-slice approach provides a microprocessor system with a performance well into the high end of the mini-computer range. Bit-slice devices were therefore chosen for the protection processor.

To illustrate the different performance of the microprocessor types, table 4.1 shows in abbreviated form the results of a study undertaken by Penney[44]. Speed of execution, number of program statements and memory requirements are listed for a selection of representative tasks implemented on a variety of typical devices. For comparison the corresponding figures for the protection

	16 BIT ADDITION			8 BIT MULTIPLICATION			OUTPUT PROGRAMMED		
	Ns	BYTES	TIME (µs)	Ns	BYTES	TIME (µs)	Ns	BYTES	TIME (µs)
4 Bit	14	19	561	-	-	- (a)	14	19	124 (c)
Average 8 Bit	9.5	17.25	97	25	37	850	7.75	14.25	75.25
Average 16 Bit	3	10	19.33	12	31.33	255	5.66	11.33	39
Protection Processor.	3	6	2.6	3	6	19.6 (b)	2	4	1.8 (d)

Ns = number of source program statements.

BYTES = number of 8 bit bytes in compiled program.

a) Too complex, not attempted.

b) 16 bit multiplication (signed)

c) 4 bit characters.

d) These figures are for a single 16 bit word. Because of the addressing structure used they must be repeated for each word output.

Table 4.1

processor are included in the table.

4.1.4 The protection processor structure.

Fig.4.1 is a block diagram of the protection processor, showing a 16 bit central processor with micro-programmed control. Program and data memory is restricted to that required for relaying applications and the architecture provides a very efficient memory addressing structure. Input and output facilities are included to meet the particular demands of the protection task. To optimise the performance of the unit extensive use is made of pipeline techniques.

The factors related to the specialised protection application of the equipment and their influence upon the design of each processor component are described in the following sections. Again a modular construction method is employed to allow for flexibility of memory or input/ output requirements and to simplify maintenance of the unit.

4.2 The Central Processor unit (CPU)

4.2.1 Introduction.

To achieve high operation speeds, the bit-slice microprocessor families, considered here, are fabricated with bipolar semiconductor technology. Heat dissipation problems from which L.S.I. devices using this technology

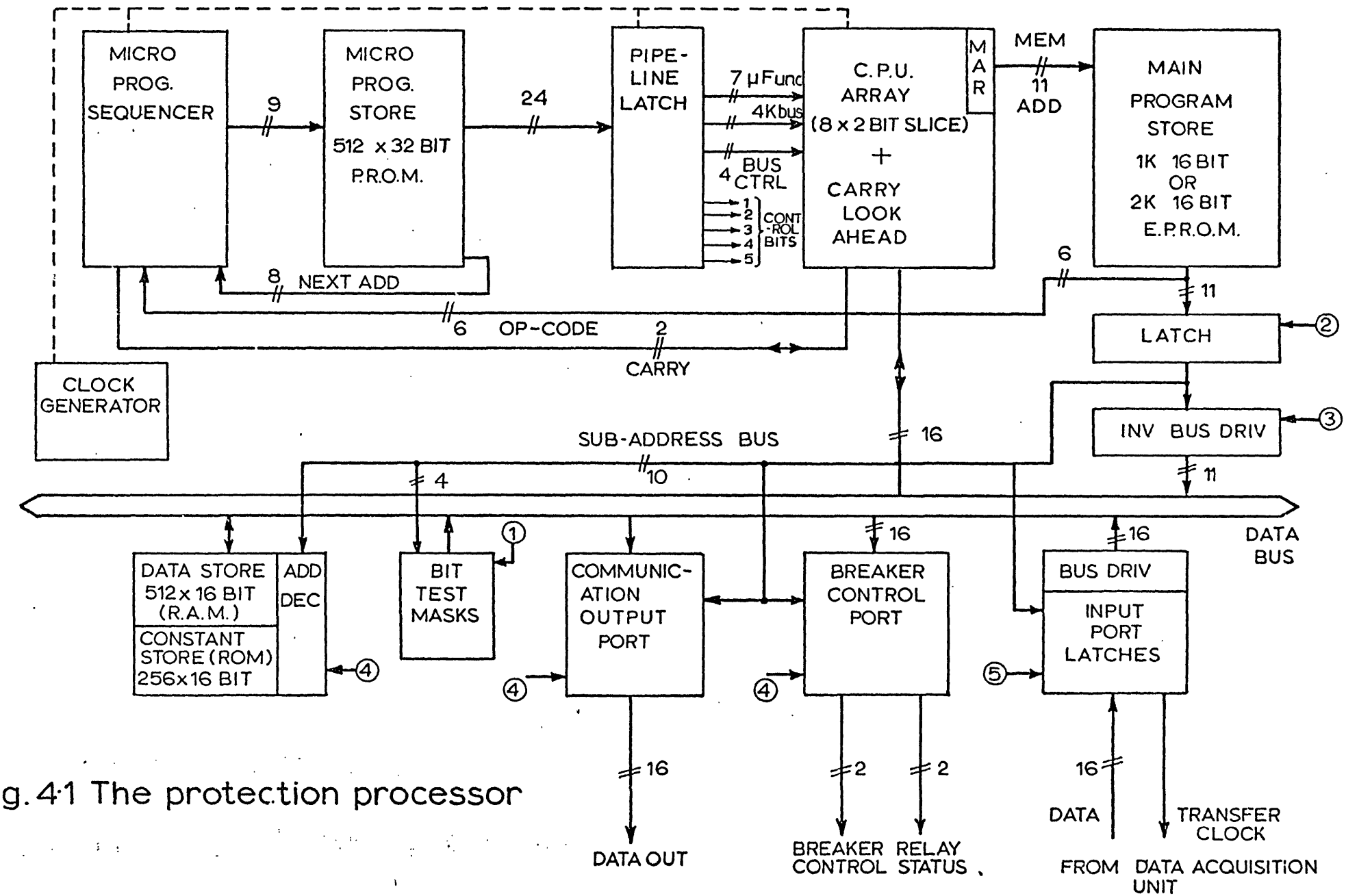


Fig. 4·1 The protection processor

suffer, have been overcome by restricting the complexity of individual devices. For the CPU functions these restrictions have produced integrated circuits which represent 2 or 4 bit wide slices through the central processing section of a digital computer. A complete central processor for a given word width can be constructed from an array, consisting of an appropriate number of these devices connected together[45].

Several families of bit-slice microprocessors are available, and each offers different CPU capabilities. The basic design choice lies between 2 bit or 4 bit slice components. In many respects the internal data processing options provided by the 4 bit devices are very attractive, particularly for the construction of general purpose computing equipment[46]. However, the smaller slice width of the 2 bit elements allows additional input/output data buses to be accomodated in a given number of package pins. They also offer a more basic set of processing instructions, and in these respects are well suited to specialised architectures such as the protection processor.

4.2.3 Two bit central processing elements.

Fig.4.2 shows the internal organisation of the 2 bit processing element[47] from which the CPU of the protection processor is constructed. The central feature of the device is the arithmetic and logic section (ALS)

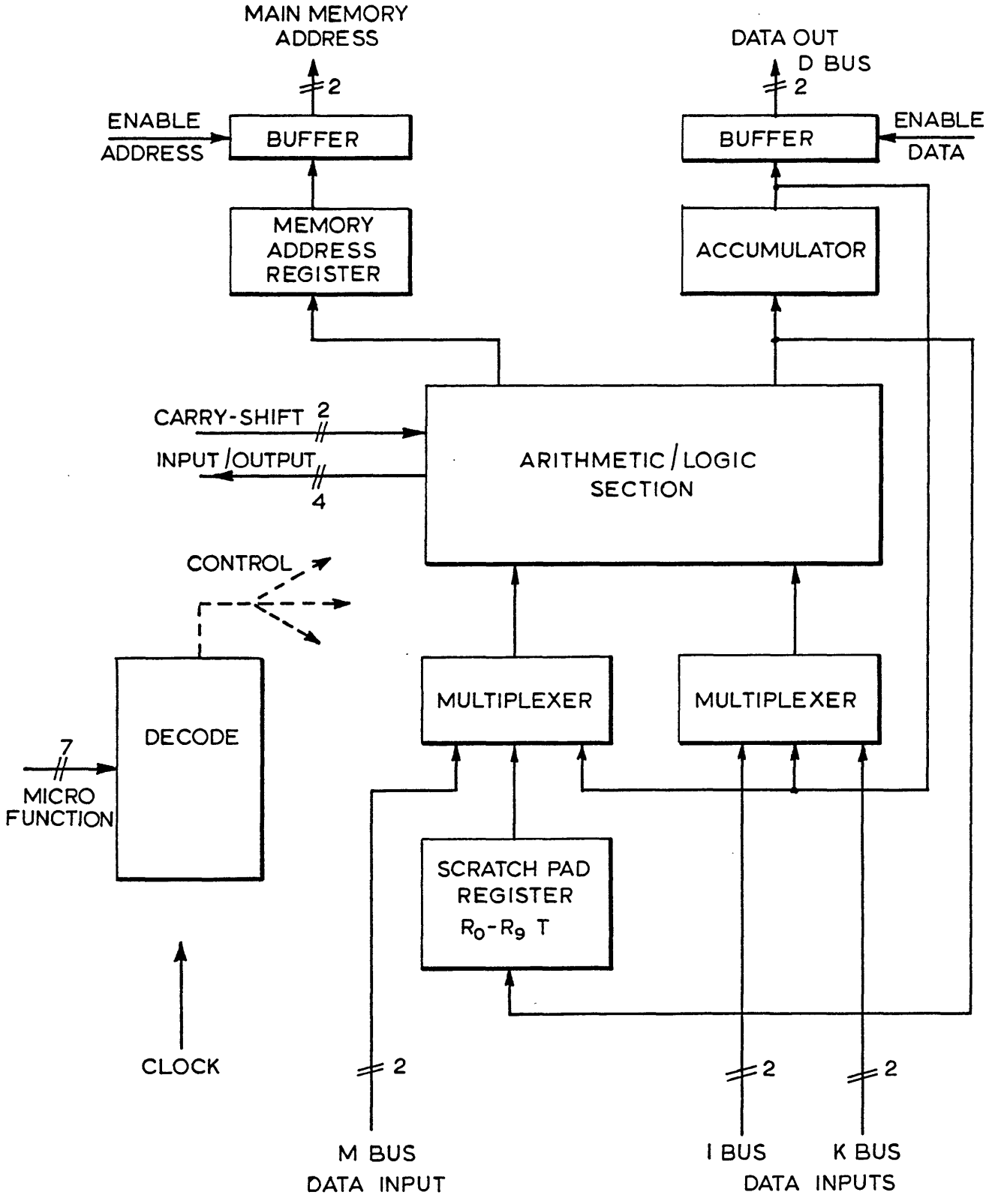


Fig. 4·2 Internal organisation of a 2 bit processing element.

which processes data supplied from internal or external sources, as determined by the two multiplexers. An independent accumulator register is available for storing the results of the processing operations. Alternatively, if memory address calculations are performed the ALS output can be directed to the memory address register. Output buffers associated with these registers interface the processor to the data and main memory address buses respectively.

Eleven general purpose registers designated RO-R9 and T, form a scratch pad within the device. The outputs and inputs of these registers are accessible to the ALS and they are allocated specific functions in the overall processor design. These functions are described in the following section, as is the organisation of the three data input buses M, K and I which are available.

Control of the device is achieved by data presented on the micro-function inputs in conjunction with a master clock signal. The micro-function is decoded internally to select a data source and destination, and to specify the ALS function.

4.2.4 Register functions and bus organisation.

The devices described in the preceding section are the basic elements of a bit-slice microprocessor. A complete CPU realised by an array of these devices is termed the macro-level machine. It is at this level that the processor is of interest to the relay programmer. To

execute the instructions which form the application program of the equipment, the macro-CPU must provide several registers which are available to the programmer and others which are required for internal "house keeping" functions. The registers of the bit-slice processing elements are allocated for these functions as follows:

i) Accumulator (AC)

The protection processor is designed as a single accumulator machine. The macro-CPU accumulator (AC), holds the results of arithmetic or logical operations. All programmed data transfers to or from storage or output ports, are routed via this register.

The processing element accumulator is used for this function. This approach reduces the internal data transfer operations which would be required if a scratch-pad register is used and therefore improves the CPU performance . However, it must be noted that the processing element accumulator may be modified by instructions unrelated to AC manipulation, e.g. flag status testing. In these cases temporary storage is provided by the scratch-pad register R9 for the contents of AC.

ii) Program Counter (PC)

The execution sequence of the application program is controlled by the main memory address output of the CPU. This address is supplied by the memory address register of the processing elements which in turn is maintained by

the program counter (PC), RO in the scratch-pad is used as the PC. R7 stores the PC contents during subroutine calls i.e. the main program return address following the completion of the subroutine.

iii) Loop Counter (LC)

To simplify multiple loops in a program the CPU provides a loop counter register. Single instructions are required to either load this register with the loop count or decrement the count and perform a "skip if zero" test. R8 is the loop counter.

iv) Multiple shift counter (SC)

Instructions are included in the processor set to facilitate multiple left or right shifting of the accumulator contents. A part of these instructions specifies the number of shifts required. During execution, this value is stored in the shift counter (register T of the processing elements), which is decremented and tested for zero to control the shifting function.

v) Multiplication registers.

The registers R1 and R2 are used by the multiplication instruction for counting and temporary storage functions. 32 bit products of the multiplication are stored in the accumulator (MSBs) and T register (LSBs).

In a similar way the input/output buses of the processing elements are organised to form the bus structure

of the macro-machine. The buses are:

i) Data bus.

A bi-directional data bus is central to the protection processor. Through this bus the CPU communicates with memory devices and I/O ports. The M and D buses of the processing elements are paralleled to produce the data bus. An enable data control signal (Fig.4.2) determines the bus operation mode i.e. data to or from the CPU. When the bus is in the CPU input mode it is also available for direct data transfer between the unit input port and data memory. During this transfer the CPU is inactive and data present on the bus has no effect on CPU operation.

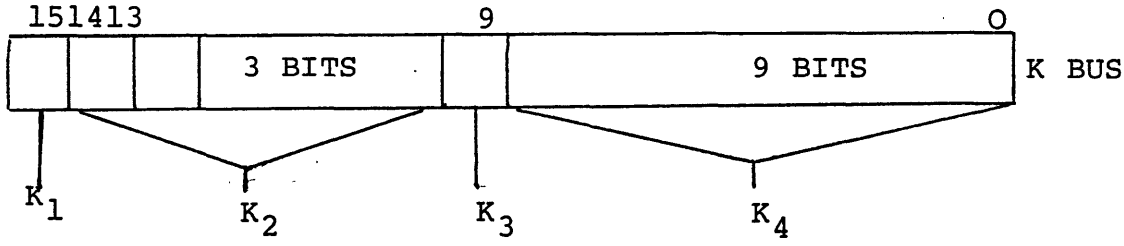
ii) Program address bus.

The main address bus outputs of the CPU are used solely to address the program memory of the processor. This bus is permanently enabled and outputs the contents of the CPU memory address register. Although the bus has a full 16 bit width only the 11 LSBs are required to address the program memory which is limited to 2K words, (Section 4.4.1).

For reasons discussed in Section 4.5.2, the processor architecture employs a separate sub-address bus to access data storage and I/O ports. The sub-address bus is derived directly from the instruction register and has no connection with the CPU.

iii) Bit testing-masking bus.

Arithmetic and logical operations of the CPU are greatly influenced by the K bus data inputs. When used in conjunction with the controlling micro-function bus, the K bus allows direct masking or bit testing of the ALS data. The K bus data is specified by 4 bits of each micro-program word and is organised to facilitate several specialised functions as shown in table 4.2. below.



K BUS DATA K_1	K_2	K_3	K_4	Function
0	0	0	0	Select complete 16 bit word. No masking, i.e. normal operation.
1	1	0	1	Select A-D converter data sign for testing(a)
1	1	1	0	Select A-D data magnitude(a)
1	0	1	1	Select input data range or ident bits 2,3(a)
0	1	1	1	Select word sign or ident bit 1(a)
1	0	0	0	Select magnitude of word.
0	0	1	1	Extend sign of A-D data.

(a) These functions are related to the format of the data provided by the accquisition interface, (Chapter 3).

Table 4.2 K bus functions.

It should be noted that the K bus is used exclusively to control the CPU at the micro-instruction level. An entirely seperate masking memory (section 4.5.4) is employed during macro-instruction bit testing.

iv) Flag input bus.

Peripheral equipment status flags which must be tested during program execution are connected to the CPU I bus. In the relay design only one status flag is involved, that of the data acquisition interface output. This flag is connected to bit 9 of the I bus which allows the A-D converter sign mask (provided by the K bus) to be used in flag testing instructions.

4.2.5 The CPU module.

The 16 bit CPU module comprises an array of 8-2 bit processing elements and a carry look-ahead generator[48]. This device enhances the arithmetic performance of the CPU by anticipating a carry across the complete 16 bit word. Delays which result from carry ripple-through are thus eliminated. A block diagram of the complete CPU module is shown in Fig.4.3.

4.3 Processor Control.

4.3.1 Microprogramming.

The basic operation of the control portion of

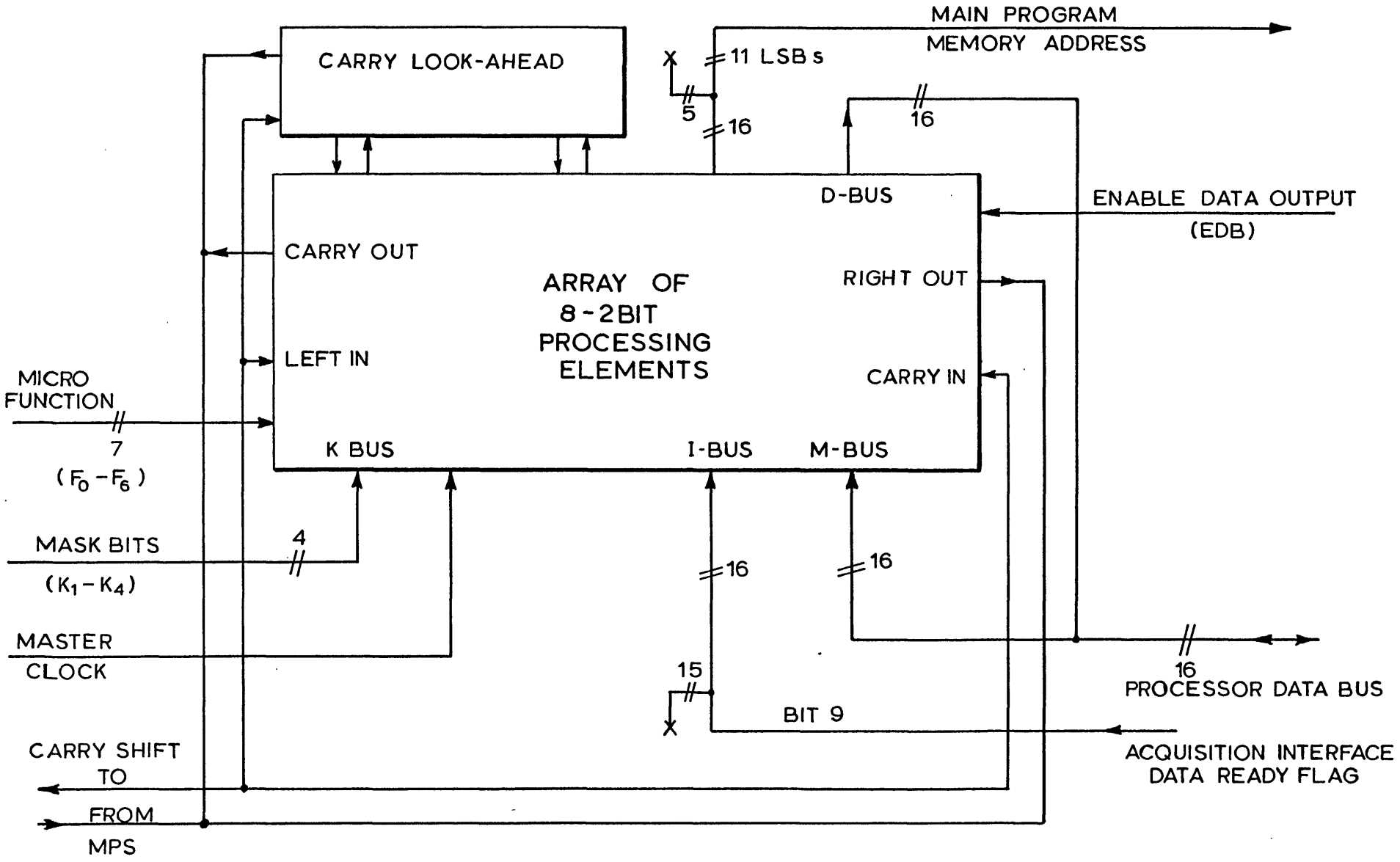


Fig. 4-3 Central processor module.

the processor consists of selecting instructions sequentially from memory and executing them. To achieve this it issues a series of control commands to the CPU and memory devices.

Conventionally the control portion is realised by hard-wired logic. An alternative method of control unit design, proposed by Wilkes[49], uses command "tables" stored in a control memory. These tables form the micro-program control of the machine.

In a micro-programmed computer each main program instruction (macro-instruction) is broken down into several micro-instruction steps. The micro-instructions are executed sequentially and generate the control signals which cause the processor to perform the function specified by the macro-instruction.

Micro-program control has a number of advantages which are not provided by a fixed logic design. They are:

i) Flexible instruction set.

Since the function of each macro-instruction is defined only by the micro-program, an optimum instruction repertoire can be designed for specialised applications.

ii) Simplified hardware design and modification.

Modifications or additions required during the design of the processor CPU and memory hardware can be made without affecting the control unit design. Changes can be

easily accommodated by alteration of the micro-program format.

iii) Simplified maintenance.

Maintenance of the processor is simplified, since only sequences of elementary operations are implemented in the hardware. Consequently the structure is relatively uncomplicated and very effective test programs can be devised at the micro-instruction level. i.e. micro-diagnostics.

These attributes are particularly valuable for special purpose control processor applications, as required in the digital relay.

4.3.2 The control module.

The control module of the protection processor is shown in Fig.4.4.and can be seen to consist of 3 major components: i) the micro-program memory. ii) the micro-program sequencer (MPS) iii) a pipeline register.

4.3.3 Micro-instruction format and the micro-program memory.

The control micro-program consists of many micro-instructions. Each micro-instruction must be able to fully define control actions for the CPU, the memories and the I/O ports of the processor. In addition, the instruction must contain information which identifies the

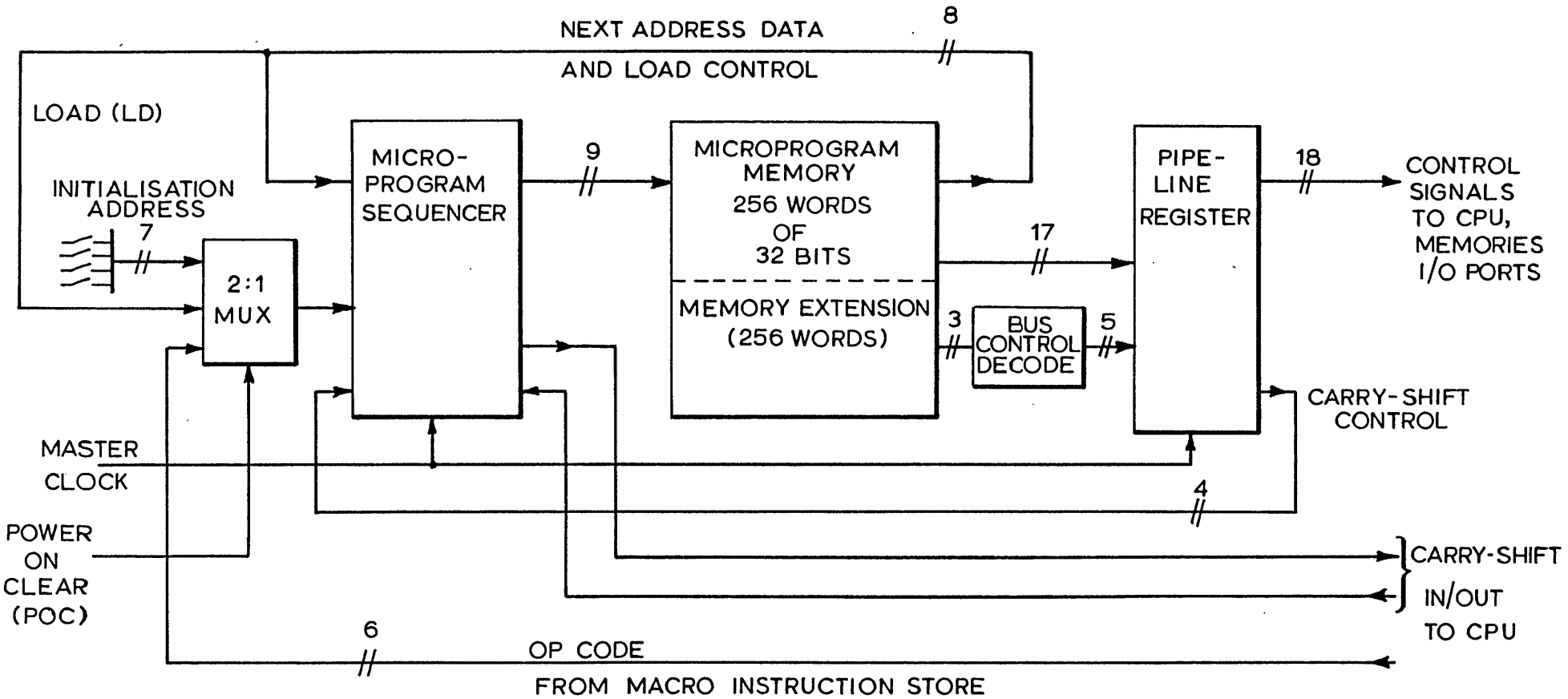
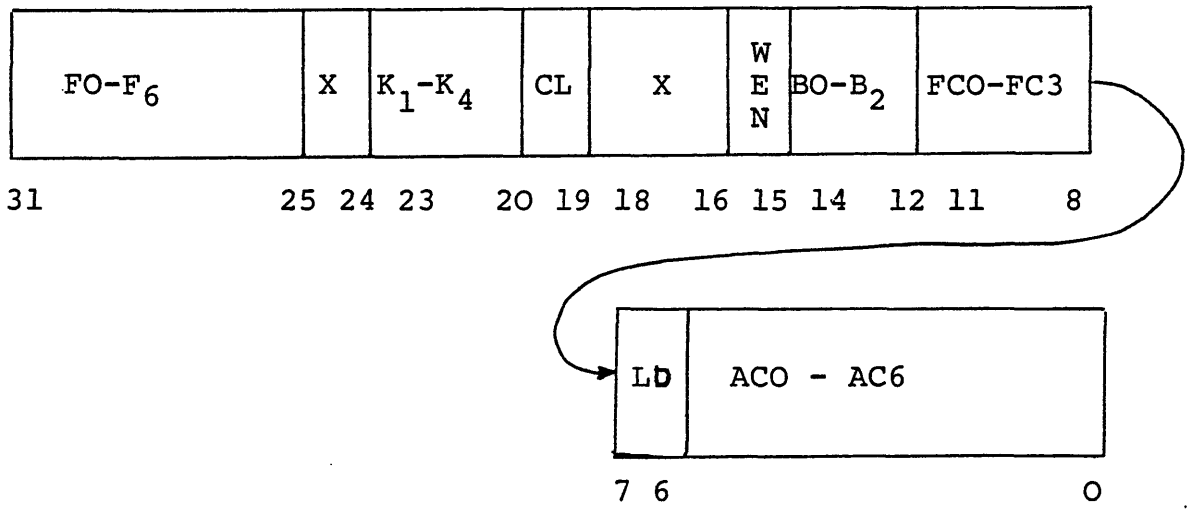


Fig. 4·4 Processor micro-program control module.

next micro-instruction required.

To satisfy these criteria, a 32 bit word length is used for the micro-instructions. The format of a micro-instruction word is shown below and the function of each field is described.



X Not used.

FO - F6 The micro-function field. This field supplies the primary control data to the processing element array of the CPU.

K1 - K4 The K bus descriptor, used in conjunction with the micro-function field to control CPU operation.

CL Clock latch. This bit provides a clock signal to the macro-program memory pipeline register during the instruction fetch cycle.

WEN Write enable clock for the RAM data memory and output ports.

B0-B2 Bus control. These 3 bits are encoded and determine which data source is connected to the processor data bus as follows:

ED - Enable data output of the CPU

JMPE - Enable the jump address or constant output field of the macro-memory.

IPE - Enable the input port.

EMM - Enable the masking memory.

REN - Read enable the data or constant memory.

The remaining 12 bits of each micro-instruction are used by the MPS.

FC0-FC3 The flag field, which controls the carry-shift inputs and outputs of the CPU.

LD Load. This single clock bit forces the op-code of a macro-instruction onto the MPS micro-program address outputs. It is used to define the start of the micro-program sequence which executes the macro-instruction.

AC0-AC6 From this field the MPS computes the address of the next micro-instruction required.

A maximum of 512 micro-program locations can be addressed by the MPS. In the protection processor, the memory is divided into pages of 256-32 bit words implemented

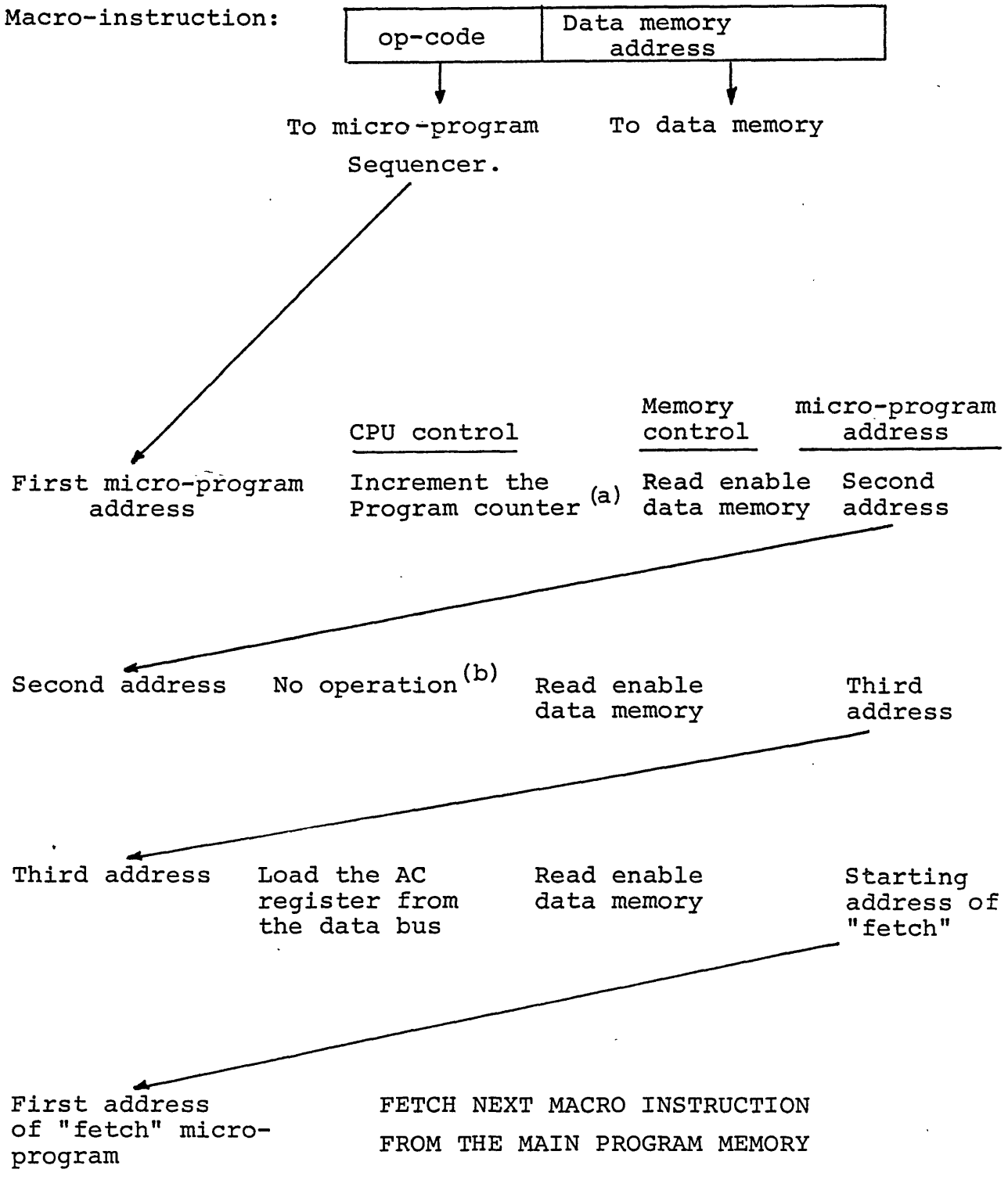
by four 256-8 bit PROMS[50]. The first 256 word page contains the micro-code required to implement the basic instruction set (Section 4.8.2). An extended instruction set is facilitated by the addition of a further 256 word page, to cater for more complex protection applications.

4.3.4 Micro-program execution and the micro-program sequencer.

The execution of micro-instructions which implement a micro-program is controlled by the micro-program sequencer (MPS). This single integrated circuit[51] provides a series of addresses to the micro-program memory and thus determines the order in which micro-instructions are executed.

Initially, the starting address for an area of micro-program is derived by the MPS from the operation code of the macro-instruction. Further addresses in the particular sequence are supplied by the device in response to control data included in each micro-instruction word, as described in the previous section.

To illustrate the process, the following micro-program example realises the "LOAD ACCUMULATOR FROM DATA MEMORY" macro-instruction.


Macro-instruction:
op-code
Data memory address
To micro-program Sequencer.
To data memory
CPU control
Memory control
micro-program address
First micro-program address
Increment the Program counter (a)
Read enable data memory
Second address
Second address
No operation (b)
Read enable data memory
Third address
Third address
Load the AC register from the data bus
Read enable data memory
Starting address of "fetch"
First address of "fetch" micro-program
FETCH NEXT MACRO INSTRUCTION
FROM THE MAIN PROGRAM MEMORY

(a) Prepare main program memory address for the next instruction fetch cycle.

(b) Do nothing, wait for access time of the data memory to elapse.

The MPS also controls the carry and shift inputs to the CPU. It has the facility to modify a micro-program address sequence following internal testing of the carry or shift outputs from the CPU. Conditional testing and program branching instructions can therefore be implemented.

Finally during start-up of the processor, additional logic forces the MPS output to micro-program address zero which is the start of an initialisation routine, (Section 4.7.2).

4.3.5 Pipeline registers.

To fetch and execute a series of micro-instructions the control portion of the processor must perform several functions. These are:-

i) Current micro-instruction address control field to MPS.

ii) Computation of next address by MPS.

iii) Application of this address the micro-program memory.

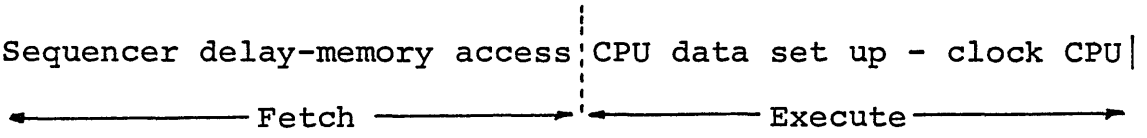
iv) Next micro-instruction available from memory.

v) Execute micro-instruction.

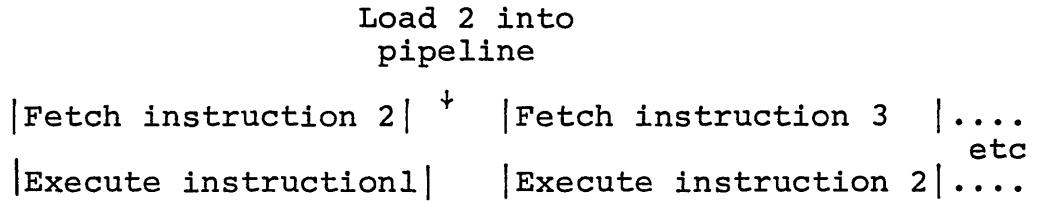
vi) Repeat.

Each of these functions has a corresponding timing requirement. Those concerned with the micro-instruction fetch, i.e. i, ii, iii, iv must take account of the micro-program memory access time and the propagation/set-up delays of the MPS. The execution phase, v, is governed by similar constraints in the CPU array.

A complete micro-cycle (fetch and execute one micro-instruction) thus becomes:



The provision of temporary storage, a "pipeline" register, for the CPU and memory control bits of the micro-instruction word enables the fetch and execute operations to be overlapped. i.e.



The pipeline register, which is located between the micro-program memory outputs and the processor control buses reduces the micro-cycle time by 30%. A corresponding improvement in the overall performance of the processor results.

Pipeline techniques are also employed in the main program memory outputs to produce a similar overlap of macro-instruction fetch and execute cycles.

4.4 The Macro-Program Memory.

4.4.1 Program storage requirements.

Application programs which determine the function of the digital relay are stored in the macro-program memory module. The program storage capacity required will clearly be determined by the design and implementation of specific algorithms. Additionally, the efficiency and power of the processor instruction set has a considerable influence on storage requirements.

To determine an appropriate memory size for the dedicated relay, three factors are considered. They are:

i) An examination of several protection algorithms developed using a single accumulator mini-computer (a DEC PDP15), indicates that typically less than 1K words of assembler level instructions are employed. This figure refers only to the basic fault detection routines and excludes input-output software, e.g. graphics routines and device handlers.

ii) In dedicated relay applications, fault detection algorithms will generally be required to operate within the primary data sampling interval. This interval is

unlikely to exceed 2.5 ms. i.e. 8 s.c. With a mean instruction execution cycle of 1μs, it is obvious that the program length must be restricted to 2.5 K words.

The ability of a digital processor to perform repetative tasks within program loops can considerably reduce the overall size of a program. However, it must be noted that since instructions within a loop are executed many times, the dynamic performance of the program does not improve and hence timing constraints remain. The 2.5K instruction limit therefore applies to the dynamic op-code frequency which will invariably be greater than the static frequency, i.e. the absolute program length.

iii) Execution of each macro-program instruction is defined by a micro-program sequence. This facility allows the processor instruction set to be readily extended by the addition of further micro-code sequences. Special purpose or frequently used operations can thus be reduced to single macro-instructions, with corresponding savings in main program size and execution time.

A maximum program storage capacity of 2K-16 bit instructions is therefore provided which will be adequate for the majority of applications.

4.4.2 Storage devices.

It is most important that protection equipment is able to resume correct operation following interruption of its power supply, particularly when the installation is unattended or inaccessible. For digital relays, therefore, it is necessary to ensure that application programs are retained during power failure. This can only be achieved by the use of non-volatile memory, such as that provided by semiconductor PROMS.

Specifically the MOS erasable PROM (EPROM)[52] offers many advantages, the major of which are:

i) High storage density. Single devices with capacities of 16K bits are readily available, which greatly reduces interconnections and hence improves reliability.

ii) Low cost.

iii) Low power consumption.

iv) Erasable. During software development the ability to erase and re-program the device is very valuable.

v) Mask-programmable replacement.

The availability of directly compatible memory devices, which are programmed during manufacture, enables high volume memory costs to be reduced once application software has been finalised.

4.4.3 The memory module.

The program memory module shown in Fig.4.5, comprises 3 areas:-

i) Program ROMs. Program storage is provided by 1K -8 bit EPROMS used in pairs to give pages of 1K-16 bit instructions. A maximum of 2 pages i.e. 2K words, can be accomodated in the module. By using only a single ROM pair, this organisation allows savings in memory power and cost to be made for small programs of less than 1K words.

An 11 bit memory address is supplied to the module by the program counter of the CPU via the memory address register. The MSB of this address forms the page select control which determines the ROM set in use.

ii) Pipeline register. The memory devices have an access time, i.e. address in to data out, of 450ns. To achieve an average instruction fetch and execute cycle of 1μs, it is therefore necessary to include a pipeline register in the memory data outputs. This register enables the fetch and execute operations for successive instructions to be overlapped with consequent time savings. The principles of pipeline operation are described in section 4.3.5. Instructions which cause a transfer of program control, e.g. program jumps and skip on arithmetic or logical test (if a skip occurs) reduce the effectiveness of the pipeline. In these examples the program sequence

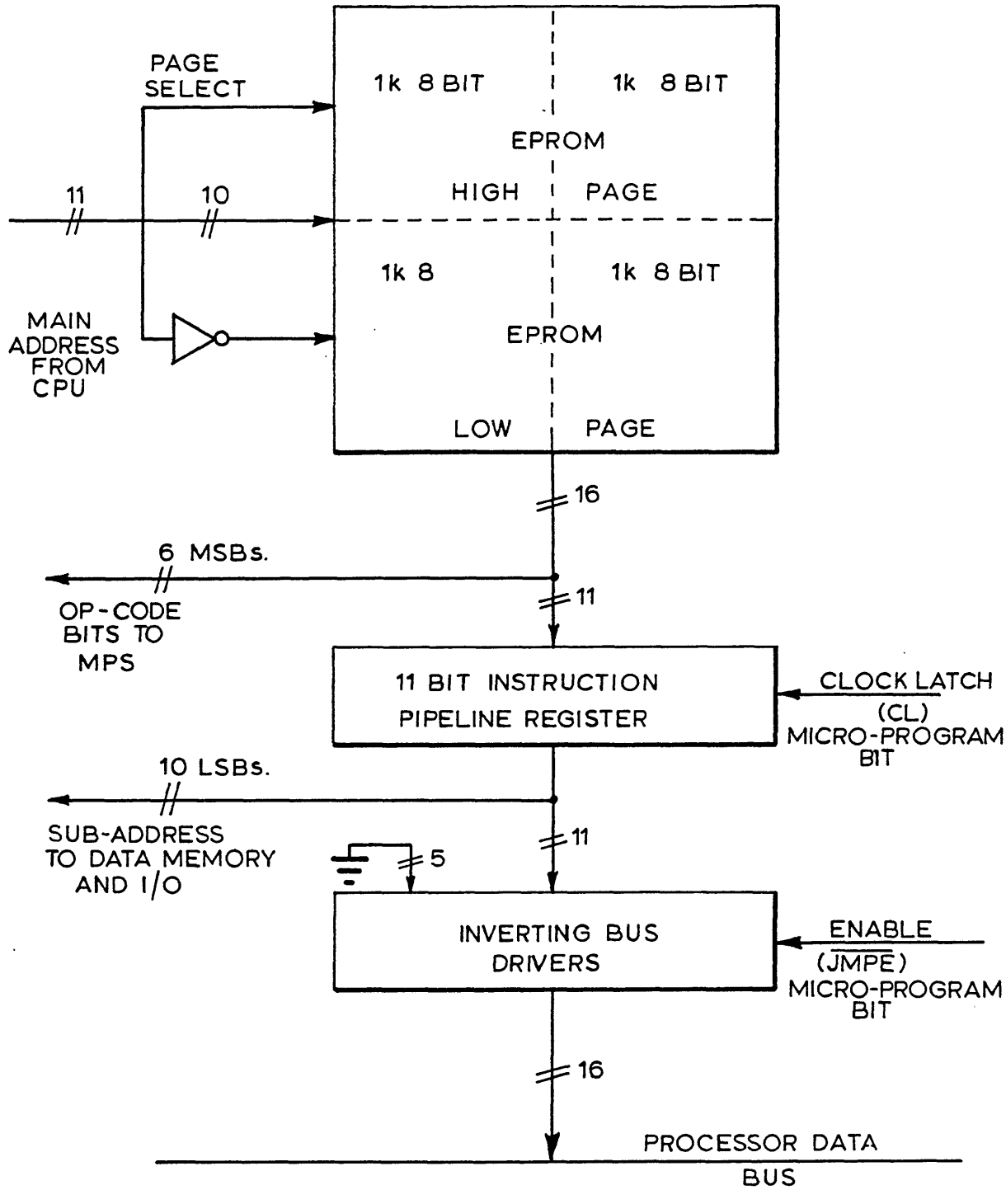


Fig. 4-5 Macro-program memory module.

is not continuous and a fresh access must be made to memory for the next instruction. The micro-code which executes these instructions therefore contains the delay required to allow for this memory access.

iii) Data bus drivers. Several of the instruction formats contain data in the LSBs of the program word which is required by the CPU. Examples are; program jump, JLP(JUMP ADDRESS) and load the loop counter, LDL (VALUE TO BE LOADED). The contents ( ), in these examples, stored in the 11 LSBs of the corresponding instruction word, are placed in the program counter or loop counter registers of the CPU respectively.

To provide for this data transfer, bus drivers are included in the memory module. When enabled by the JMPE bit of the micro-program word, the drivers connect the pipeline register outputs to the processor data bus and thus to the CPU.

4.5 Data and Constant Storage.

4.5.1 Data storage capacity.

As with program storage, an examination of protection algorithms developed using mini-computers, provides an insight into the data storage requirements of the dedicated relay processor. It is found that these requirements are modest, typically less than 200 memory locations. The data comprises measured input quantities,

processed values and constants.

4.5.2 Memory organisation.

In the majority of general purpose digital computers, having random access, read/write memory (RAM) program instructions, data and constants are generally stored contiguously within a common memory.

Clearly, since program storage in the protection processor is implemented by ROMs, an additional area of memory is required to cater for the changing data used during computations. This memory must be accessible to the CPU for both reading and writing. Semiconductor RAM devices[53] are therefore required. Further, since the data memory requirements are limited, a complete seperation of the program and data storage areas for address purposes provides several advantages.

Conventional computer architectures employ a single program counter and a common address bus to access program information and data stored in memory. In this configuration, the generation of addresses and control of the program counter are time consuming overheads.

The protection processor architecture provides an independent address bus, the sub-address bus of Fig.4.1, which is exclusively used to access the data/constant memory and I/O ports. Addresses are supplied to the memory via this bus from the 10 LSBs of the appropriate instructions.

1024 locations of storage or I/O port can thus be directly addressed.

This approach avoids the calculations which are associated with indirect, indexed or relative addressing techniques employed to provide an adequate address range within a 16 bit (or longer) instruction word. It also eliminates the need to store and reload the program counter, which occurs in common memory systems whenever a program instruction address is replaced by a data memory reference address and vice-versa. A considerable improvement in the processor performance results.

4.5.3 The data memory module.

The 10 bit sub-address bus gives an address range of 0-1023 locations which must specify both data/ constant memory and the I/O ports of the processor. Table 4.3 shows the address allocation used, from which it is seen that a maximum of 768 data or constant memory locations can be addressed. These locations consist of a combination of RAM data storage and PROM constant storage.

The memory module, shown in Fig. 4.6 is organised to provide 512 words of RAM memory with sockets to accomodate a 256 word bank of PROM memory.

	SA9									SAO	
0	0	0	0	0	0	0	0	0	0	0	RAM
511	0	1	1	1	1	1	1	1	1	1	
512	1	0	0	0	0	0	0	0	0	0	ROM
767	1	0	1	1	1	1	1	1	1	1	
768	1	1	0	0	0	0	0	0	0	0	I/O PORTS OR MEMORY EXTENSION
1023	1	1	1	1	1	1	1	1	1	1	

Table 4.3 Sub-address allocation for data/constant memory and I/O ports.

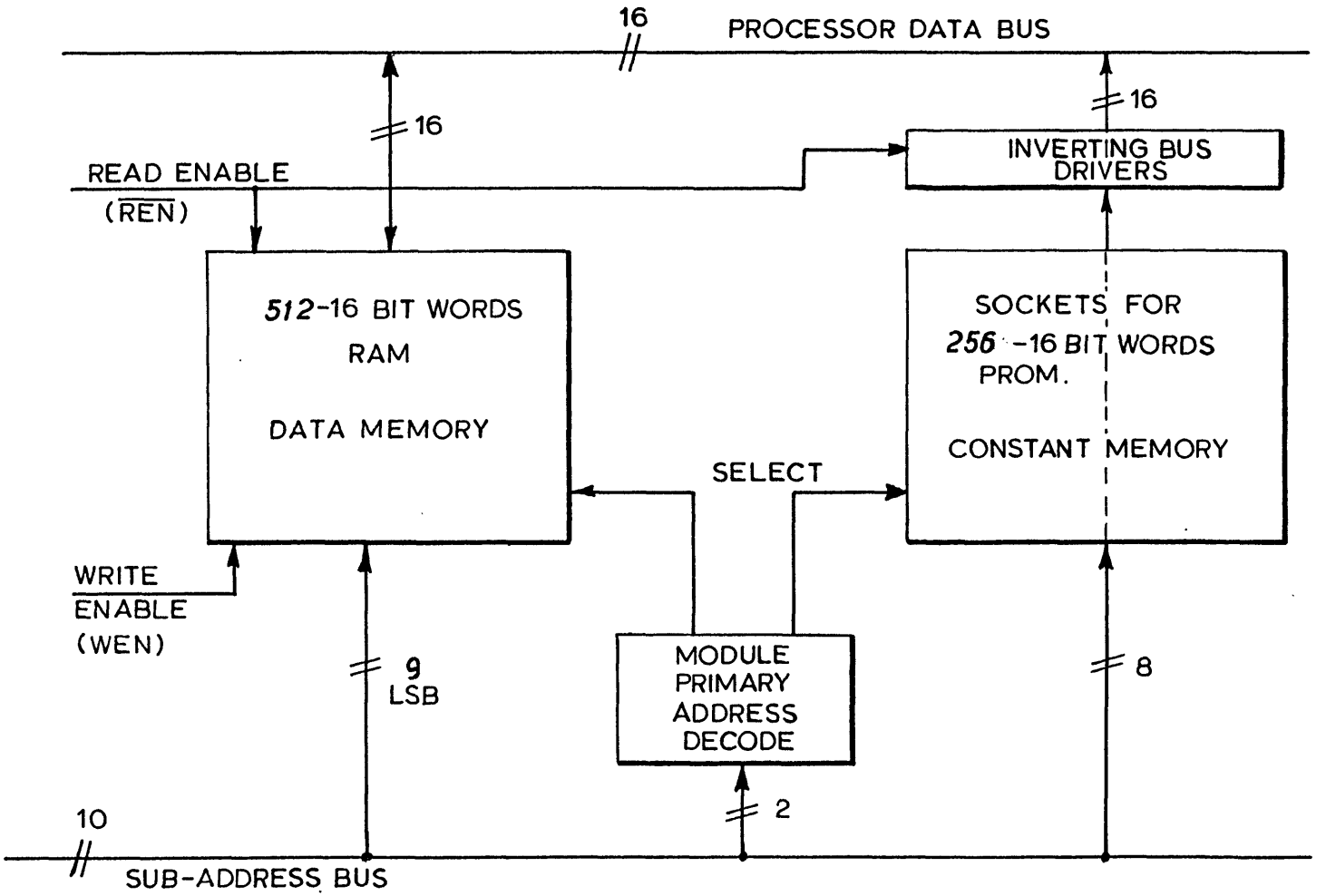


Fig. 4.6 Data-constant memory module.

Primary address decoding for the module responds to the 2 MSBs of the memory address and selects either the RAM or ROM areas of storage. During output port operations the decoder disables the memory module.

Reading and writing control of the memory is achieved by the "read enable" (REN) and "write enable" (WEN)micro-instruction bits respectively. Micro-program sequences which issue these commands contain a delay to allow for the set-up and access time of the MOS RAM devices which are used to provide a high storage density. A read or write cycle takes 600ns.

It is obvious that macro-instructions which involve a a memory write cycle can only reference the RAM area of the module (or an output port). This constraint must be observed by the relay application software.

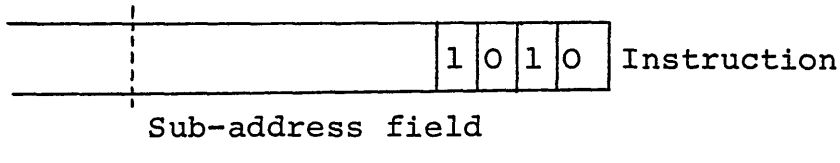
Inverting bus drivers are included to interface the constant memory outputs and the "active low" processor data bus. The inversion provided by these drivers simplifies the relay programming task, in that it allows a user to specify constants, stored in the PROMs, as conventional i.e. "active high" values.

4.5.4 The masking memory.

The storage provided by the masking memory module is not directly accessible to the relay user. However, in view of its function as a pseudo "constant" store, it is described here.

The memory consists of 2 ROMs, containing 16 pre-programmed words which are used to mask individual bits of the CPU accumulator data for bit testing purposes. Its address is derived from the sub-address bus. During execution of the bit testing instructions i.e. skip if bit set (SBS) or skip if bit clear (SBC), the memory is enabled and supplies a 16 bit mask to the CPU via the data bus. The 4 LSBs of the instruction sub-address field indicate which accumulator bit is to be tested. By placing these 4 bits on the sub-address bus the appropriate mask is obtained from the memory. For example:

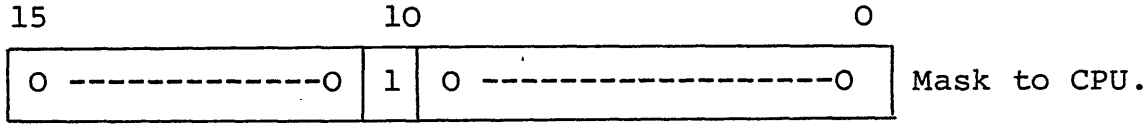
Skip if bit 10 is set. SBS 10



↓

Masking Memory

↓



Methods of bit testing which employ the carry output of the CPU involve cumbersome accumulator data shifting, (and subsequent restoration), to move the bit which is to be tested to the carry output. The masking

memory provides a rapid and efficient parallel method of performing bit testing operations.

4.6 Processor Communication.

4.6.1 The input-output task.

To fulfil its design function within a complete relay, the protection processor must be able to communicate with the data acquisition interface and circuit breaker control equipment. In addition to these basic requirements, a further general purpose data output facility must be provided if many of the valuable features of a digital relay are to be fully exploited. Performance monitoring, diagnostic reporting and alarm signalling are examples of relay data output. In sophisticated protection schemes, employing integrated back up protection and dedicated digital main protection, described in Chapter 2, this communication facility is essential.

These communication tasks are performed by 3 modules of the processor.

4.6.2 Data input.

The data input port module provides an interface between the processor data store, and the output buffer of the data acquisition unit.

The port, shown in Fig.4.7, comprises a 16 bit

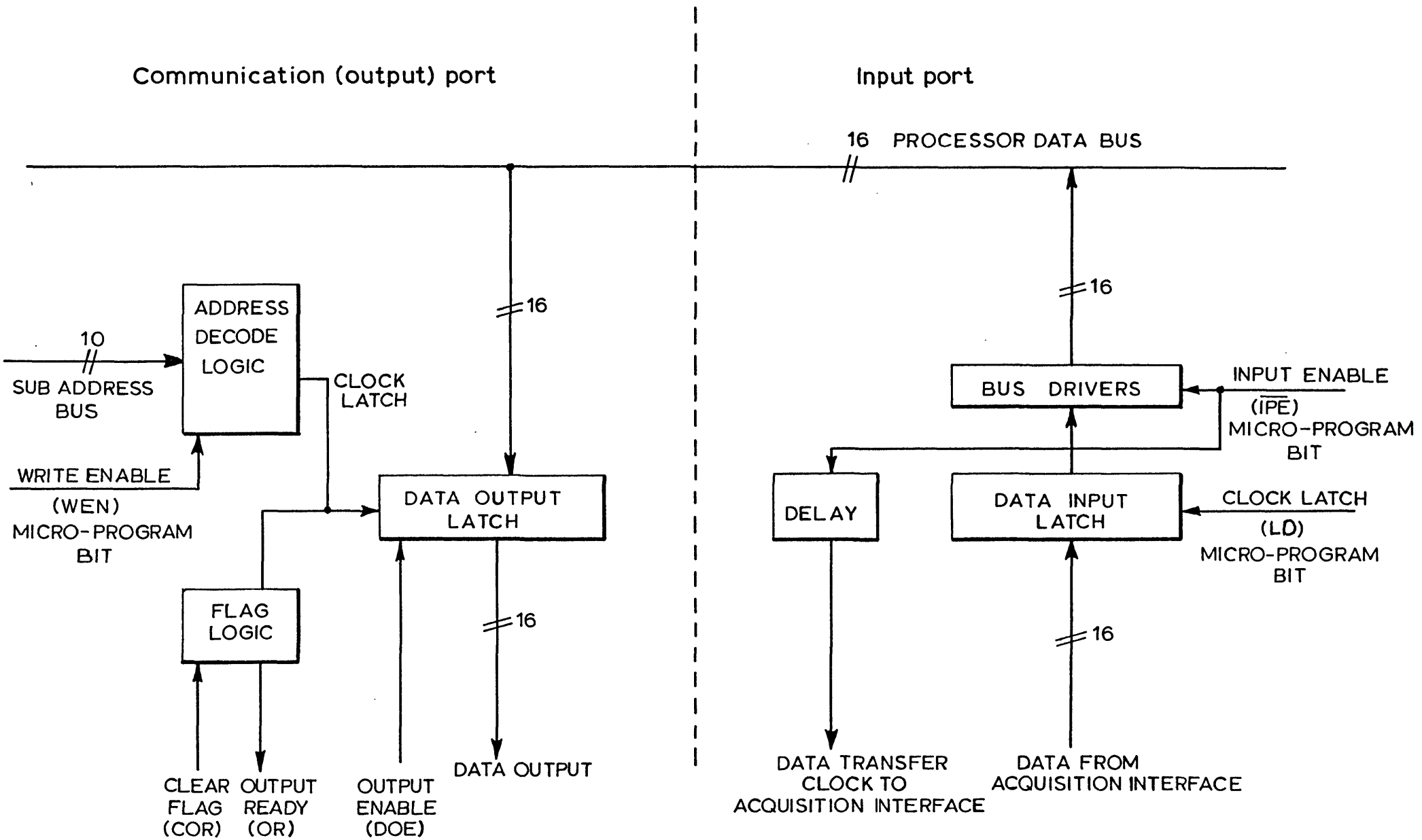


Fig. 4-7 Input-output ports.

latch which is clocked by the "LD" micro-program bit at the start of every processor instruction fetch cycle. A single word of data, from the acquisition unit, is thus repetitively input to the latch. It must be noted that clocking the processor input latch has no effect on the acquisition unit output buffer. The first data word remains static on the buffer outputs until a data transfer instruction is encountered in the processor software.

During execution of either data transfer instruction, IDM (input data to memory) or IDB (input data block), the port bus drivers are enabled by the "IPE" micro-program bit. Data present in the latch is transfered via the processor data bus to the data memory location specified by the instruction. At the completion of each word transfer, the port generates a "transfer complete" signal which shifts the following word to the output of the acquisition unit buffer store.

The control "data ready flag" of the acquisition unit is not directly used by the input module. This flag is connected to bit 9 of the processor I bus, and its status is tested by the WFS (wait until flag set) instruction. The WFS instruction effectively halts the processor operation until the data ready flag is set, whereupon normal program execution is restored. A typical data input routine is:

```
WFS        Wait here until flag set.
IDM  1     Input first data word
           to memory location 1.
IDM  15    Second data word to memory
  :        location 15.
  :            :
  :
  :        continue with program
```

4.6.3 Data output.

Several possible applications of the general purpose 16 bit data output port are discussed in section 4.6.1. Fig. 4.7 shows the output port module, which consists of a single 16 bit word storage register.

From the software view-point, the port is regarded as a data memory location $1777_{(8)}$. The output register is thus loaded with the contents of the processor accumulator by the store instruction i.e. STA 1777.

The communication structure, which is most applicable to a digital protection scheme employing dedicated main equipments with a supervisory, back up/monitoring computer, has yet to be defined. Inter-processor interfaces are very dependent upon the final overall configuration adopted. For this reason the output interface of the relay communication port has been designed for maximum simplicity. This ensures that the port output can be readily adapted to provided compatibility with a wide range

of data bus structures and transfer protocols.

The 16 bit TTL logic level data interface is controlled by a single data output enable (DOE) signal. When DOE is "low" (logical 0) the data outputs are placed in a high impedance "OFF" condition. Multiple relay outputs may thus be connected to a common data bus. By providing encoded DOE commands, a supervisory computer is able to communicate exclusively with individual relays.

An output ready flag (OR) is included in the interface. This flag is set whenever the processor stores data in the output register, and forms a service request to supervisory equipment. The flag may only be cleared by the external signal (COR). Since the output function is of secondary importance to the main protection operation of the relay, no provision is made to test the status of the OR flag within the processor. It is therefore incumbant upon the supervisory machine to ensure that data transfer is completed before the output register is again loaded by the processor.

4.6.4 Circuit breaker control.

Circuit breaker control signals form the primary output of the relay. This output is catered for by the breaker control module, Fig.4.8. As with the communication port, the module is regarded as a location of data memory. Data is transfered to the 16 bit register within the module

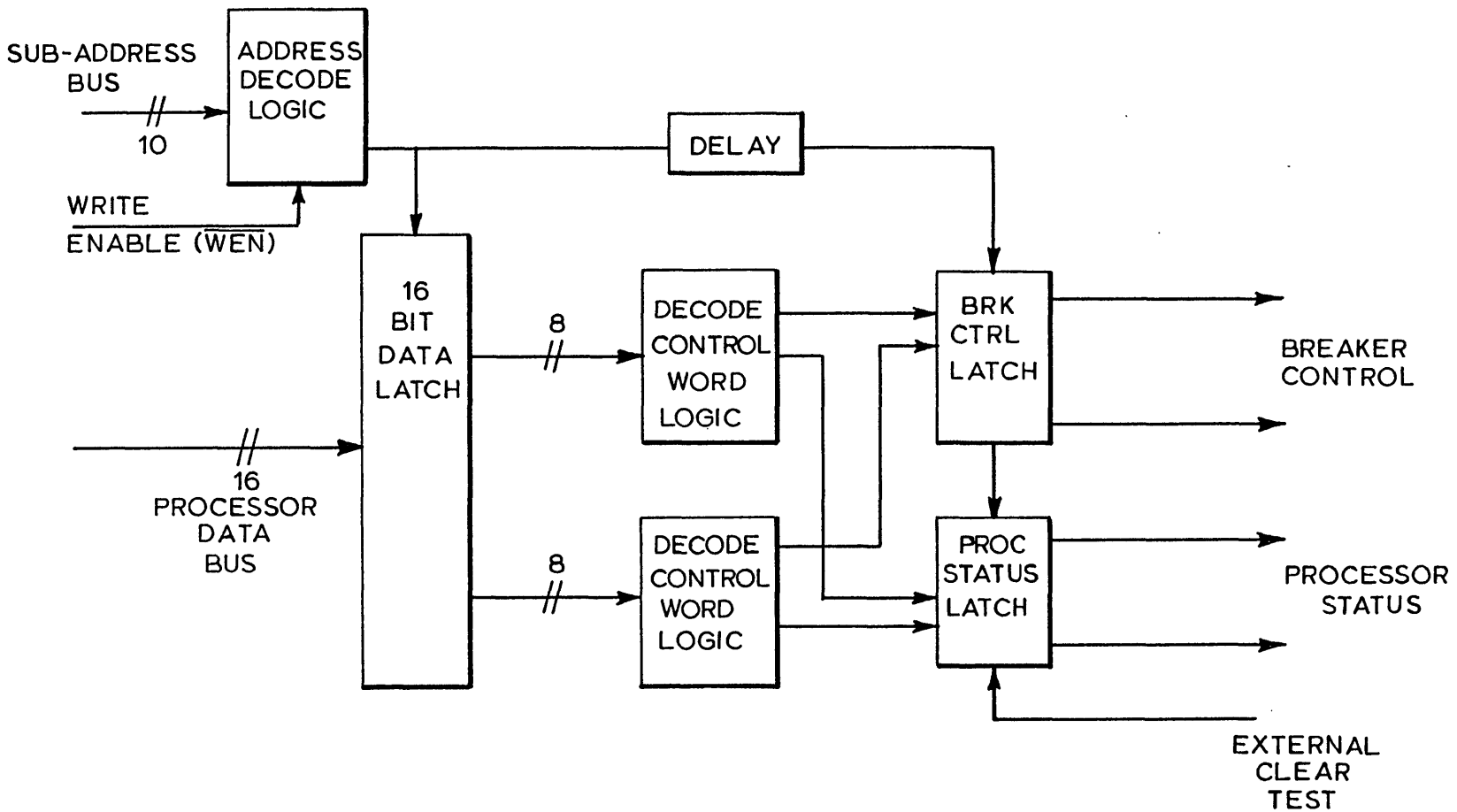


Fig. 4-8 Breaker control port.

by the memory reference instruction STA, using the address $1477_{(8)}$ i.e. STA 1477.

At the completion of each iteration of the fault detection algorithm the decision logic of the application software determines the appropriate status of the circuit breaker. Depending upon the outcome of this process, one of two possible valid control words must be stored in the module register, corresponding to the conditions:

1) TRIP - RELAY HEALTHY
2) NO TRIP - RELAY HEALTHY

All non valid words are interpreted by decoding logic within the module to mean:

NO TRIP - RELAY FAILED

The two words which define valid conditions are:

0101..... 0101 TRIP - RELAY HEALTHY
1010.....1010 NO TRIP - RELAY HEALTHY

These formats are used as they satisfy two criteria which improve the reliability of this vital processor function.

i) the control words are numerically a maximum distance from each other and from the "all zeroes" or "all ones" states, which may occur as the result of a processor fault.

ii) if it is assumed that the probability of error is equal

for each bit within the word, then the probability of errors causing the transformation of one valid format into the other is at a minimum for complementary words. The possibility of breaker control malfunction is therefore reduced.

Any erroneous data which is stored in the control register due to hardware or software failure, will have no effect on the current breaker status. A failure indication will, however, be set in the relay status latch to initiate external action, e.g. back up protection and operator alarms.

Following each data input to the control register, the decision of the decoding logic is stored in the breaker status and relay status latches of the module. Control outputs provided by these latches are 2 bit complementary logic levels. These 2 bit signals allow a measure of error detection by equipment external to the relay.

The truth table for the control outputs of the relay is:

CONDITION	BREAKER STATUS		RELAY STATUS	
	Q_B	$\overline{Q}_B$	Q_R	$\overline{Q}_R$
NO-TRIP — RELAY HEALTHY	1	0	0	1
TRIP — RELAY HEALTHY	0	1	0	1
EITHER STATE — RELAY FAULT	unchanged		1	0

A provision is made to externally force the relay status output to the "failure" state, by either a manual input or automatically by a monitoring computer.

If the relay is operating correctly the status should revert to the "normal" condition at the completion of the program iteration following the removal of the external input. This change of state can be confirmed by test equipment or by the remote computer.

The detection of relay "stuck" failure modes is thus possible. Such failures in digital equipment are caused by faulty hardware clock operation or software malfunction, e.g. program corruption resulting in an endless loop. In on-line applications, such as protective relaying, which do not require continuous control action, undetected failures of this type are a serious hazard.

4.7 Processor Timing.

4.7.1 Processor clock functions.

The primary function of the clock is to control the micro-instruction "fetch" and "execute" cycles. To achieve this, clock pulses are supplied to the CPU and MPS modules of the processor. The characteristics of these clock pulses and corresponding processor actions, are shown in Fig.4.9. From this figure it is seen that a complete micro-cycle, i.e. fetch and execute one micro-instruction, takes 200 ns.

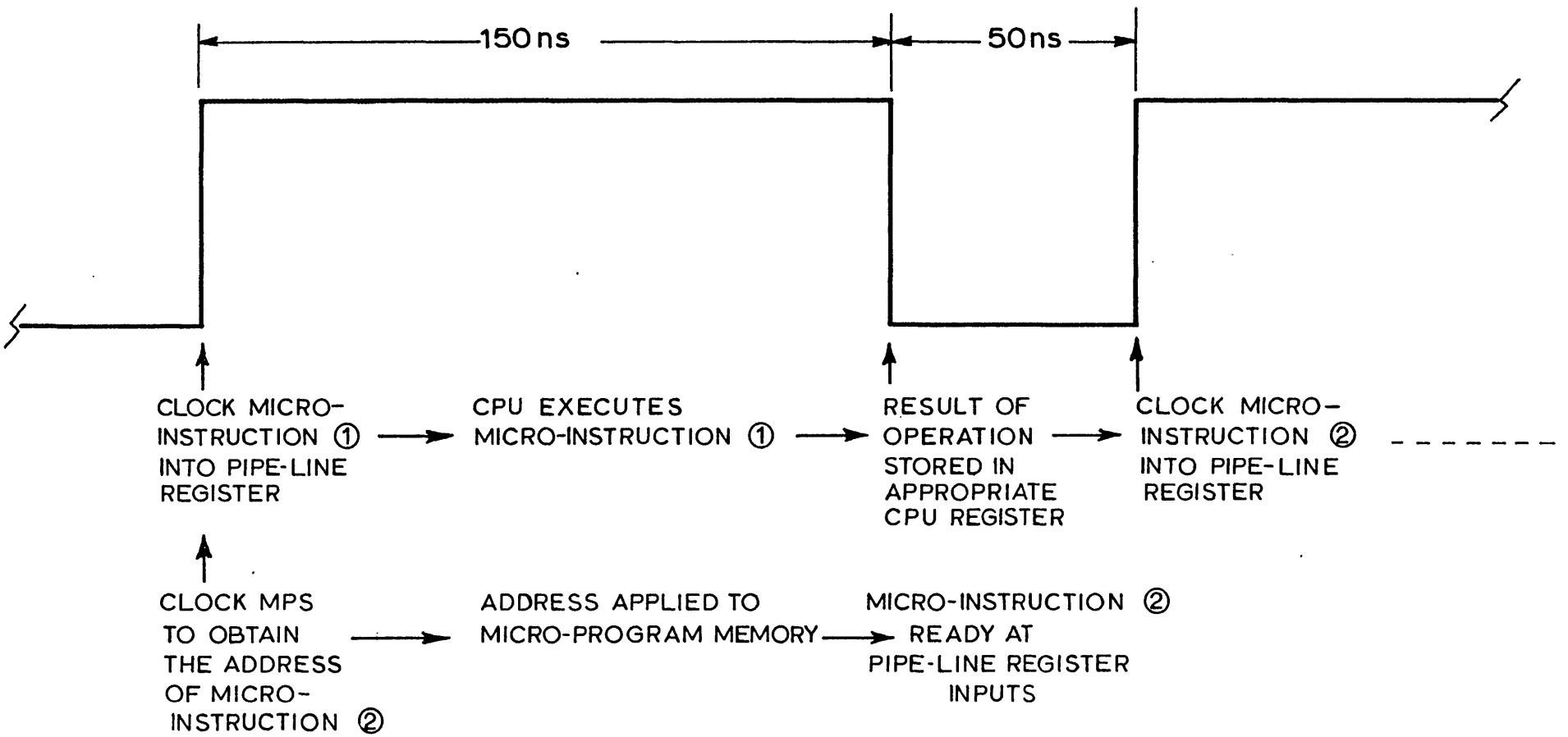


Fig. 4·9 Processor clock waveform and timing.

Micro-cycle timing is dictated by the fundamental propagation delays and data set-up requirements of the processor components[54]. The most significant of these are:

Propagation delay of micro-instruction pipe line registers.	30ns
Set-up and hold time for CPU control buses.	80ns
Minimum clock pulse width for CPU.	33ns
	143ns

A minimum cycle time of 143 ns is therefore necessary. The 200ns cycle used for the processor provides an ample margin to accomodate worst case tolerances for these delays, and thus ensures reliable operation.

4.7.2 Start-up sequence.

In common with many on-line control applications, it is important that the digital relay has a stable and predictable start-up performance following initial switch on, or transient disturbance of its power supply.

To achieve this a "restart" clock sequence is used. This sequence consists of a "power on clear" (POC) pulse during which period a single clock cycle is supplied to the processor. The master clock is inhibited for the duration of the restart sequence.

The effect of this sequence is to force the MPS

to recommence micro-instruction execution from location 0 of the micro-program store. This location contains the start of an initialisation routine. When the restart sequence ends and the master processor clock is restored, the initialisation routine causes macro-instruction execution to begin from location 0 of the main program store.

By providing appropriate software, starting at location 0 in the application program, the operation of the relay following power-up can be precisely defined.

4.7.3 The clock generator module.

The clock generator module provides the processor master and restart sequence, clocks.

To ensure timing stability, the processor clock is derived from a crystal controlled oscillator running at 19.6608 MHz. The oscillator output is frequency divided and gated to produce the clock waveform of Fig.4.9.

A facility to halt the clock and execute either micro or macro instructions on a single step basis is included in the module. This facility greatly simplifies application software development and hardware fault location.

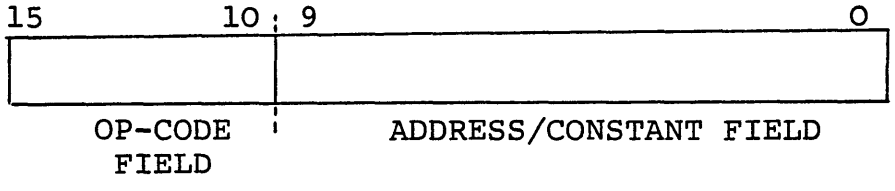
The restart sequence is initiated by logic within the clock module which detects the presence of the processor power supply. In addition to this automatic power-up feature, the logic also makes provision for a local, manual, reset/restart input. A further input enables remote equipment, e.g. a supervisory computer, to re-initialise the relay upon demand.

4.8 Software.

4.8.1 Instruction format.

The format used for the 16 bit instruction words is designed to take account of the limited memory capacity of the processor, and to provide efficient addressing for the segregated data/program memories.

Each word consists of 2 fields, a 6 bit operation code (op-code) field and a 10 bit address/constant field, as shown below:-



The op-code field which occupies the 6 MSBs of the word enables a maximum of $2^6 = 64$ instructions to be defined. Instruction groupings and definitions are described in the following sections.

The remaining 10 bits of the word form the address or constant field. This field serves various functions relating to the execution of the instruction specified by the op-code. It may contain.

a) a main program jump address
   (or subroutine address)
b) a data/constant memory address.
c) an input/output port address.
d) a constant.

For address purposes, the 10 bit field provides a maximum range of 1024 locations. This range is adequate to directly specify any location within the data/ constant memories, or the input/output ports. During execution of instructions which reference these areas, the contents of the address field are placed on the processor sub-address bus. The CPU is thus relieved of address calculation tasks which would otherwise add considerably to the micro-program complexity and hence overall instruction execution time.

To provide for program jumps in the main memory, the 10 bits of the address field are transfered when required to the CPU and thus to the program counter. The 10 bit field enables program pages of 1K words to be addressed. A further address bit is included in the relevant op-code (bit 10 of the instruction word), to define the page to which the address field refers. Again, by allowing direct addressing of the 2K program memory, this format avoids the need for time consuming CPU intervention.

Finally, certain instructions require definition in addition to that provided by the op-code. For example, the number of shifts to be performed by a multiple shift data instruction must be specified. In instructions of this type, the additional data is provided by the address/ constant field.

4.8.2 Instruction grouping.

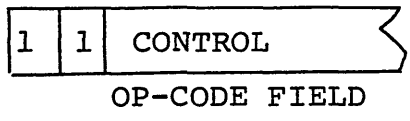
The micro-program control used by the processor, produces great flexibility in the design of the instruction set. A basic set of typical mini-computer assembler level instructions enables initial testing and evaluation of the relay to be undertaken. Additionaly, several instructions required by the specialised processor application, or which significantly reduce program complexity are included in the basic set.

To simplify the operation code format, the instruction set is sub-divided into 4 groups. The allocation of an individual instruction to a specific group is determined by the effect which the instruction has on the processor operation. Each group is defined by the 2 MSBs of the op-code. The groups are:

i) Control.

Instructions in this group directly influence the global control of the processor operation. E.g. halt program execution, restart, pause.

Format:



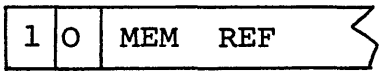
OP-CODE FIELD

ii) Memory Reference.

These instructions cause data transfer to or from a specified memory location or input/output port. In the majority of examples the data source or destination is a

register within the CPU. Memory locations refered to must form part of the data/constant memory. E.g. Load the accumulator from memory, deposit zero in memory.

Format:



OP-CODE FIELD

iii) Operate.

Operate instructions modify registers within the CPU, primarily the accumulator and loop counter, without reference to memory. E.g. clear the accumulator, negate the contents of the accumulator.

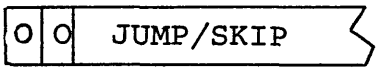
Format:

OP-CODE FIELD

iv) Jump-skip.

This group contains the instructions which alter the execution sequence of the main program. Some of the instructions directly cause a transfer of control to a new area of program, e.g. jump, jump to subroutine, whilst for others the transfer depends upon the outcome of a specified data test, e.g. skip on zero accumulator, skip on positive accumulator.

Format:



OP-CODE FIELD

These four groups define the function of the majority of the instructions. However, in some examples an instruction has a function which overlaps two groups. The groups in which instructions are placed is determined by their most significant effect on processor operation. E.g. increment the contents of a memory location and skip the next instruction if the result is zero (ISZ), is placed in the memory reference group rather than the jump/ skip group, since it modifies the contents of the location addressed.

4.8.3 Basic instruction set.

The basic processor instruction set comprises 42 instructions. Mnemonics which describe the instructions are similar to those used in currently available mini-computers, and are chosen to be self explanatory wherever possible. To illustrate aspects of the instruction set a representative sample is included here. A complete instruction list,and the micro-program sequences used to implement each instruction are given in appendix 3.

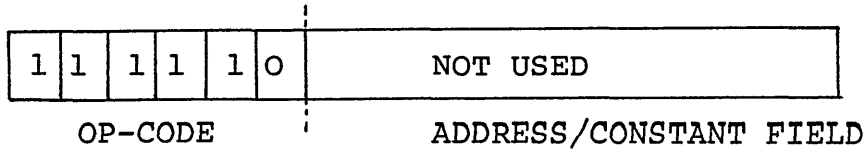
4.8.4 Instruction examples.

i) Operate group.

INSTRUCTION Halt the processor

MNEMONIC. HLT

FORMAT:

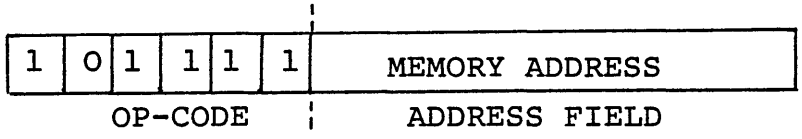


ii) Memory reference group.

INSTRUCTION. Load the accumulator from the data/ constant memory location specified by the contents of the address field.

MNEMONIC. LDA

FORMAT:

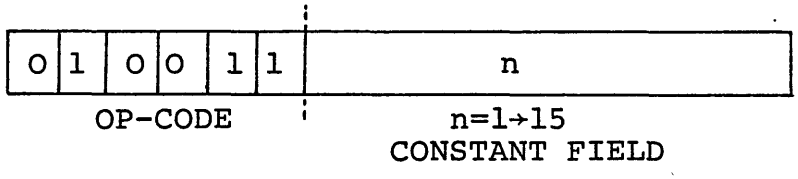


iii) Operate group.

INSTRUCTION. Shift the contents of the accumulator left n times.
(Multiple shift). n is specified by the 4 LSBs of the constant field.

MNEMONIC. MSL

FORMAT:

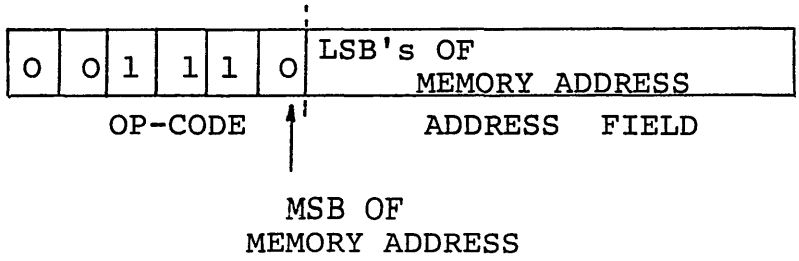


iv) Jump/skip group.

INSTRUCTION. Jump in low page. i.e. fetch the next instruction from the program memory location specified within the range 0→1023.

MNEMONIC. JLP

FORMAT:



v) Special instructions

Many of the special instructions included in the basic set are designed to simplify the processing of 10 bit analogue/digital converter data, supplied by the acquisition interface. E.g. extend the sign of the converter data to the most significant bits of the accumulator, (EXS). This instruction transforms the 10 bit 2's complement format of the converter data into a 16 bit 2's complement word which is compatible with the processor structure. By eliminating the need for a routine consisting of bit testing and addition to perform the function, the EXS instruction saves both program storage space and execution time.

4.8.5 Extended instruction set.

Provided that storage space is available in the processor micro-program control memory the basic instruction set can be extended or modified at will. Experience gained during the development of protection application software can thus be incorporated in the relay to optimise its performance.

The micro-code which implements the basic instruction set occupies 151 of the 512 available locations in the micro-program memory. Ample space therefore exists for expansion of the instruction repertoire.

Several additional instructions which provide

more sophisticated operations are included in the extended instruction set. Examples are, double precision arithmetic (i.e. 32 bit) manipulation, and a signed 16 bit multiplication which produces a 32 bit product in 19μs.

Appendix 3 contains full details of the extended instruction set.

4.8.6 Subroutines.

The "jump to subroutine" (JSR) and "return from subroutine" (RET) instructions allow for single subroutine calls only. An internal CPU register,(R7) is used to store the program counter contents, i.e. the return address, before entry is made to the subroutine. Therefore, since only one return address can be accommodated, subroutines cannot be "nested" within the processor software.

Although this organisation imposes a slight restriction on program design, it does provide a fast entry/exit for subroutines. It also avoids the complexity of address stacks, stack pointers, indirect addressing and other techniques required for nested subroutine calls.

4.8.7 Off-line support.

To facilitate development of protection processor software, off-line support aids consisting of an assembler for the processor language, and an EPROM programmer control package are essential. These aids have been designed to operate in conjunction with a host mini-computer. More

comprehensive support aids e.g. text editor, relay simulator, would ease the task of relay programming still further. Considerable scope for improvement exists in this area.

A portion of a typical relay program and its assembler listing is shown below:

```
GO
000630      LDA   000026     136026      000274     000026
000631      ADD   000030     132030      000264     000030
000632      ADD   000032     132032      000264     000032
000633      STA   000040     134040      000270     000040
000634      LDA   000000     136000      000274     000000
000635      SBS   000014     024014      000050     000014
000636      JLP   000641     034641      000071     000241
000637      LDA   000040     136040      000274     000040
000640      STA   000377     134377      000270     000377
000641      LDA   000040     136040      000274     000040
000642      SPA               020000      000040     000000
000643      NEG               072000      000164     000000
000644      STA   000040     134040      000270     000040
000645      DZM   000041     116041      000234     000041
000646      LDA   000042     136042      000274     000042
000647      STA   000041     134041      000270     000041
000650      LDA   000043     136043      000274     000043
000651      STA   000042     134042      000270     000042
000652      MSL   000002     046002      000114     000002
000653      SUM   000041     114041      000230     000041
000654      LDA   000044     136044      000274     000044
000655      STA   000043     134043      000270     000043
000656      SHL               062000      000144     000000
000657      SUM   000041     114041      000230     000041
000660      LDA   000045     136045      000274     000045
000661      STA   000044     134044      000270     000044
000662      MSL   000002     046002      000114     000002
000663      SUM   000041     114041      000230     000041
000664      LDA   000046     136046      000274     000046
000665      STA   000045     134045      000270     000045
COMMAND?
```

CHAPTER 5

RELAY APPLICATIONS

(i) GENERATOR UNBALANCED LOAD PROTECTION

5.1 Introduction

To demonstrate the operation and to validate the design of the dedicated digital relay, its application to specific protection tasks was studied.

For the protection functions chosen, it was considered important that, within laboratory constraints, the final digital system should provide a performance comparable with that of corresponding static analogue equipment. The major features in this respect are: i) number of signal transducers required. ii) operating range. iii) trip characteristic output. iv) provision of standard alarm outputs. v) sensitivity and accuracy of fault detection. vi) stability during normal load and fault conditions for which operation must not occur.

In addition to these basic requirements, improvements in the overall protection facilities, made possible by the use of digital techniques, were investigated. Several of the features described in Chapter 2, for example, self-test and inter-processor communication, are included in the implementations.

5.2 Generator Unbalanced Load Protection.

5.2.1 Introduction

Power system disturbances which result in unbalanced loading of the generator(s) supplying the system cause potentially harmful modifications to the electromagnetic fields within the machine(s).

In particular, the field associated with the negative sequence component of the unbalanced load current is of concern[55]. This field induces double frequency currents in the rotor body and windings of the generator[56]. At the elevated frequency eddy currents are greatly increased and produce considerable heating. In extreme cases the heating can cause mechanical failure of the rotor components and severe machine damage occurs.

To prevent rotor heating from reaching destructive levels, protection must be provided which will detect and disconnect severe or prolonged asymmetrical loads from the generator. Unbalanced load (negative sequence) relays are used for this task.

A large modern generator represents a massive capital investment. The consequences of generator protection failure are likely to be more serious than those encountered in most other protection applications. Relays of predictable performance and high reliability are thus essential. In this respect digital equipment has much to offer.

5.3 Task Definition.

The ability of a generator to withstand unbalanced load conditions is determined by the thermal characteristics of its rotor. Since heating depends on the negative sequence reaction field and hence current, the generator withstand capability is expressed as a continuous negative sequence rating, being generally a percentage of the continuous mean i.e. normal load rating (CMR). Table 5.1 lists these ratings for various generator types, and Fig.5.1 shows their withstand characteristics.

For short rotor heating periods, i.e.up to 100 seconds, of interest during system faults, heat losses from the machine can be neglected. During this period the rotor heating law is expressed as[2]:

$$I_2^2 \, t = K \qquad 5.3.1$$

where I_2 = negative sequence current (per unit of CMR)

t = time (s)

K = constant proportional to the thermal capacity of the rotor.

Unbalanced load relay operation thus comprises two tasks. These are i) the determination of the negative sequence current magnitude, present in the generator output and ii) provision of an inverse square law trip characteristic which matches the rotor heating characteristic of expression 5.3.1.

In addition an alarm output is normally provided by the relay when a preset negative sequence current magnitude

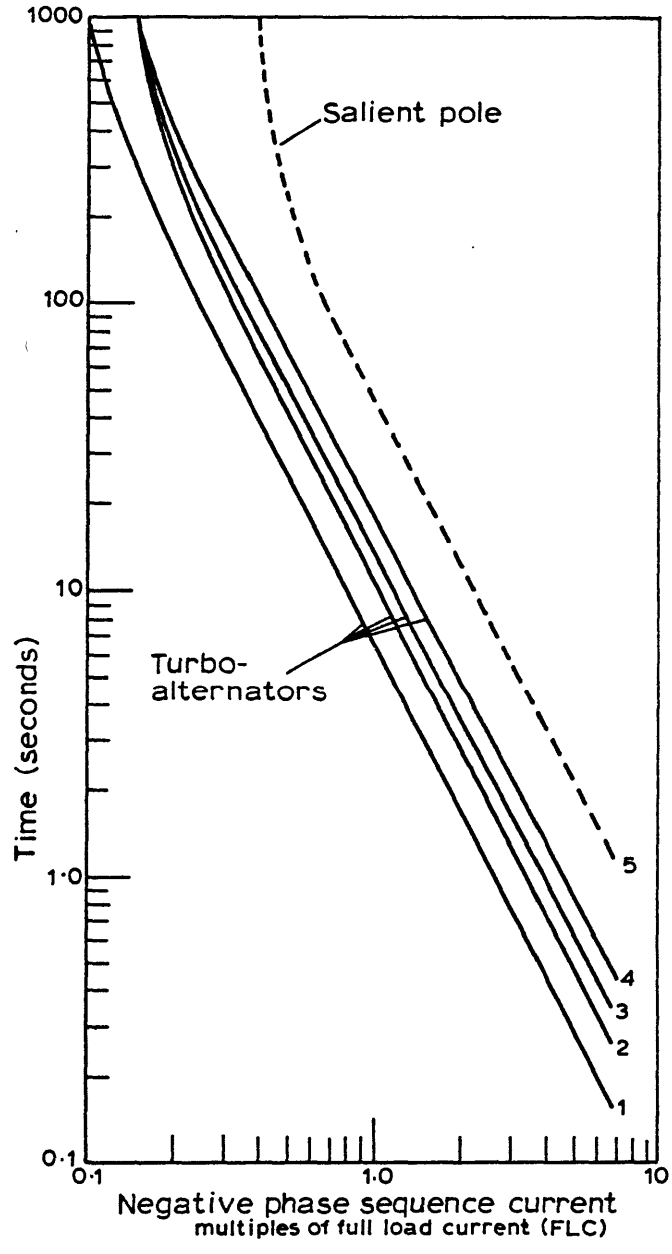


Fig. 5·1 Negative sequence current withstand for various generator types.

Curve number	Cooling	Continuous Neg. Seq. rating	$I_2^2 t$ value
1	Direct Hydrogen 2bar	10 %FLC	7
2	Conventional Hydrogen 2bar	15 "	12
3	Conventional Hydrogen 1bar	15 "	15
4	Air or Hydrogen 0·03 bar	15 "	20
5	Air	40 "	60

Table 5·1 Negative sequence ratings of various generator types.

is exceeded. This alarm allows time for plant operators to locate and rectify the cause of the load imbalance before tripping occurs. The alarm usually incorporates a definite time delay to prevent incorrect operation during transient conditions[2].

Fig.5.2 shows a typical analogue unbalanced load relay. The transformers T_1,T_2 remove zero sequence components from the primary phase current measurements after which the filter network Z_1, Z_2 is employed to detect the negative sequence component. The filter output, V(x,y), described by the phasor diagrams of Fig.5.3 (a)(b) then provides the control signal to the tripping and alarm elements.

5.4 Digital Relay Implementation

5.4.1 General requirements of protection algorithms

In developing algorithms for digital protection functions several important constraints must be considered.

Clearly it is essential that the final algorithm should provide satisfactory fault detection and tripping characteristics. However, it must be borne in mind that hardware limitations exist both in mini-computers, and more especially in microprocessors. These limitations take the form of operation speed, storage space and arithmetic restrictions.

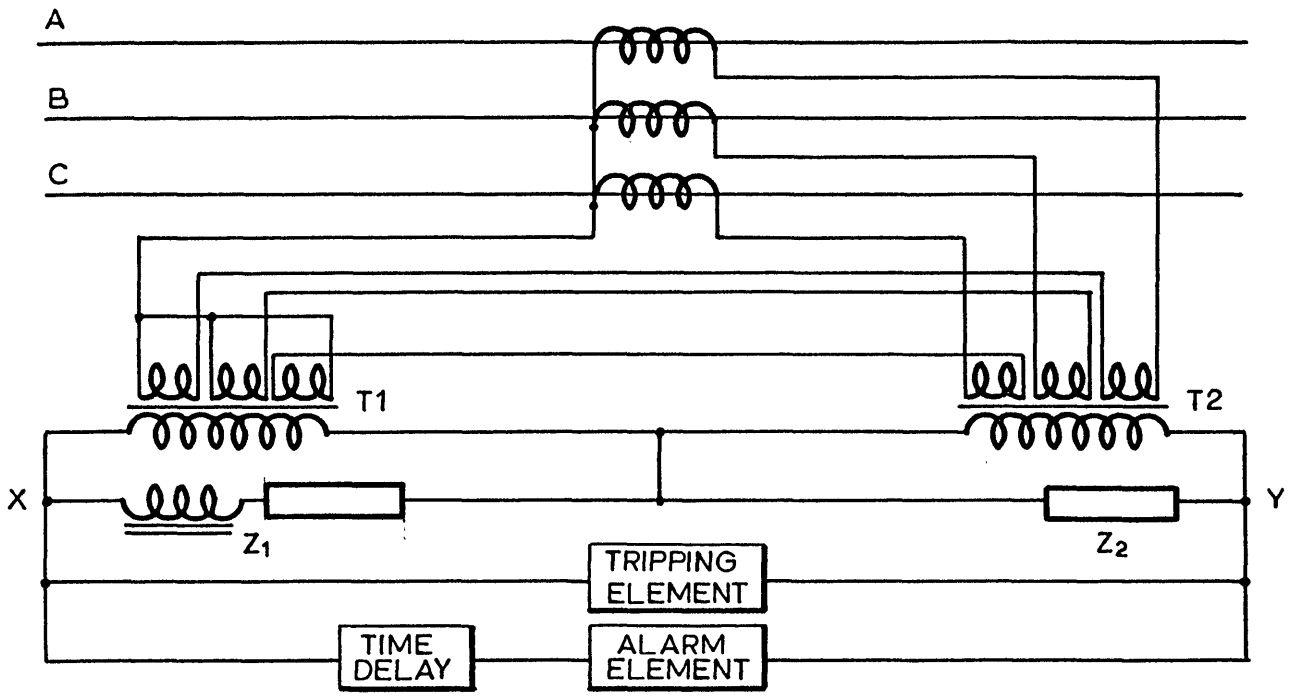


Fig. 5·2 Simplified analogue unbalanced load relay.

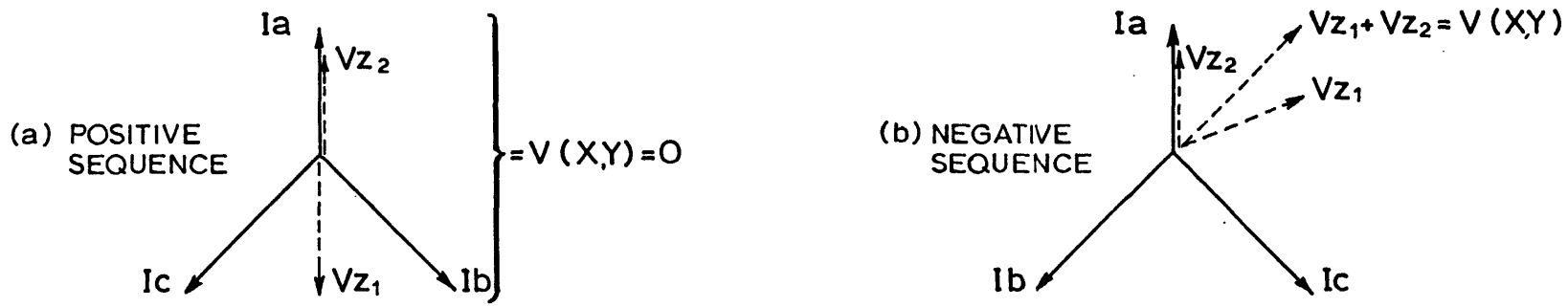


Fig. 5·3 Relationship of voltage V (X,Y) to sequence components of phase currents.

Particular care must be exercised in data representation during processing. Even with the 16 bit word length used in the dedicated relay, precautions to minimise overflow and truncation errors are essential. Integer arithmetic techniques[57] allow very efficient processing, suitable for high speed real time tasks such as protection. Floating point representation must be avoided, since without costly additional hardware it requires complex software which is prohibitive from the viewpoint of operating time and storage space.

A similar situation exists with multiply and divide arithmetic operators. The micro-program facility of the digital relay processor allows a far more efficient realisation of these functions than is possible by software routines, but they still remain the most time consuming instructions. Algorithms involving numerous multiplications or divisions should therefore be avoided whenever possible. In some cases, e.g. for data filtering, multiplication or division by constant terms can be replaced by multiple shift operations i.e. multiply or divide by 2^n where n is the number of shifts. The sensitivity of the algorithm to these approximations, and an analysis of the errors which arise, then become part of the design considerations.

In general protection performance improvements which are achieved by the use of sophisticated algorithms,

must be carefully weighed against increased software complexity.

5.4.2 Unbalanced load protection algorithm.

The digital unbalanced load relay algorithm was designed to take account of the constraints outlined in 5.4.1. Integer arithmetic was employed throughout the algorithm which comprises 5 sections:

i) Data input and filtering.
ii) Negative sequence current detection.
iii) Rectification, smoothing and R.M.S.evaluation.
iv) Trip characteristic calculation.
v) Supporting routines, including real time clock, self test, alarm and data output, and circuit breaker control functions.

5.4.3 Data input and filtering

This section of the program fulfils the functions of data input from the data acquisition system, and data filtering.

The input routine cyclicly tests the status of the acquisition system data ready flag. When the flag "set" condition is detected data transfer is initiated. The data block assembled in the acquisition system output buffer, is transfered directly to locations within the protection processor data store. For unbalanced load

protection each data block consists of five words. These are:

i) Program control data word. (P.C.W)

ii) Phase a current measurement (ia)

iii) Phase b current measurement (ib)

iv) Phase c current measurement (ic)

v) Reference check value (Vr)

Converted analogue values, i.e. inputs ii-v, are transfered as 10 bit 2's complement words. Input routine processing extends the sign bit of this data to produce a 16 bit format compatible with the protection processor structure.

Additionally the input routine synchronises the protection processor operation with the data collection cycle of the acquisition system. At the commencement of each acquisition cycle, determined by the sampling clock, the output buffer of the unit is cleared. Following initialisation of the equipment it is possible that this clearance may occur during the data transfer sequence. Data errors would thus arise. By discarding the first data transfer, the input routine eliminates this hazard. Subsequent data transfer and processing only occurs immediately after the completion of an acquisition cycle.

Following data transfer, bit 15 of the P.C.W. is tested to determine if the phase current values are to be

filtered. This option allows either raw data samples or their filtered version to be used in later fault detection routines.

Analogue filtering of the phase current signals is restricted to that required to prevent aliasing during sampling (Section 5.5.1).

The removal from the current signals of unwanted harmonics, switching transients and data acquisition noise, entails the use of filters with a cut-off frequency close to the fundamental, i.e. 50 Hz. Phase distortion of the fundamental caused by such filters will be considerable. A large phase shift in the current waveforms is acceptable, provided that it is equal for each phase, since subsequent processing requires that the relative phase relationships of the input quantities are maintained. For this reason the phase characteristics of the filters must be closely matched. Analogue filters which satisfy this condition, particularly for long service periods, will be costly. Digital filters which offer highly stable and repeatable characteristics are thus used.

Each phase current filter is a second order lowpass Butterworth function, with a cut-off frequency of 90 Hz.

Using Z transforms the transfer function $H_{(z)}$, of a second order recursive filter may be defined as:

$$H_{(z)} = \frac{1 + A_1 Z^{-1} + A_2 Z^{-2}}{1 + B_1 Z^{-1} + B_2 Z^{-2}} \qquad (5.4.1)$$

The filter output $Y_{(z)}$ which corresponds to an input sequence $X_{(z)}$ is given by:

$$Y_{(z)} = H_{(z)} X_{(z)} \qquad (5.4.2)$$

Substituting for $H_{(z)}$:

$$Y_{(z)} = \frac{1 + A_1 Z^{-1} + A_2 Z^{-2}}{1 + B_1 Z^{-1} + B_2 Z^{-2}} X_{(z)} \qquad (5.4.3)$$

and expanding gives:

$$Y_{(z)}(1 + B_1Z^{-1} + B_2Z^{-2}) = (1 + A_1Z^{-1} + A_2Z^{-2}) X_{(z)} \qquad (5.4.4)$$

thus, the output $Y_{(n)}$ corresponding to the sampling instant n is:

$$Y_{(n)} = X_{(n)} + A_1X_{(n-1)} + A_2X_{(n-2)} - B_1 Y_{(n-1)} - B_2 Y_{(n-2)} \qquad (5.4.5)$$

A filter design program[58] was employed to compute the coefficients A_1, A_2, B_1, B_2 for the required 90 Hz cut-off, when the sequence $X_{(z)}$ in 5.4.2 was derived

by sampling the input quantities at 600Hz (Section 5.4.4). To produce unity gain in the passband the design program also provides a filter gain coefficient A_o. This is used as a scaling factor for either the input or output sequences, i.e.

$$Y_{(n)} = A_o(X_{(n)} + A_1X_{(n-1)} + A_2X_{(n-2)}) - B_1Y_{(n-1)} - B_2Y_{(n-2)} \quad (5.4.6)$$

A complete block diagram of the filter is shown in Fig. 5.4, for which the computed coefficients are:

$$A_o = 0.131106$$

$$A_1 = 2.0$$

$$A_2 = 1.0$$

$$B_1 = -0.747789$$

$$B_2 = 0.272215$$

Implementation of the filter within the relay software is achieved without multiplication. The coefficients are realised by multiple shifting, addition and subtraction operations, giving:

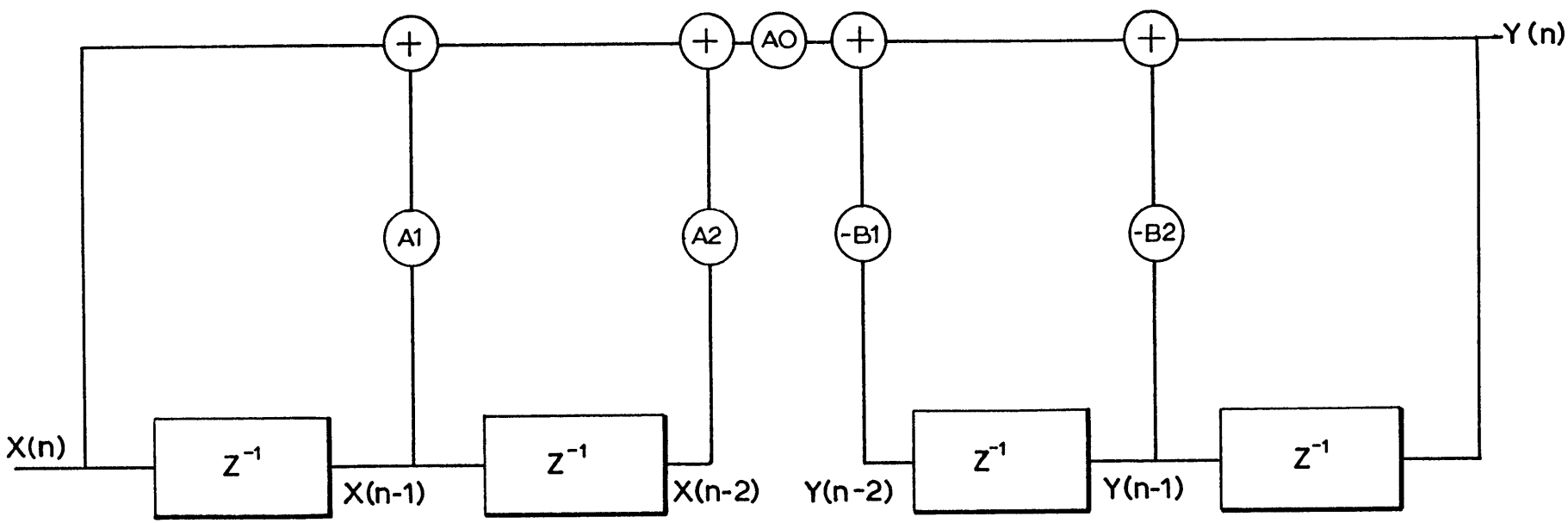


Fig. 5·4 Second order digital filter in linear form.

$$A_o = (\frac{1}{8} + \frac{1}{128}) = 0.132812$$

$$A_1 = 2.0$$

$$A_2 = 1.0$$

$$B_1 = -(1 - \frac{1}{4} - \frac{1}{512}) = -0.748047$$

$$B_2 = (\frac{1}{4} + \frac{1}{32} - \frac{1}{128}) = 0.273437$$

Using these approximations, expression 5.4.6 becomes:

$$Y_{(n)} = \frac{X}{8} + \frac{X}{128} + Y_{(n-1)} - \frac{Y_{(n-1)}}{4} - \frac{Y_{(n-1)}}{512} - \frac{Y_{(n-2)}}{4} - \frac{Y_{(n-2)}}{32} + \frac{Y_{(n-2)}}{128} \qquad (5.4.7)$$

where $X = (X_{(n)} + 2X_{(n-1)} + X_{(n-2)})$

To determine the validity of the approximations an off-line simulation of the filter was necessary. Fig.5.5 illustrates the effects of the coefficient approximations on the gain characteristics of the filter. It can be seen that the approximations cause no significant variation in performance. Filter stability[59] is also unimpaired, as shown by the impulse response characteristics, Fig.5.6.

Expression 5.4.7 is therefore used within the relay software to implement the required filter functions.

The effectiveness of the filter in processing the relay input data is demonstrated by the oscilligrams of

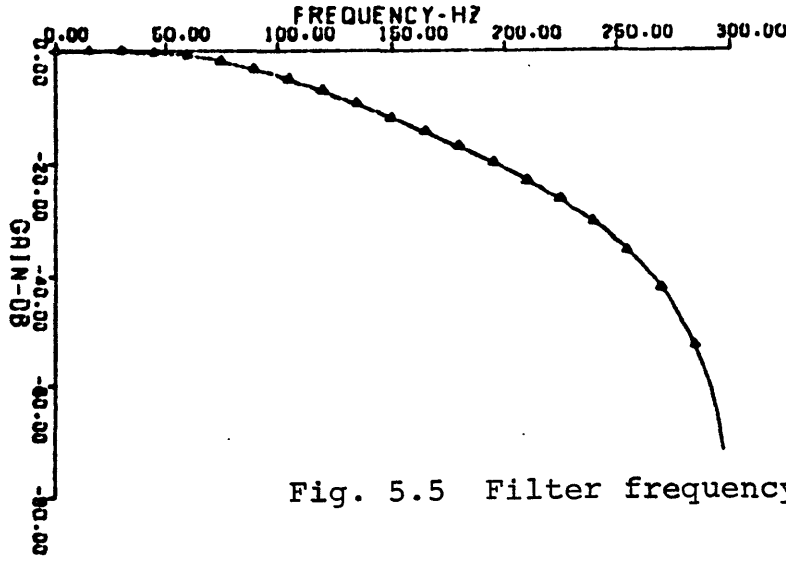


Fig. 5.5 Filter frequency response.

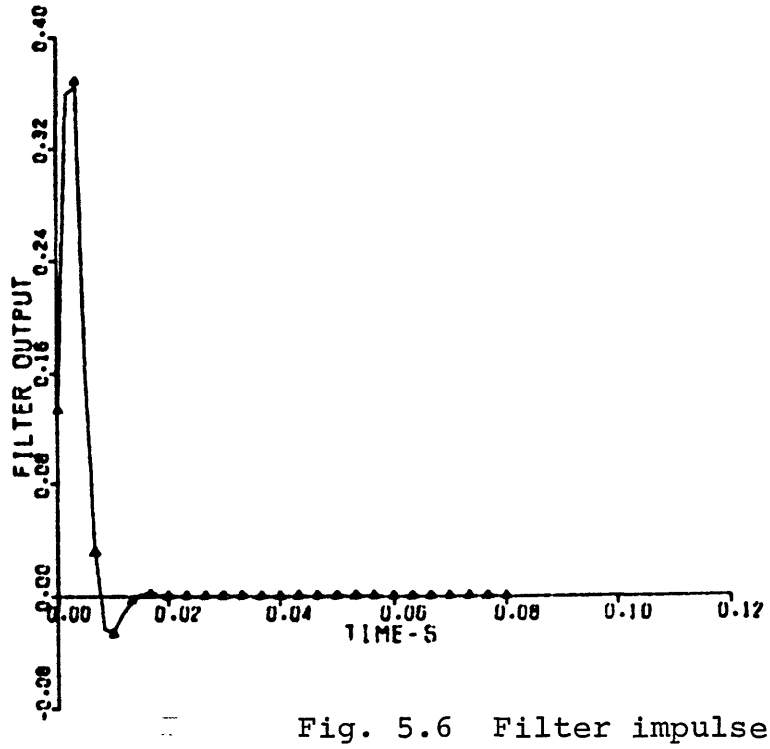


Fig. 5.6 Filter impulse response.

—— Computed coefficients.

▲ Approximate coefficients.

Fig. 5.7. In each figure the lower trace shows a phase current input during step load changes, whilst the upper trace is the corresponding filtered data. The switching transients, present in the raw data samples, are clearly removed by the filter.

5.4.4 Negative sequence current detection

The digital filter routine outputs are integer two's complement values of the 3 generator phase currents. These outputs occur at discrete time intervals, determined by the sampling rate. The currents corresponding to an instant t are $i_{a(t)}$, $i_{b(t)}$, $i_{c(t)}$. From these quantities the negative sequence component, of interest in unbalanced load protection, must then be derived.

By applying symmetrical component analysis[60] it can be shown that:

$$i_{a2(t)} = \frac{1}{3}\left(i_{a(t)} + a^2 i_{b(t)} + a i_{c(t)}\right) \qquad (5.4.8)$$

$$i_{b2(t)} = \frac{1}{3}\left(i_{b(t)} + a^2 i_{c(t)} + a i_{a(t)}\right) \qquad (5.4.9)$$

$$i_{c2(t)} = \frac{1}{3}\left(i_{c(t)} + a^2 i_{a(t)} + a i_{b(t)}\right) \qquad (5.4.10)$$

where $i_{a2(t)}$, $i_{b2(t)}$, $i_{c2(t)}$ are the instantaneous negative

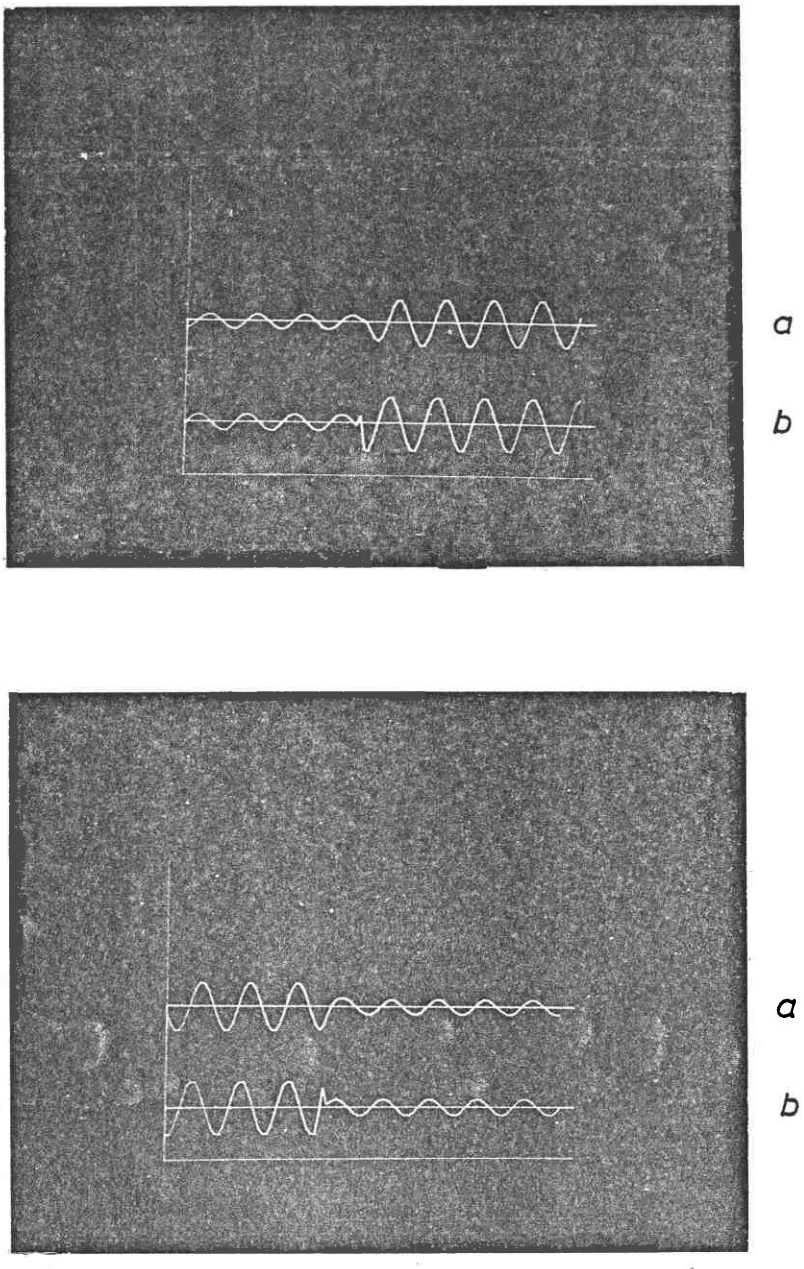


Fig. 5·7 Filtered phase current data *(a)*
Raw input data *(b)*
During step load changes of 3:1

sequence current vectors, and a is the complex operator $(-\frac{1}{2} + j\frac{\sqrt{3}}{2})$.

It is immediately apparent that a direct realisation of the above expressions would require complex arithmetic routines. The extensive processing required by these routines makes them unsuitable for on-line microprocessor implementation. An alternative approach must therefore be adopted.

In polar notation the operator a may be defined as $1 \angle 120^{\circ}$. The product term: $a\, i_{(t)}$, thus represents an anticlockwise rotation by 120° ($\frac{2\pi}{3}$ rad) of the phasor $i_{(t)}$ If $i_{(t)}$ is defined as:

$$i_{(t)} = I \sin \omega t \qquad (5.4.11)$$

then:

$$ai_{(t)} = I \sin(\omega t + \frac{2\pi}{3}) \qquad (5.4.12)$$

Provided that a sinusoidal waveform is sampled at a fixed multiple, n, of its frequency then the instantaneous values are:
for the k^{th} sample

$$i_{(t+k)} = I \sin (\omega t + \frac{2k\pi}{n}) \qquad (5.4.13)$$

Therefore, by arranging the sampling frequency such that, n,

is an integer multiple of 3, the expressions 5.4.12 and 5.4.13 may be equated, giving:

$$a\ i_{(t)} = i_{(t+\frac{n}{3})} \tag{5.4.14}$$

Similarly, since $a^2 = 1\ \angle 240^\circ$ then:

$$a^2\ i_{(t)} = I \sin\ (\omega t + \frac{4\pi}{3})$$

hence:

$$a^2\ i_{(t)} = i_{(t+\frac{2n}{3})} \tag{5.4.15}$$

As the magnitudes of the phasors $i_{a2(t)}$, $i_{b2(t)}$ and $i_{c2(t)}$ are equal, it is only necessary that one be evaluated for protection purposes. Thus substitution from 5.4.14 and 5.4.15 into 5.4.8 gives:

$$i_{a_{2(t)}} = \frac{1}{3}(i_{a_{(t)}} + i_{b_{(t+\frac{2n}{3})}} + i_{c_{(t+\frac{n}{3})}}) \tag{5.4.16}$$

Expression 5.4.16 is a "forward" method for evaluating

the instantaneous magnitude of the negative sequence component $i_{a2(t)}$. As such it requires predictive algorithms to compute values for $i_{b_{(t+\frac{2n}{3})}}$ and $i_{c_{(t+\frac{n}{3})}}$ at each iteration.

To avoid the complication of predictive methods 5.4.16 is re-written in its "backward" form, i.e.:

$$i_{a_{2(t-\frac{2n}{3})}} = \frac{1}{3}(i_{a_{(t-\frac{2n}{3})}} + i_{b_{(t)}} + i_{c_{(t-\frac{n}{3})}}) \qquad (5.4.17)$$

$i_{a_{(t-\frac{2n}{3})}}$, $i_{c_{(t-\frac{n}{3})}}$ are directly available as previously stored values of $i_{a_{(t)}}$ and $i_{c_{(t)}}$.

The use of expression 5.4.17 introduces a delay factor, $(t-\frac{2n}{3})$ in the evaluation of i_{a_2} at each sampling instant. However, since high speed of operation is not an important criterion for unbalanced load protection, this does not constitute a serious disadvantage. This method is therefore employed in the relay.

Four possible sampling rates provided by the data acquisition system fulfil the requirement of being multiples of 3 samples per cycle, i.e. 3, 6, 12 and 24 samples per cycle (s.c.).

The lower values, i.e. 3,6, s.c. are not suitable because to avoid aliasing at these rates, lowpass analogue filters withcut -off frequencies less than 75Hz and 150Hz

respectively are required. Phase characteristic mismatch within these filters causes substantial errors as discussed in previous sections.

A sampling rate of 12s.c., i.e. 600Hz provides ample time, 1.67 ms, for computation between successive samples. Additionally, data storage and manipulation, are reduced in comparison to that required for a 24 s.c. algorithm. For a 12 s.c. rate, 5.4.17 becomes:

$$i_{a_{2(t-8)}} = \frac{1}{3}(i_{a_{(t-8)}} + i_{b_{(t)}} + i_{c_{(t-4)}}) \qquad (5.4.18)$$

The negative sequence current I_{a2}, extracted from the measured data is a sinusoidal quantity. At this stage the division by 3 shown in expression 5.4.18 is not performed. The magnitude of I_{a2} is thus 3 times greater than its true value.

In the most severe case of load imbalance, when only a single phase is loaded, I_{a2} cannot exceed the peak value of this input current, i.e. 9 L.S.B's of the data word. There is thus no possibility of overflow

occuring, and truncation errors which may occur in the division are avoided.

5.4.5 R.M.S. evaluation

The tripping characteristic of the relay is derived from the approximate rotor heating law, expression 5.3.1, i.e:

$$I_2^2 \; t = K \tag{5.4.19}$$

I_2 in this expression is the R.M.S. value of the negative sequence current, which must be derived from the output I_{a2} of the symmetrical component routine.

Integration over half a cycle of a sinusoidal quantity, having a peak value $\hat{I}$ gives:

$$\int_0^{\pi} \hat{I} \sin\theta \, d\theta = \left[-\hat{I} \cos\theta \right]_0^{\pi}$$

$$= 2\hat{I} \tag{5.4.20}$$

If the sinusoid is "full wave rectified" prior to integration $\hat{I} \sin\theta$ becomes $\hat{I} \, |\sin\theta|$, and the limits of integration may be generalised to:

$$\int_a^{a+\pi} \hat{I} \, |\sin\theta| \, d\theta = 2\hat{I} \tag{5.4.21}$$

The R.M.S. value of the original sinusiod can thus be obtained by:

$$I_{R.M.S.} = 2\hat{I}\ \frac{1}{2\sqrt{2}} \qquad (5.4.22)$$

For the discrete quantity $i_{a2(t)}$ "rectification" is achieved by complementing negative half cycle values.

The integration is then performed using Simpsons [61] method, which for any discrete function f(x) is expressed as:

$$\int_{a}^{a+2hn} f(x)\,dx = \frac{h}{3}\left[f(a)+4f(a+h)+2f(a+2h)+4f(a+3h)+---+f(a+2hn)\right] \qquad (5.4.23)$$

where h is the sampling interval.

If $f(x) = \hat{I}_{a_2} \sin \omega t$, and $h = \frac{\pi}{6}$ for 12 s.c., then integration of f(x) over half a cycle becomes:

$$\int_{o}^{\pi} f(x)\ dx = \frac{\pi}{6\times 3}\left[f(o) + 4f(\tfrac{\pi}{6}) + 2f(\tfrac{\pi}{3}) + 4f(\tfrac{\pi}{2}) + 2f(\tfrac{2\pi}{3}) + 4f(\tfrac{5\pi}{6}) + f(\pi)\right] \qquad (5.4.24)$$

The terms $f(\pi, \frac{5\pi}{6}----o)$ are replaced by stored values of $i_{a2(t)}$ at (t, t-1, ---t-6) giving:

$$2\, i_{a_{2(t)}} = \frac{\pi}{18}\left[f(t-6)+ 4f(t-5)+ 2f(t-4)+ 4f(t-3) + 2f(t-2) + 4f(t-1) + f(t)\right] \qquad (5.4.25)$$

From 5.4.22 the corresponding R.M.S. value $i_{2(t)}$ is

$$i_{2(t)} = M.\left[f(t-6) - - -f(t)\right]$$

where $M = \frac{\pi}{18}\cdot\frac{1}{2\sqrt{2}}$

The division by 3 of the negative sequence current evaluation, expression 5.4.18, is included in the constant M, thus:

$$M' = \frac{\pi}{18}\cdot\frac{1}{2\sqrt{2}}\cdot\frac{1}{3} = 0.02057$$

This R.M.S. evaluation is thus implemented by summation of previously stored values of $i_{a2(t)}$. Multiplication of stored terms by 2 or 4 is achieved when required by one or two left shifts respectively. M' is stored as a constant.

5.4.6 Trip characteristic

At intervals of 0.5s, determined by the real time clock, the trip calculation routine samples the R.M.S. value I_2. The magnitude of I_2 is compared with a stored limit $I_{2\ max}$, corresponding to the continuous

negative sequence rating of the generator.

Provided that $I_2 < I_{2\ max}$ a "normal" entry to the breaker control routine is made, the operation of which is described in the next section. If, however, I_2 exceeds $I_{2\ max}$, I_2^2 is computed and used as an input to the trip characteristic calculation.

The approximate rotor heating law of 5.4.19 can be rewritten as:

$$\int_{o}^{T_r} I_2^2 \, dt = K \qquad (5.4.26)$$

where T_r is the time before tripping occurs.

I_2 is a "dc" quantity which varies very slowly and the integration can be replaced by a simple summation, hence:

$$\sum_{t=o}^{t=T_r} I_{2(t)}^2 = K \qquad (5.4.27)$$

In the above expression K is dependent upon the cooling method used for the generator. The values given for K in table 5.1 require that, T_r is expressed in seconds, and I_2 as a per-unit value of the rated full load machine current.

The quantity I_2 provided by the R.M.S. routine is however, an absolute magnitude. I_2 must therefore be reduced to its per-unit value:

$$I_{2(p.u)} = \frac{I_{2(abs)}}{I_p} \qquad (5.4.28)$$

where I_p is a scaling factor representing a 1 p.u. load current. The value of I_p is determined by the machine rating, current transducer/filter/amplifier gains, and the analogue-digital conversion gain, thus:-

$$I_p = I_R . A_T . A_F . A_A . A_C \qquad (5.4.29)$$

where I_R = Rated machine current (A).

A_T = Current transducer gain (V/A)

A_F = Filter gain

A_A = Amplifier gain (or attenuator)

A_C = A-D converter gain = (51.2 for a 10 bit $\pm$10V F.S converter)

Further, since summation occurs at 0.5s intervals during fault conditions, the value of K must be doubled to give T_r in seconds.

Substitution from 5.4.28 into 5.4.27 gives:

$$\sum_{t=o}^{t=T_r} I_2^2(t) = 2K I_p^2 \qquad (5.4.30)$$

The value of $2KI_p^2$ is likely to approach the maximum that can be accomodated in a 16 bit word i.e.

65,536. To avoid the risk of arithmetic overflow during summation, with the attendant software detection requirments 5.4.30 is re-written:

$$\sum_{t=o}^{t=T_r} \tfrac{1}{2} I'^{2}_{2(t)} = K I_p^2 \qquad (5.4.31)$$

While an unbalanced condition exists, i.e. $I_2 > I_{2\ max}$, the trip characteristic routine sums I_2^2 at 0.5s intervals. At each iteration the value of the summation is compared with the stored constant KI_p^2.

If

$$\Sigma \tfrac{1}{2} I^2_{2(t)} \leqslant KI_p^2$$

a "normal" entry to the breaker control routine results. The initiation of action to remove the unbalanced load occurs only when:

$$\Sigma \tfrac{1}{2} I^2_{2(t)} > KI_p^2$$

which causes entry of the breaker control software at the "trip command" point.

5.4.7 Support routines

In addition to the major functions of the algorithm, several ancillary routines are included within the software. They are:

i) Real time clock.

The clock routine is entered on completion of

each data input transfer. At 12 s.c. this occurs every 1.66 ms. A cyclic count of 60 is thus used to determine a 0.1s interval, at which the main clock counter location is incremented.

ii) Circuit breaker control.

This routine generates outputs for the circuit breaker control port. It is entered at the end of each program iteration. Two points of entry are possible, depending upon the outcome of the fault detection computation. The "normal" entry causes the routine to re-iterate the "NO-TRIP", "RELAY HEALTHY" status of the output, before a return is made to the main program.

Entry to the 'trip command' portion of the routine is prevented unless bit 8 in the P.C.W. is set. This facility allows external monitoring or back-up processors to inhibit action by the relay.

If entry is made to the trip section, the "TRIP", "RELAY HEALTHY" data word is output to the breaker control port. A "circuit breaker tripped" code is also placed in the relay communication port.

The routine then cycles in a data input mode, during which no fault calculations are performed. When bit 8 in the P.C.W. is cleared, exit from the routine, followed by a complete re-initialisation of the relay, takes place.

iii) Initialisation.

Following power-up of the relay, or exit from a breaker trip operation, the initialisation routine is executed.

The routine resets the breaker control output to the "no-trip" state, and clears all alarm conditions. Additionally many of the storage locations used by the main program are cleared, and the real time clock is zeroed. Control is then transfered to the main program. These operations greatly improve the stability of the relay under transient conditions, which may arise during start up of either the protection equipment or the primary system plant.

iv) Alarm output.

In common with conventional practice the relay must provide an alarm output when the negative sequence component exceeds a pre-determined percentage of the trip value.

The alarm routine is enabled by setting bit 7 in the P.C.W.

If a negative sequence current in excess of 0.15 CMR (i.e. 100% of the trip value) is sustained for 5 seconds the routine outputs an alarm word to the relay communication port. The word comprises alarm identification data, and the magnitude of the negative sequence component.

An indication of the time available for fault location and rectification is thus provided to the plant operator. This feature represents an improvement on the analogue relay alarm facilities.

Further outputs from the routine are inhibited until acknowledgement of the existing alarm is indicated by clearing and re-setting P.C.W. bit 7.

v) Self checking routine.

The reference voltage V_R is derived from the relay power supplies, via resistor dividers. It is sampled and converted by the acquisition interface and forms part of each data block. V_R is nominally 6.1V.

Setting P.C.W. bit 6 enables the self checking routine. The measured value of V_R is compared with limits preset at $\pm$ 1.0% of the nominal value. If the limits are exceeded, the routine issues an alarm word, consisting of identification data, and the measured value of V_R, to the communication port. The "NO-TRIP" "RELAY FAULTY" condition is output to the breaker control port, to initiate back-up protection. All processing within the relay is terminated following this alarm, until acknowledgement is indicated by clearing and re-setting P.C.W. bit 6. A full initialisation results upon exit.

The scope of self checking routines which may be included within a digital relay is clearly considerable.

This example serves to demonstrate some of the possibilities. Correct operation of the power supplies, a major portion of the data acquisition interface and many protection processor functions are verified by the test. Communication between the relay and other processors increases the value of the facility.

vi) Data output.

To assist in monitoring the performance of the protection equipment, a provision is made to access data at several stages of the algorithm computation.

On demand the data output routine transfers the required data to the communication port. The data source is determined by the status of bits Ø-4 in the P.C.W. as follows:

P.C.W. Bit(set)	Output Function
Ø	Raw phase current samples
1	Filtered currents
2	Negative sequence component x3
3	Rectified value of above
4	R.M.S. value of N.S. current.

The output routine does not effect the main program in any way. However, it provides a powerful method for remote diognostic and monitoring of a relay installation.

A flow chart for the complete algorithm including both the main and ancillary routines is given in Fig.5.8.

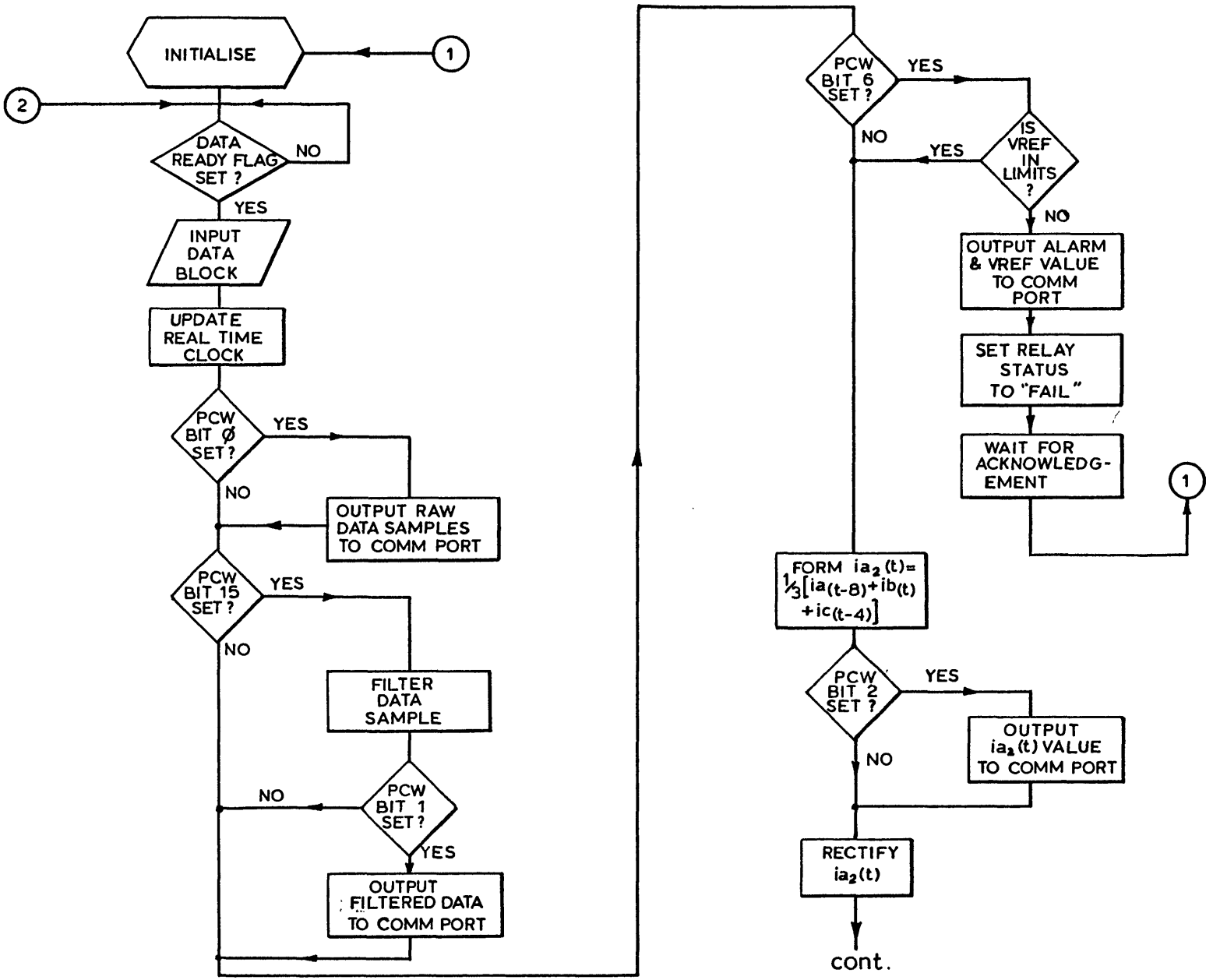
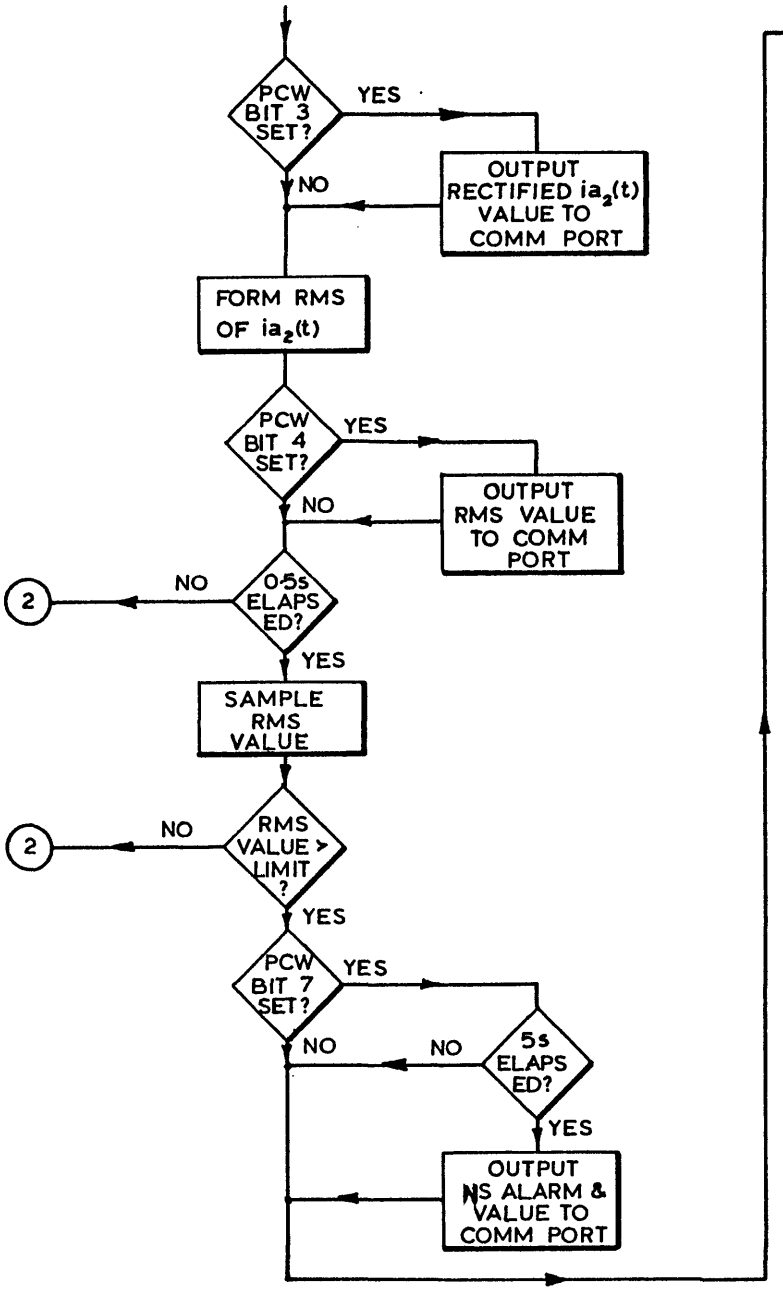


Fig. 5·8 Negative sequence protection, program flow chart.


PCW BIT 3 SET?
YES
NO
OUTPUT RECTIFIED $ia_2(t)$ VALUE TO COMM PORT
FORM RMS OF $ia_2(t)$
PCW BIT 4 SET?
YES
NO
OUTPUT RMS VALUE TO COMM PORT
0·5s ELAPSED?
NO
2
YES
SAMPLE RMS VALUE
RMS VALUE > LIMIT ?
NO
2
YES
PCW BIT 7 SET?
YES
NO
5s ELAPSED?
NO
YES
OUTPUT NS ALARM & VALUE TO COMM PORT

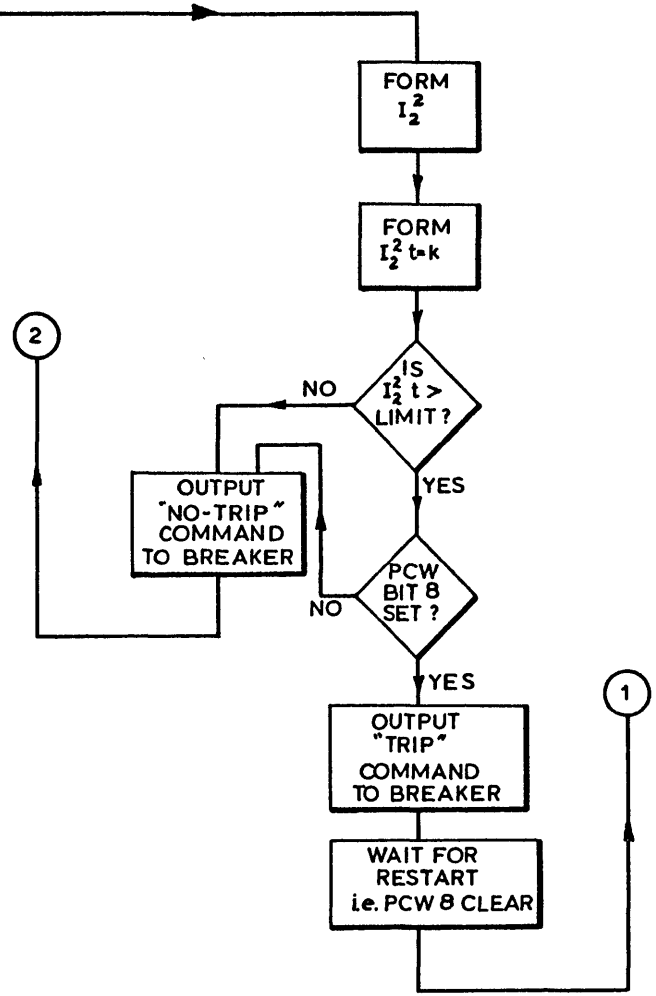

FORM I_2^2
FORM $I_2^2 t=k$
2
IS $I_2^2 t >$ LIMIT?
NO
YES
OUTPUT "NO-TRIP" COMMAND TO BREAKER
PCW BIT 8 SET ?
NO
YES
1
OUTPUT "TRIP" COMMAND TO BREAKER
WAIT FOR RESTART i.e. PCW 8 CLEAR

Fig. 5·8 cont.

5.5 Relay Test Hardware

Fig.5.9 illustrates the test configuration used for the digital unbalanced load relay. The circuit breaker, current transducers and sampling clock reference transformer shown in Fig.5.9 formed part of a laboratory sub-station model[62].

An IMP-16c, micro-processor and NOVA 1210 mini-computer control the operation of the model. Circuit breaker control signals from the digital relay were connected to the model via these two processors. The relay status data was also transfered to the NOVA 1210, which at a later date is to be programmed to provide back-up protection. This arrangement thus forms the first part of a combined integrated-dedicated protection scheme described in Chapter 2.

For test purposes a simulated generator output was derived from the 415V 3 phase main supply and a generator CMR of 1 p.u. = 3.33A was chosen. A balanced load of 25A (7.5 p.u.) could be safely sustained for short periods.

To evaluate and monitor the relay equipment and algorithm performance, the communication port of the relay was connected to a NOVA 3 mini-computer. This computer is supported by a dual flexible disk system and a storage type VDU terminal. Software packages including graphics, calibration and alarm analysis routines, were developed on this machine to assist in the testing of the relay. The

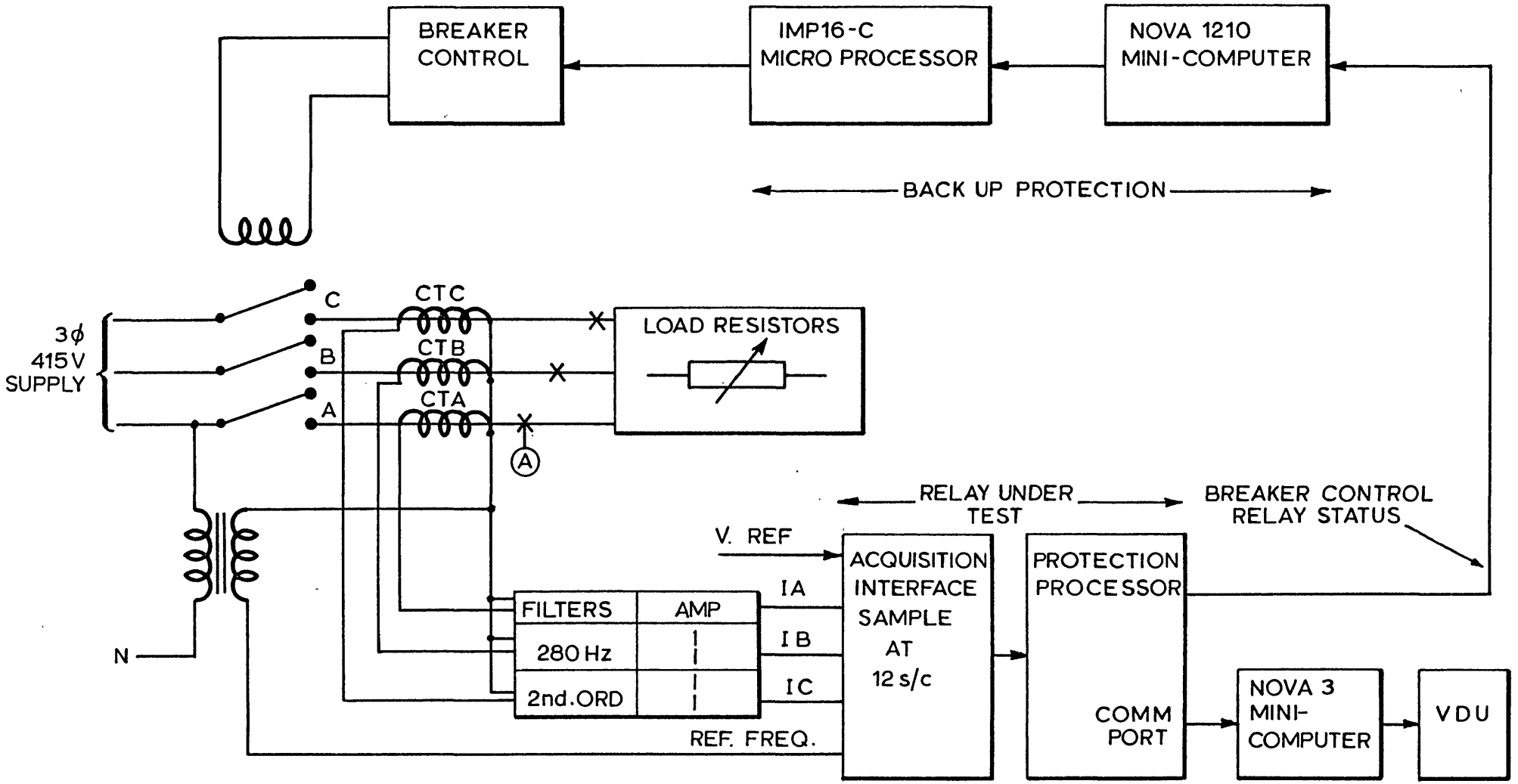


Fig. 5·9 Relay test hardware configuration.

test results described in section 5.6 were obtained using this facility.

5.5.1 Relay inputs.

Data required by the relay consist of: i) 3 phase currents derived from the current transducers. These transducers are linear couplers which provide an output of 6 mV/A. A combined amplifier and 2nd order lowpass Butterworth filter in each signal path limits the bandwidth of the current signals to 280Hz thus preventing aliasing during sampling, (sampling frequency = 600Hz). The amplifier gain is 47. Thus a 1 p.u. CMR (ie. 3.33A) primary system current corresponds to a measured signal of 0.94v.

ii) Program control word, generated manually from 16 switches. To improve the overall system reliability these inputs are entered via a digital data port of the acquisition interface as described in Chapter 3.

iii) Self test reference voltage. This input has a nominal value of 6.1V obtained by a resistor divider network from the internal $\pm$ 15V and dual +5V relay power supplies.

iv) Sampling clock reference. The sampling clock module of the acquisition interface is provided with a 50 Hz 6.3V reference signal from a voltage transformer in the sub-station model.

5.6 Test Results.

For test purposes the relay characteristic was programmed to provide negative sequence protection suitable for a turbo alternator with conventional hydrogen cooling at 2 bar.

From table 5.1 a K value of 12 is applicable for this type of machine. The current transducer/filter/amplifier conversion gain gives a scaling factor (I_p^2) of 2316. A trip limit constant K I_p^2, of 27800 was therefore used. The continuous negative sequence rating of the machine, from table 5.1, is 0.15 CMR., representing the minimum trip level of the relay. It is also used as the limit for the negative sequence alarm routine.

To evaluate the detection and tripping performance of the relay, two phases of the load were isolated. The negative sequence component was thus $\frac{1}{3}$ of the measured current in the remaining phase. Since a 1 p.u. balanced load was taken to be 3.33A, the single phase measured current directly corresponds to the per-unit negative sequence value i.e. 5A measured = 0.5 p.u. etc.

The clearance times for negative sequence currents from 0.1 CMR to 2.5 CMR are shown in Fig. 5.10. For comparison the machine withstand characteristic is drawn on the same figure.

To test the stability of the relay during fault

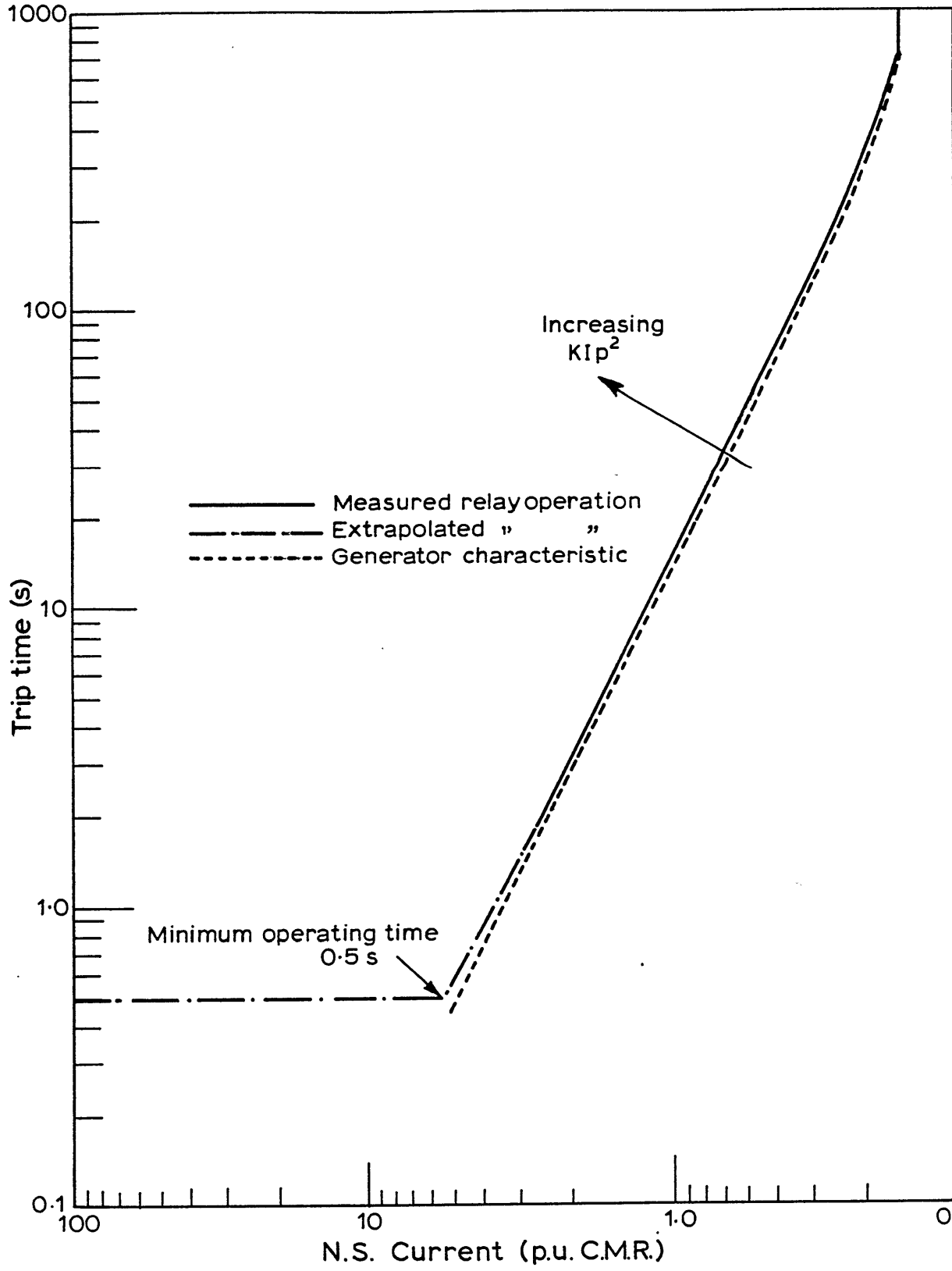


Fig.5·10 Relay operating characteristic.

conditions for which operation must not occur two cases were investigated.

i) Positive sequence current sensitivity.

In this test the balanced 3 phase load was increased in 2.5 p.u. CMR increments up to a maximum of 7.5 CMR. The R.M.S. values of the negative sequence current computed by the relay algorithm for these load conditions are plotted in Fig. 5.11.

ii) Zero sequence current sensitivity.

To produce a zero sequence current condition, the 3 phases of the load were connected together. A single phase-neutral source was then applied to the load. Current increments of 2.5 p.u. with a maximum of 7.5 p.u. were used as in the positive sequence case. Fig. 5.12 plots the computed negative sequence R.M.S. values for each load condition.

Finally Fig. 5.13 illustrates several of the alarm output analysis messages.

5.7 Conclusions

From Fig. 5.10 it can be seen that the digital relay provides a close match to the tripping characteristic required for negative sequence protection. By alteration of the stored constant KI_p^2 this characteristic can be easily modified to suit other generator types.

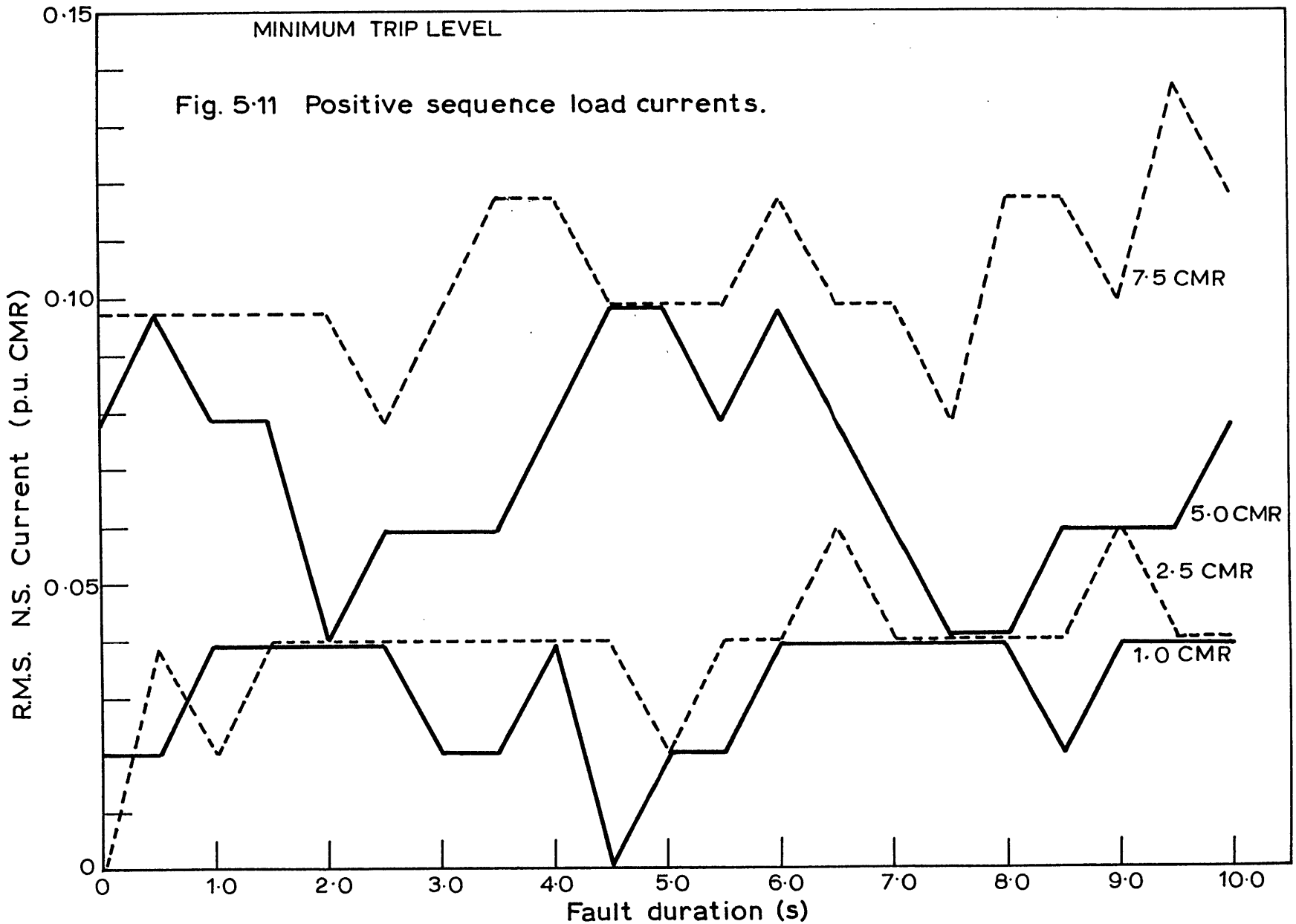


Fig. 5·11 Positive sequence load currents.

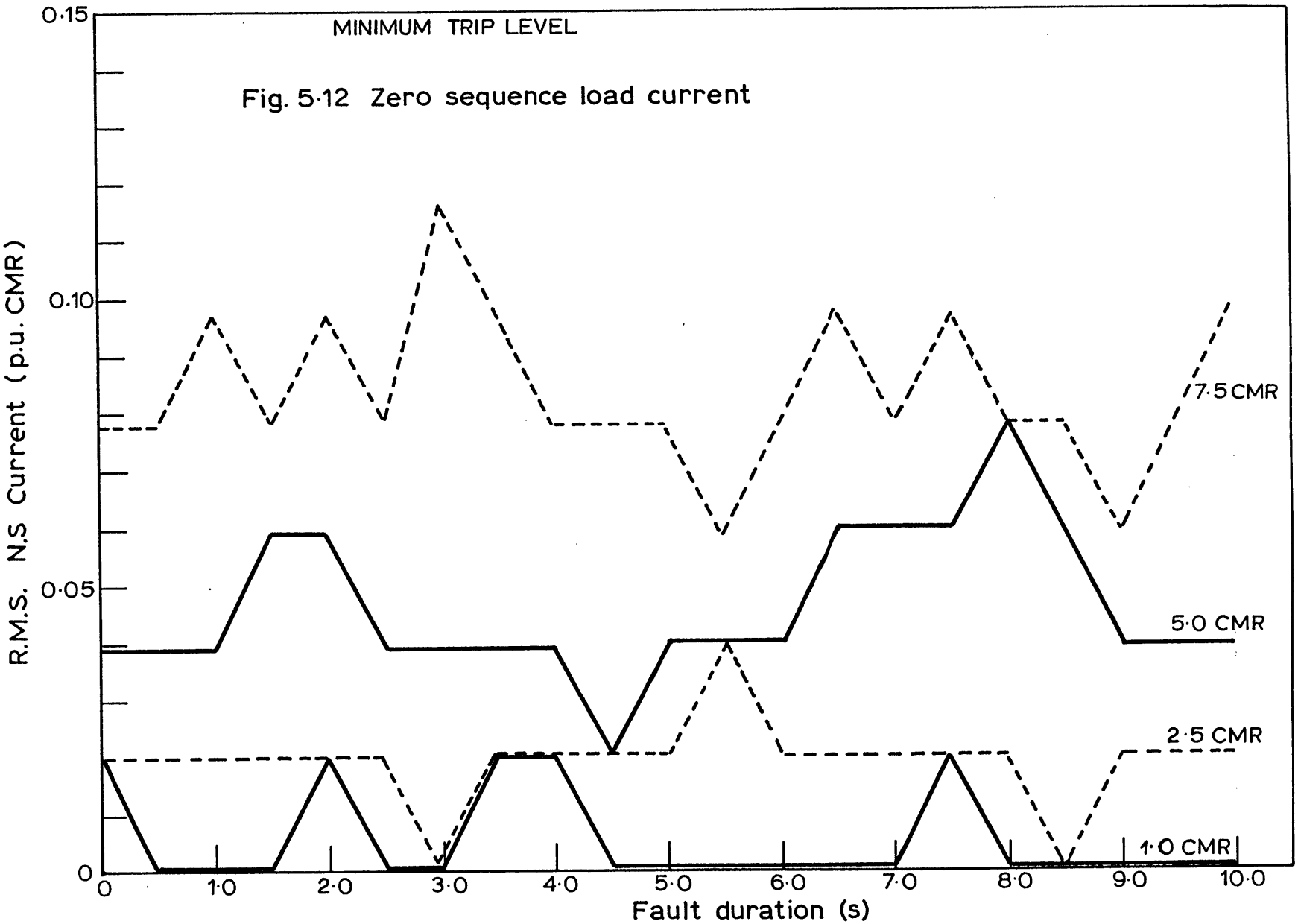


Fig. 5·12 Zero sequence load current

```
  ***RELAY MONITOR 5/10/77***
NEGATIVE SEQUENCE LIMIT ALARM
VALUE=0:94CMR
DATE:23/11/77 TIME:16/04/14
TYPE 1 TO ACKNOWLEGE 1


 ***RELAY MONITOR 5/10/77***
BREAKER TRIPPED BY PROTECTION
DATE:23/11/77 TIME16/04/20
TYPE 1 TO ACKNOWLEGE 1


 ***RELAY MONITOR 5/10/77***
MEASUREMENT ERROR
VALUE=-0.02V  CORRECT VALUE=6.12V
DATE:23/11/77 TIME:16/07/26
TYPE 1 TO ACKNOWLEGE 1
```

Fig. 5·13 Typical relay alarm analysis

During positive and zero sequence overcurrent conditions the relay remains stable. The computed negative sequence values for these conditions, shown in Figs. 5.11, 5.12, do not exceed the "minimum trip level" for primary currents up to 7.5 p.u. It must be noted that the "minimum trip level" corresponds to the continuous negative sequence rating of the machine and implies a tripping time of 800s-1000s for the characteristic used. The computed values shown in Figs. 5.11, 5.12 can be attributed to imbalance in the CT's, filters, amplifiers and loads.

The method used to calculate the negative sequence component of the primary system currents relies heavily on the sampling rate of 12 s.c. It is therefore essential that the sampling clock is synchronised to the power frequency and its reference frequency source must be carefully chosen. Provided this precaution is taken, the operational accuracy of the relay will be maintained for all primary system frequencies within the lock range of the clock (Chapter 3). In this respect, the digital relay performance is superior to that of existing analogue equipments in which static sequence filters are used. These filters are clearly frequency sensitive. Variations in system frequency will impair their operation and hence the overall characteristics of the protection.

Implementation of the protection software,

including the output routines requires 550 words of program storage, 67 data locations and 11 constants. The basic protection routines are executed in approximately 220μs. At 12 s.c. this represents 14% of the time available between samples. These requirements lie well within the design limits of the equipment.

CHAPTER 6

RELAY APPLICATIONS (ii) OVERCURRENT AND EARTH FAULT PROTECTION

6.1 Introduction

As additional demonstration of the flexibility of the digital relay a second application was programmed. Overcurrent (O/C) and earth fault (E/F) protection were chosen for this purpose.

O/C relays provide a relatively simple and inexpensive form of protection and are consequently widely used. To satisfy the diverse requirements of their application, electromechanical and static analogue O/C equipments are available offering a range of performance characteristics. Commonly used O/C relay characteristics include inverse definite minimum time (IDMT), very inverse or extremely inverse time/current functions, whilst other types feature high set instantaneous, definite time or voltage controlled modes of operation[2].

In many applications O/C relays are supplemented by separate earth fault protection. The earth fault relay provides a more sensitive response to phase-ground faults than can be obtained by O/C protection alone.

From an economic view-point it is unlikely that

the replacement of individual O/C or E/F relays by digital equipments, could be justified. However, if these functions are incorporated within a single relay offering a range of pre-programmed characteristics and logical switching control, e.g. auto-reclose, then a digital approach becomes more attractive.

The combination of O/C and E/F protection within a single relay is effectively an integration of protection processes. It is important that the shortcomings of this technique, discussed in Chapter 2, are avoided. Thus the relay must be able to completely execute the protection routines between each data sampling instant without recourse to "starting" methods or an "operating system". To satisfy this condition, simple, robust fault detection and trip characteristic algorithms were developed.

6.2 Program Organisation

In Chapter 5 the negative sequence protection software was written as a single, continuous program to perform a specific task. To provide the flexibility of application which is a major feature of the digital O/C, E/F relay an alternative software structure was employed.

Each of the relay tasks required for the various protection functions was pre-programmed as a sub-routine. The main application program encompassing the operation of the complete relay was then simply written as a series of sub-routine "calls".

For the O/C, E/F application the following sub-routines were developed: i) Relay initialisation, ii) Data input, iii) Real time clock. iv) Data filtering v) Earth fault detection and tripping. vi) Overcurrent fault detection. vii) "High set" O/C tripping. viii) 3s Inverse definite minimum time (IDMT) O/C characteristic. ix) Extremely inverse (EI) characteristic x) Circuit breaker control. xi) Self-test. xiii) Simple auto-reclose.

These routines were pre-programmed in the macro-memory EPROMS and occupied 863 locations. The main relay program of 57 locations comprised a sequence of "calls" to the sub-routines.

Several of the main program calls were made conditional upon the status of a control word supplied by external switches. It was thus possible to test various relay configurations and characteristics by setting the appropriate switches. Fig. 6.1 is a flow chart of the main program.

6.3 The Sub-Routines

The O/C and E/F relay sub-routines are described in the order in which they are used by the main program. Flow-charts for each routine, identified by a mnemonic name, are given in Fig. 6.2. The first five relay functions were largely derived from the negative sequence application

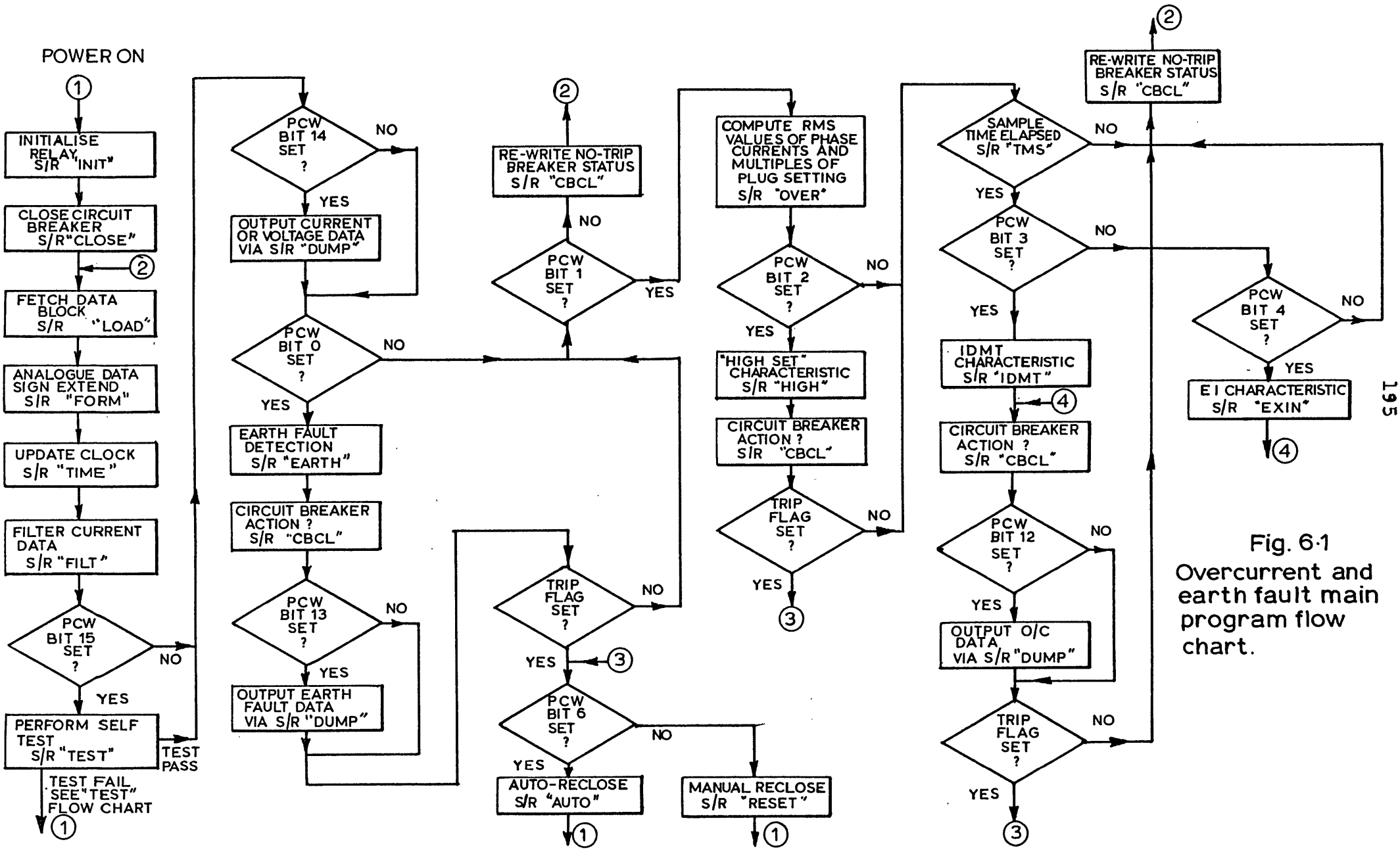


Fig. 6·1
Overcurrent and earth fault main program flow chart.

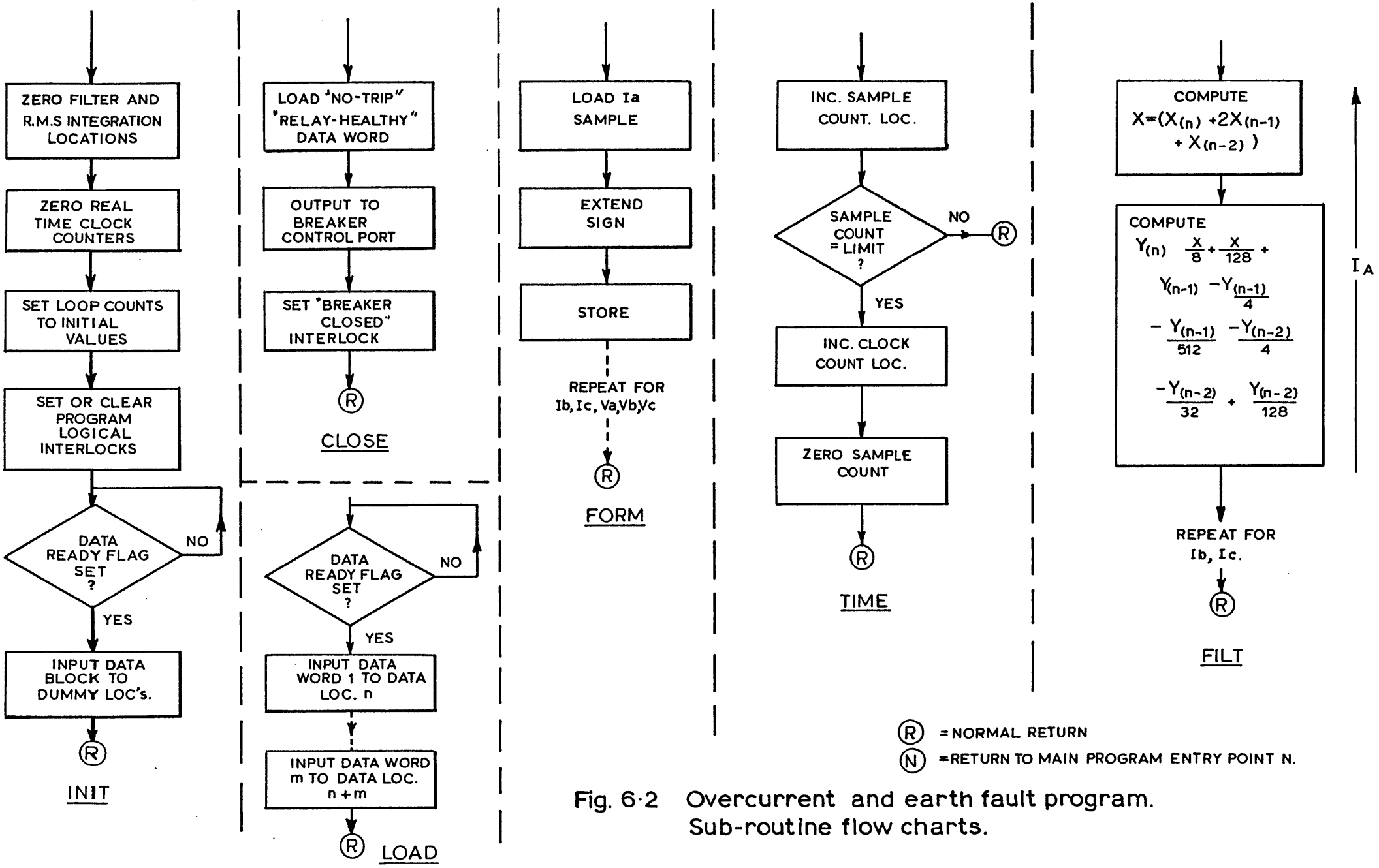


Fig. 6·2 Overcurrent and earth fault program. Sub-routine flow charts.

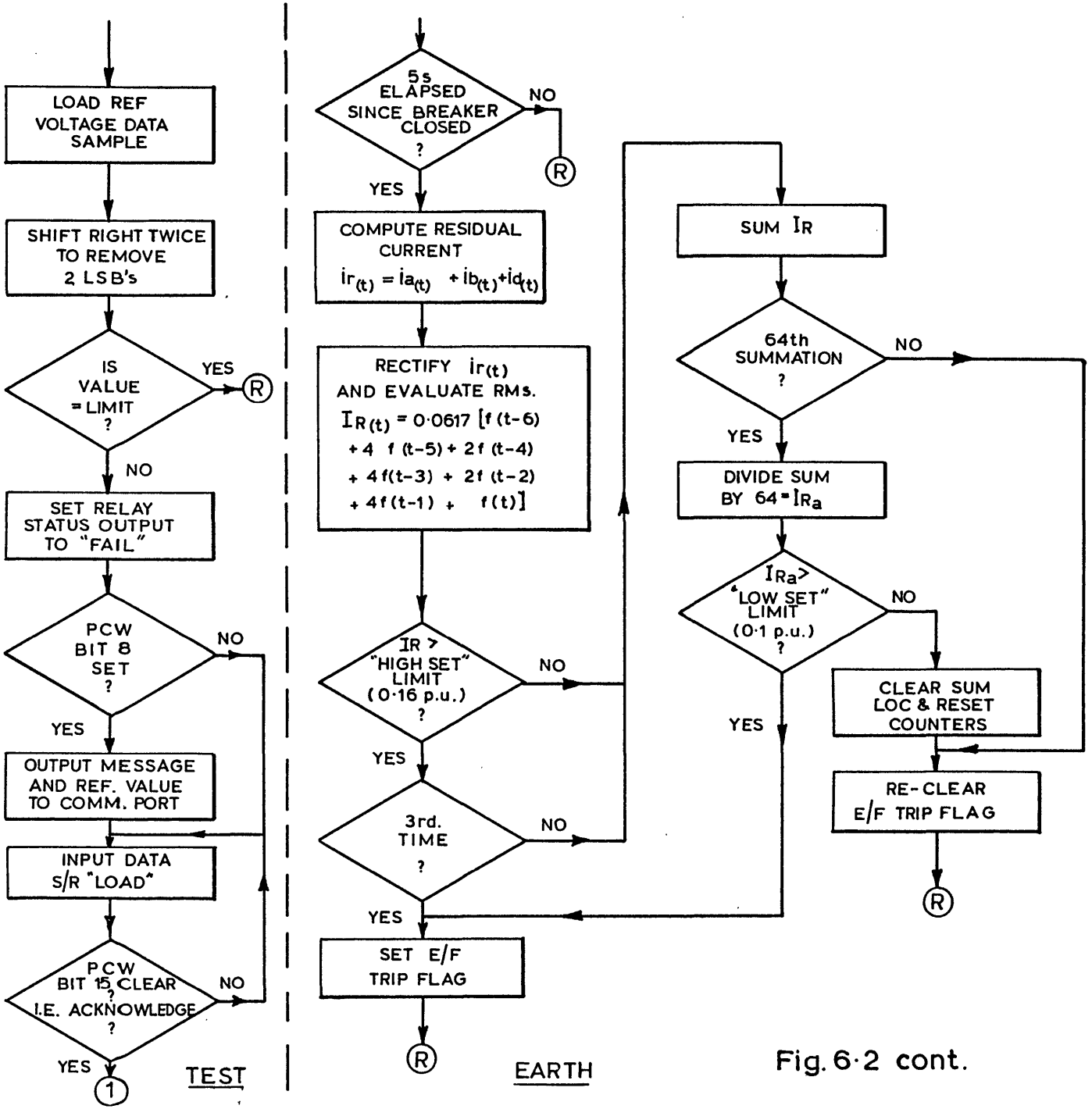

LOAD REF VOLTAGE DATA SAMPLE
SHIFT RIGHT TWICE TO REMOVE 2 LSB's
IS VALUE = LIMIT ?
YES
R
NO
SET RELAY STATUS OUTPUT TO "FAIL"
PCW BIT 8 SET ?
NO
YES
OUTPUT MESSAGE AND REF. VALUE TO COMM. PORT
INPUT DATA S/R "LOAD"
PCW BIT 15 CLEAR I.E. ACKNOWLEDGE ?
NO
YES
1
TEST
5s ELAPSED SINCE BREAKER CLOSED ?
NO
R
YES
COMPUTE RESIDUAL CURRENT
$ir_{(t)} = ia_{(t)} + ib_{(t)} + id_{(t)}$
RECTIFY $ir_{(t)}$ AND EVALUATE RMS.
$I_{R(t)} = 0.0617\,[f(t-6) + 4f(t-5) + 2f(t-4) + 4f(t-3) + 2f(t-2) + 4f(t-1) + f(t)]$
IR > "HIGH SET" LIMIT (0·16 p.u.) ?
NO
YES
3rd. TIME ?
NO
YES
SET E/F TRIP FLAG
R
EARTH
SUM IR
64th SUMMATION ?
NO
YES
DIVIDE SUM BY 64 = I_{Ra}
I_{Ra} > "LOW SET" LIMIT (0·1 p.u.) ?
NO
YES
CLEAR SUM LOC & RESET COUNTERS
RE-CLEAR E/F TRIP FLAG
R
PCW BIT 5 SET ?
NO
YES (TRIP ENABLED)
E/F TRIP FLAG SET ?
YES
NO
REPEAT TEST FOR Ia, b, c O/C TRIP FLAGS
ANY FLAG SET
NO FLAG SET
RE-LOAD BREAKER PORT WITH "NO TRIP" WORD
CLEAR "BREAKER TRIPPED" FLAG
R
LOAD "TRIP" "RELAY-HEALTHY" DATA WORD
OUTPUT TO BREAKER PORT
PCW BIT 8 SET ?
NO
YES
OUTPUT MESSAGE AND IDENT TO COMM. PORT
SET "BREAKER TRIPPED" FLAG
R
CBCL

Fig. 6·2 cont.

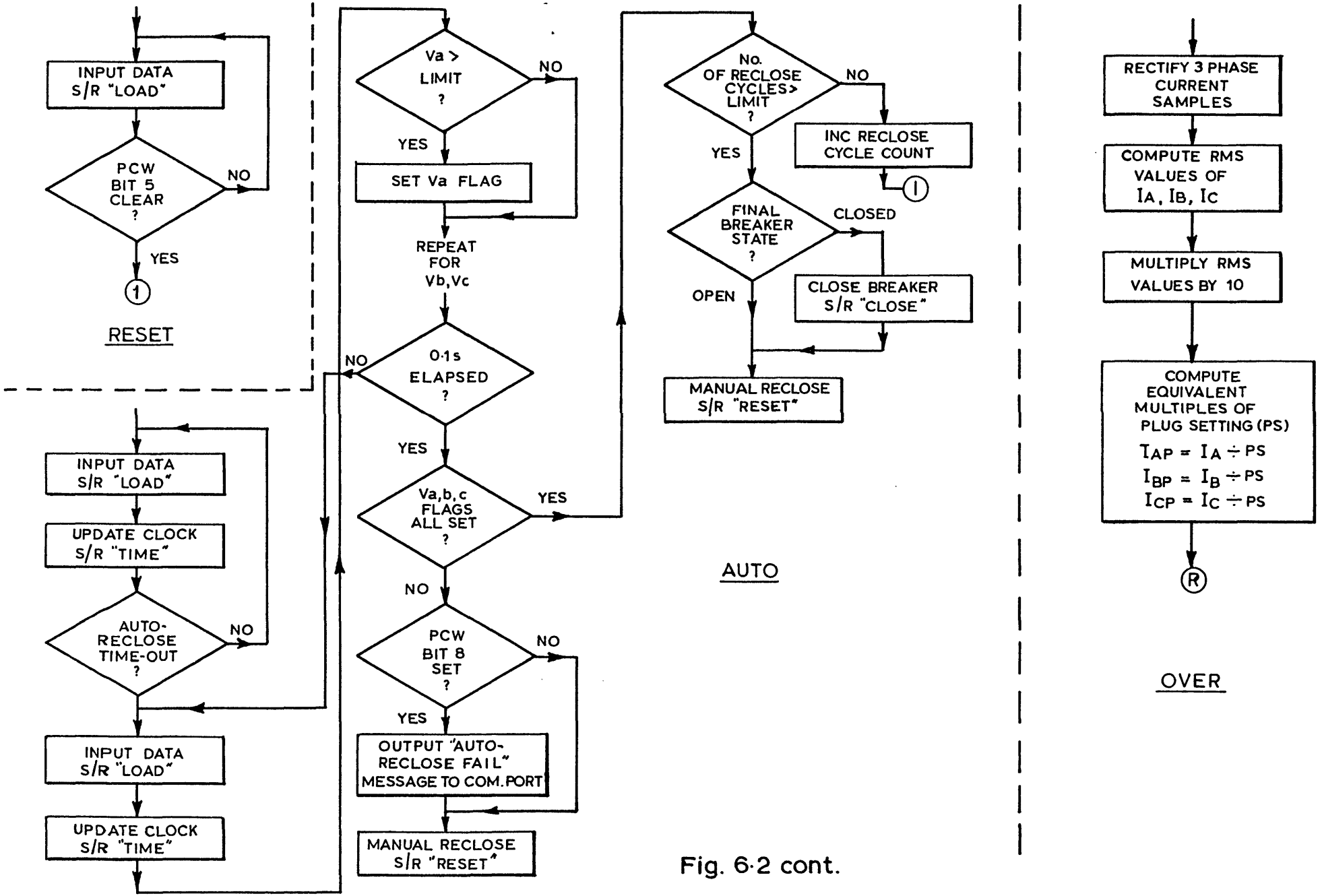
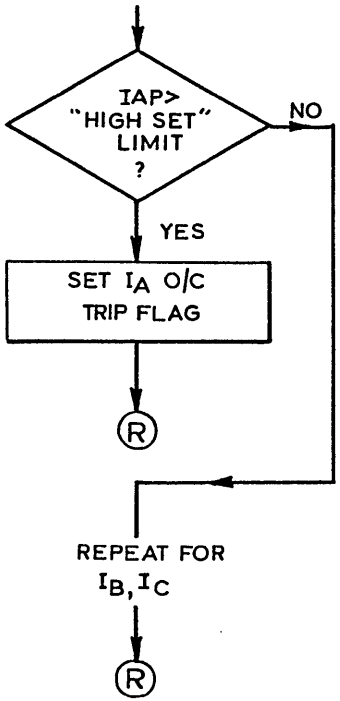
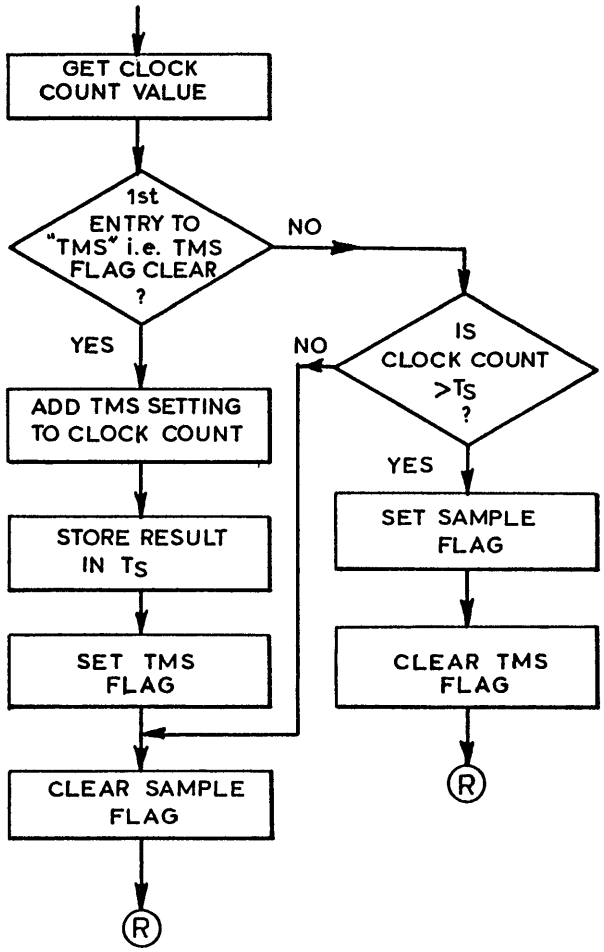


Fig. 6·2 cont.

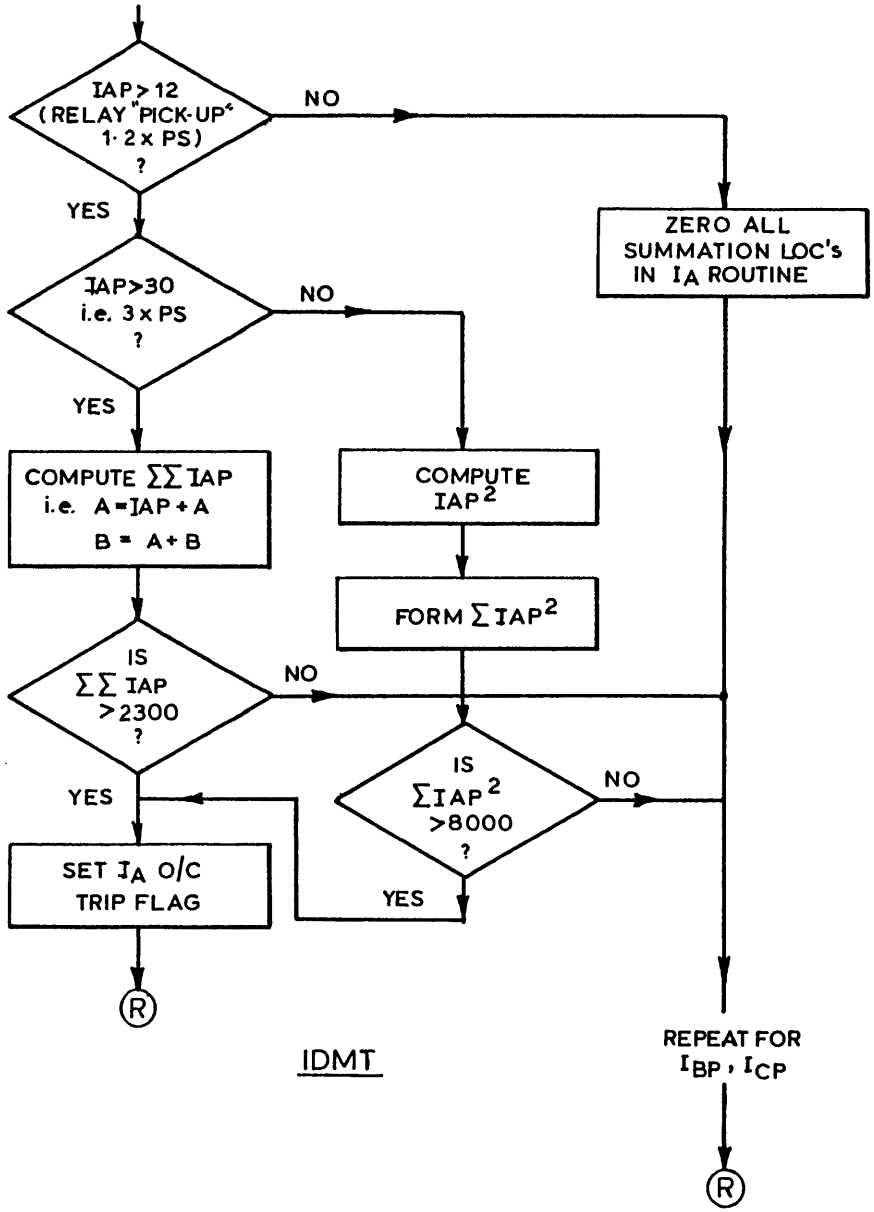
OVER



HIGH



TMS



IDMT

Fig. 6·2 cont.

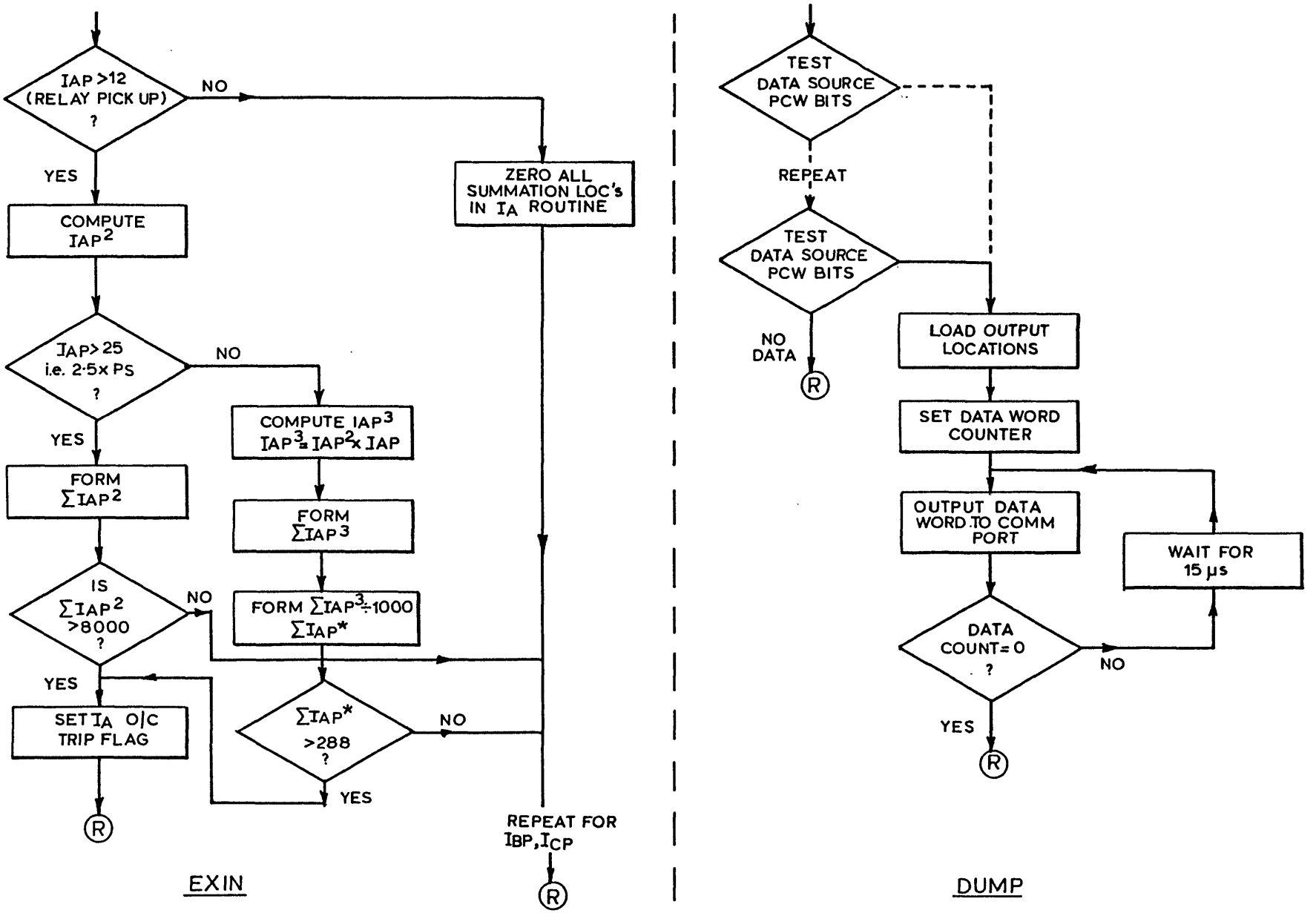

IAP >12 (RELAY PICK UP) ?
NO
YES
ZERO ALL SUMMATION LOC's IN IA ROUTINE
COMPUTE IAP2
IAP> 25 i.e. 2·5x PS ?
NO
YES
COMPUTE IAP3 IAP3=IAP2x IAP
FORM ΣIAP2
FORM ΣIAP3
IS ΣIAP2 >8000 ?
NO
FORM ΣIAP3÷1000 ΣIAP*
YES
ΣIAP* >288 ?
NO
YES
SET IA O/C TRIP FLAG
R
REPEAT FOR IBP, ICP
R
EXIN
TEST DATA SOURCE PCW BITS
REPEAT
TEST DATA SOURCE PCW BITS
NO DATA
R
LOAD OUTPUT LOCATIONS
SET DATA WORD COUNTER
OUTPUT DATA WORD TO COMM PORT
WAIT FOR 15 µs
DATA COUNT= 0 ?
NO
YES
R
DUMP

Fig. 6·2 cont.

software. Since their operation is fully described in Chapter 5 they are only briefly mentioned here.

6.3.1 Relay initialisation

Automatic initialisation of the equipment is performed following switch-on or upon exit from certain of the other routines. Two sub-routines, INIT and CLOSE implement this function.

INIT clears specific data memory locations used by other parts of the software, resets the real time clock and pre-sets logical program flags. The routine then synchronises the protection processor and acquisition interface operation by discarding the first data sample, (Chapter 5). CLOSE initialises the circuit breaker control port to the "NO-TRIP", "RELAY-HEALTHY" state.

6.3.2 Data input

The transfer of input data from the acquisition interface buffer store to the processor data memory, controlled by the interface data ready flag, is accomplished by sub-routine LOAD. A further routine, FORM, is then employed to extend the sign of 10 bit converted analogue data to give a 16 bit two's complement format.

The input data block for the O/C, E/F application comprises: i) Program control word (PCW), ii) 3 phase currents, iii) 3 phase-neutral voltages, iv) self-test reference voltage.

6.3.3 Real time clock

Sub-routine TIME is entered following each data input transfer. The routine maintains a timing location in data memory, which is updated as a function of the relay sampling period.

For the O/C, E/F relay a sampling rate of 12 s.c. was used and the clock was updated at 0.005 s intervals i.e. every 3 samples.

6.3.4 Self test

The relay self-test function, described in Chapter 5, is embodied in sub-routine TEST.

6.3.5 Input filtering

A sub-routine, FILT, comprises a 90Hz, 2nd order, lowpass, Butterworth filter algorithm.

In the O/C, E/F program only the relay current inputs were processed by FILT. The voltage inputs which are required by the auto-reclose routine were not digitally filtered.

6.3.6 Earth fault detection and tripping

Upon entry the earth fault sub-routine, EARTH, computes the residual component of the measured phase currents by summation, i.e.

$$i_{r(t)} = i_{a(t)} + i_{b(t)} + i_{c(t)} \qquad 6.3.1$$

where, i_{a}, b, $c_{(t)}$ are the instantaneous phase current values at sampling time t.

$i_{r_{(t)}}$ is the corresponding residual.

The residual current is sinusoidal and its RMS value, IR, is determined by "rectification" and integration using the techniques described in Chapter 5. IR then forms the input to the trip decision portion of the routine.

In concept the provision of an instantaneous tripping characteristic for earth faults is easily achieved by comparing IR with a preset limit stored in the constant memory.

However, tests performed (Section 6.5.1) showed that the 3 phase system suffered from imbalance when supplying a nominally balanced 1 p.u. load. Moreover the measured imbalance was found to vary during short periods i.e. approximately 1 cycle, between 0.02 p.u. and 0.08 p.u. A mean value, obtained over several cycles, of 0.06 p.u. was therefore used as the continuous imbalance measurement.

To improve the stability and performance of the relay a tripping logic was developed which comprised two parts:

i) The average value of IR is repetitively computed for a period of 64 samples i.e. approximately 5 cycles at 12 s.c. This value is then compared with a

"low set" limit corresponding to 0.1 p.u. Thus with a continuous mean imbalance of 0.06 p.u. the routine initiates tripping for earth fault levels in excess of 0.04 p.u.

ii) To reduce tripping time for severe earth faults IR is also compared at each sampling instant with a "high set" limit of 0.16 p.u. Tripping occurs if IR exceeds this limit for 3 consecutive samples. The "high set" limit corresponds to earth fault levels of approximately 0.1 p.u.

The tripping routines set or clear a logical "trip flag" word as appropriate, before a return is made to the main program. Subsequently this flag is tested by the breaker control sub-routine to determine the required action.

Finally, a facility is provided at the entry point of EARTH to inhibit its execution for a pre-defined period following closure of the circuit breaker, (indicated by a call to sub-routine CLOSE). Erroneous operation of the earth fault protection caused by transient spill currents in the current transformers, or by unequal phase closure of the circuit breaker can thus be eliminated. The required inhibit time is stored in the constant memory. A 5s inhibit time was used during testing.

6.3.7 Circuit breaker control

CBCL, the circuit breaker control sub-routine is called by the main program following the execution of either the E/F or O/C detection routines.

The logical trip flags, associated with the fault detection routines and stored in the data memory, are first examined by CBCL to determine its subsequent operation. Two modes of operation are possible:

i) Trip not enabled by appropriate program control word bit, or, all flags "clear". CBCL re-iterates the "NO-TRIP", "RELAY-HEALTHY" command to the breaker control port and returns control to the main program.

ii) Any flag "set" CBCL initiates circuit breaker operation by storing the "TRIP" "RELAY-HEALTHY" command in the control port. If the "report enable" bit of the PCW is set, CBCL then outputs an encoded message to the relay communication port. This message indicates that the breaker has been tripped and provides data, from the trip flags, which identify the cause of the operation. A "breaker tripped" flag is then set and a return made to the main program.

6.3.8 Reset and auto-reclose

When CBCL returns control to the main program the status of the "breaker tripped" flag is tested. If "set" the main program interrogates the PCW to determine

the reset mode required. The two reset modes are:

i) Manual reset and reclosure.

The sub-routine RESET causes the relay to cycle in a data input mode. After each data transfer the "trip enable" bit of the PCW is examined. When RESET detects that "trip enable" has been cleared (by manual intervention), an exit is made via the INIT and CLOSE sub-routines. The relay is thus initialised and the circuit breaker reclosed.

ii) Auto-reclose.

An auto-reclose facility is provided by sub-routine AUTO. This routine cycles in a data input mode and maintains the real time clock by repetitively calling LOAD, FORM and TIME. The cycle continues until the "time out" portion of AUTO is completed i.e. the auto-reclose time has elapsed. A further section of the sub-routine then inputs data and compares the relay voltage samples with a pre-set limit (stored in the constant memory) for a period of 0.1s. If, after this time, any of the 3 phase voltage inputs have failed to reach the set limit AUTO places an encoded "auto-reclose failure" message in the relay communication port and invokes the manual reset procedure via RESET.

Provided that the phase voltage test is passed AUTO initialises the relay program and recloses the circuit

breaker (INIT and CLOSE). A reclose cycle count is maintained in the routine and further operations are inhibited when this count reaches a pre-defined limit. In this case AUTO determines and sets the required final state of the breaker before transfering control to RESET.

Four parameters stored in the constant memory thus define the operation of AUTO, they are:
i) Auto-reclose time. ii) Phase voltage limit
iii) Number of reclose cycles before lockout. iv) A logical indicator for the circuit breaker status if lockout occurs.

During the tests performed on the relay these parameters were set as: i) 2s ii) 0.8 p.u. iii) 1 iv) 1 i.e. circuit breaker closed on lockout.

6.3.9 Overcurrent fault detection and tripping characteristics.

To provide input data for the O/C detection and tripping routines the filtered phase current values are processed by sub-routine OVER. The first portion of this routine computes the RMS value of each current by rectification and integration (Chapter 5). These values, I_A, I_B, I_C, are then multiplied by 10 to improve the resolution of the following calculations.

$I_{A,\ B,\ C}$ are next reduced to multiples of the relay "plug setting". The plug setting (PS) value is an integer stored in the constant memory and is defined by:

$$P_S = I_S \cdot A_{CT} \cdot A_F \cdot A_{CV} \qquad 6.3.2$$

where: I_S = The primary set point current (A)

A_{CT}= Current transformer/burden transfer ratio (V/A).

A_F = Gain of analogue filters/amplifiers

A_{CV}= A-D converter gain = 51.2 for a 10 bit unit.

Thus for each current the multiple of the plug setting I_{AP}, I_{BP}, I_{CP} is given by:

$$I_{AP} = \frac{I_A}{PS}, \quad I_{BP} = \frac{I_B}{PS}, \quad I_{CP} = \frac{I_C}{PS} \qquad 6.3.3$$

The division routine consists of multiple shifts and subtractions. Approximately 300μs are required to process all three currents.

Since $I_{A,B,C}$ were previously multiplied by 10, $I_{AP,\ BP,\ CP}$ are also an order of magnitude greater than their true values. E.g. when I_A = 1.5 PS, I_{AP} = 15. It is thus possible to represent $I_{AP,\ BP,\ CP}$ as integer numbers with a resolution corresponding to 10% increments of the plug setting.

The values $I_{AP,\ BP,\ CP,}$ form the inputs to one or more of the three possible O/C tripping characteristics, selected by PCW data, for specific relay applications. The characteristics are:

i) High set (instantaneous).

Sub-routine HIGH provides a high set tripping characteristic which may be used independently, or in conjunction with, either the IDMT or extremely inverse characteristics.

HIGH compares $I_{A,\ (BC)P}$ sequentially with a pre-set limit held in the constant memory. If any of the current values exceed this limit, the appropriate O/C phase trip flag is set. A return to the main program follows immediately. When selected, HIGH is executed at each sampling instant. The high set limit was chosen to be 8 times the plug setting for test purposes, (i.e. $I_{A,(BC)P} > 80$).

ii) 3s standard inverse definite minimum time (IDMT).

iii) Extremely inverse (EI)

These characteristics have many common features and are therefore described together. In both cases the characteristic is obtained by the summation of functions of $I_{A,(BC)P}$. Tripping is initiated if the summation result exceeds stored limits.

The first common function which determines the relay tripping performance is provided by a sub-routine, TMS.

Clearly, for a given magnitude of $I_{A,(BC)P}$ and stored limits in the characteristic routines, the relay trip-time will be determined by the rate at which summation occurs. Sub-routine TMS allows the execution intervals of the characteristic routines (and hence the summation rate) to be varied in steps of 0.005s within the range 0.05s to 0.5s. Thus for any given magnitude of $I_{A,(BC)P}$ the relay provides a 10:1 range of operating times. The constant memory parameter which determines the time interval generated by TMS is equivalent to the analogue relay "Time Multiplier Setting".

To produce the required IDMT and EI characteristic algorithms an off-line simulation study was undertaken. In this study several methods of processing the current multiples of the plug setting i.e. $I_{A(BC)P}$ were investigated. The resulting time-current characteristics were compared with the 3s IDMT standard and a typical EI curve[2]. It was found that, for both characteristic types, two separate methods of computation were required, depending upon the fault current level, to produce the correct relay performance. The characteristic relationships are:

IDMT.

i) $I_{AP}^{2}\, t = K_1$ $I_{FAULT} <$ 3 times the plug setting.

ii) $I_{AP}t^2 + t = K_2$ I_{FAULT} > 3 times the plug setting.

E I.

i) $I_{AP}^3 t = K_3$ I_{FAULT} < 2.5 times the plug setting.

ii) $I_{AP}^2 t = K_4$ I_{FAULT} > 2.5 times the plug setting.

where: I_{AP} = current multiple of the plug setting (similarly for I_{BP}, I_{CP}).

t = trip time.

$K_{1,2,3,4}$ = Constants determined by the internal processor representation of I_{AP}.

The generation of a tripping law of the form $I^2t = K$ is described in Chapter 5. Since I_{AP} is a "dc" quantity integration is replaced by summation and typically tripping is initiated when:

$$\sum_{o}^{t} I_{AP}^{(n)} > K_n$$

The quadratic form of IDMT (ii) is derived by:

$$\sum_{o}^{t} \sum_{o}^{t} I_{AP} > K_m$$

Constants $K_{1,2,3,4}$ are determined by considering arbitrary time-current values on the required characteristics.

E.g. For K_1 IDMT relay operation time for I = 2.0 times the plug setting and TMS = 1.0 is found from the standard characteristic to be 10s. In this example I_{AP} = 20, and for TMS = 1.0 the TMS sub-routine causes the characteristic routines to be executed at 0.5s intervals.

Thus for tripping to occur in 10s :

$$K_1 = I_{AP}^2 \cdot 20 = 400.20$$

$$\therefore K_1 = 8000$$

Similarly K_2 = 2300: K_3 = 288,000: K_4 = 8000.

Figs. 6.3 and 6.4 illustrate the results of the off-line study and compare the trip laws derived with the standard IDMT and E I characteristics respectively.

Upon entry the selected characteristic sub-routine IDMT (IDMT) or EXIN (EI) first compares $I_{A,(BC)P}$ with a pre-set "relay pick up" value. This parameter, stored in the constant memory, is nominally set at 12, i.e. I_{FAULT} > 1.2 times the plug setting. Current values below this setting cause the routine to reset its summation locations and return a no-trip decision to the main program.

If the pick up value is exceeded IDMT or EXIN determine the magnitude of the fault current and evaluate

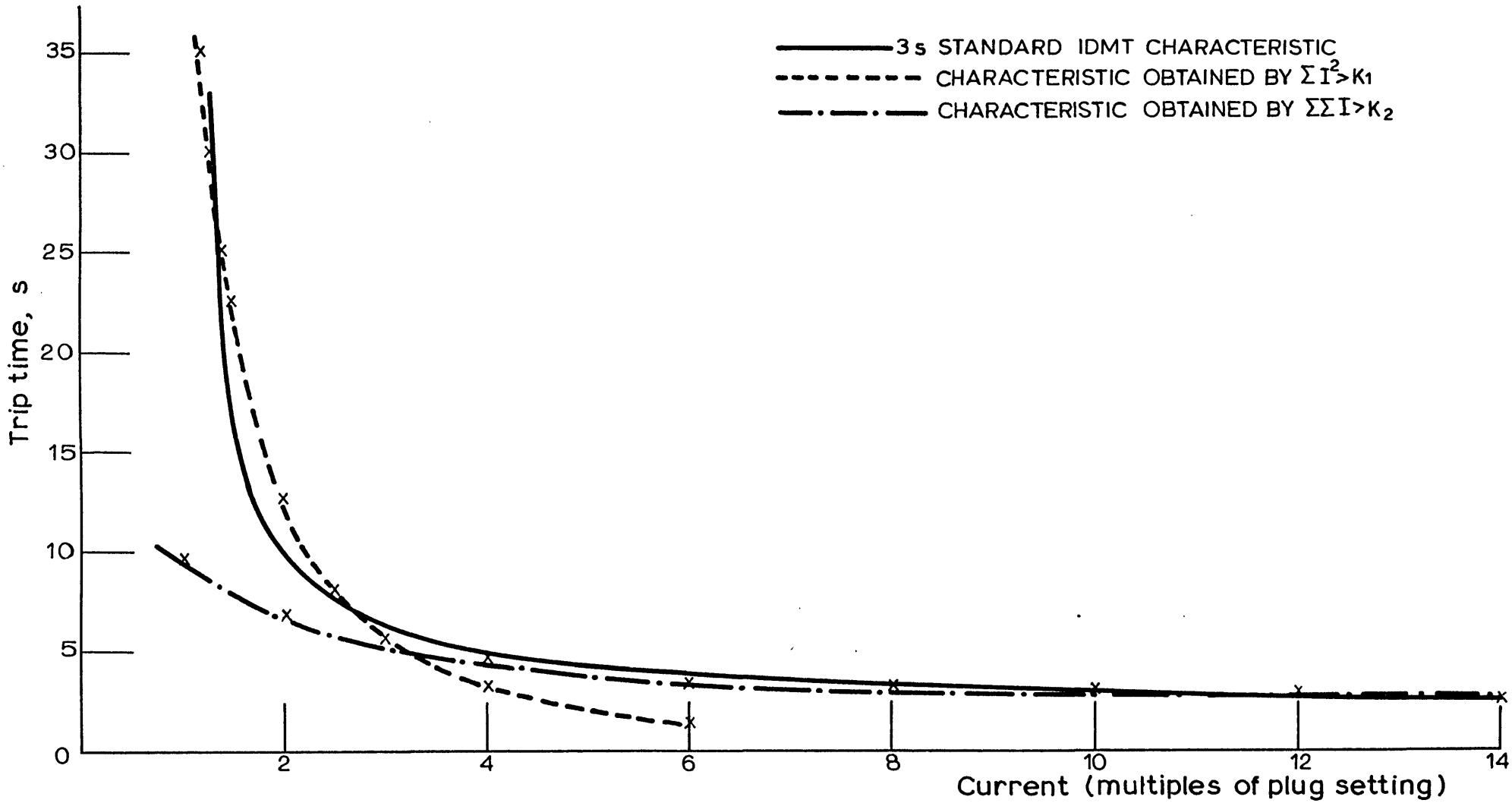


Fig. 6·3 Simulation results for IDMT characteristic matching.

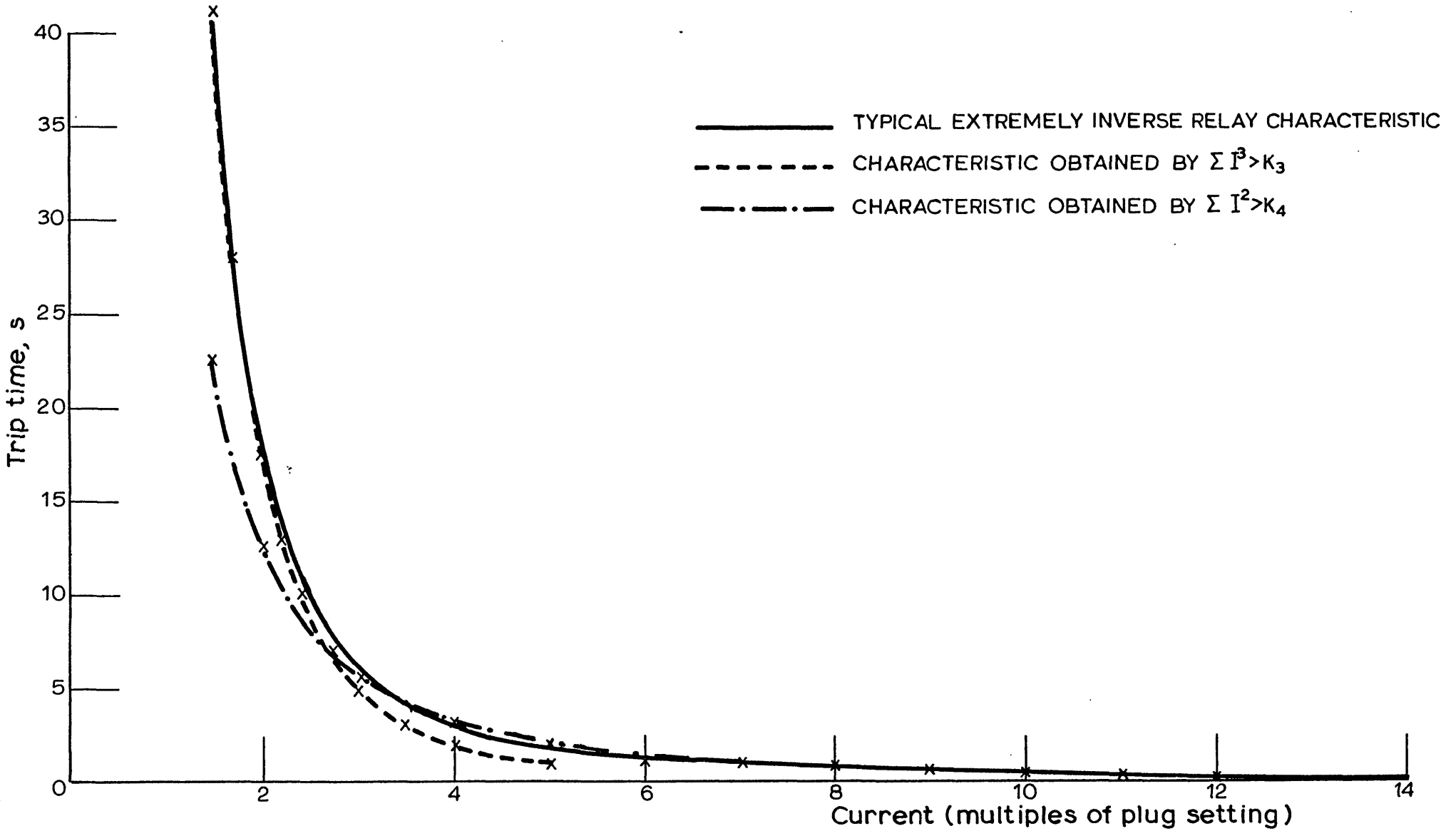


Fig. 6·4 Simulation results for E.I. characteristic matching.

the appropriate function of $I_{A,(BC)P}$. A summation is then performed the result of which is compared with the corresponding limit value. A portion of routine EXIN reduces the summation of I^3 terms, if used, by integer multiples of 1000 at each iteration. This operation allows the constant K_3 (288000) to be stored as 288 within a single 16 bit word.

6.3.10 Data output

A general sub-routine, DUMP, enables the relay to output data and messages to monitoring or back-up protection computers via the communication port. Specific data sources for DUMP are determined by the status of several bits in the program control word.

6.4 Relay Test Hardware

6.4.1 Transmission line model.

The test configuration used for the O/C - E/F relay is shown in Fig.6.5. A transmission line model fully instrumented for distance protection research formed the central feature of the 3 phase equipment. By virtue of its higher load (20A) and fault (100A) current capability and the use of conventional iron cored current transformers, the transmission line model enabled more realistic testing of the relay than would have been possible with the complete substation model employed in Chapter 5.

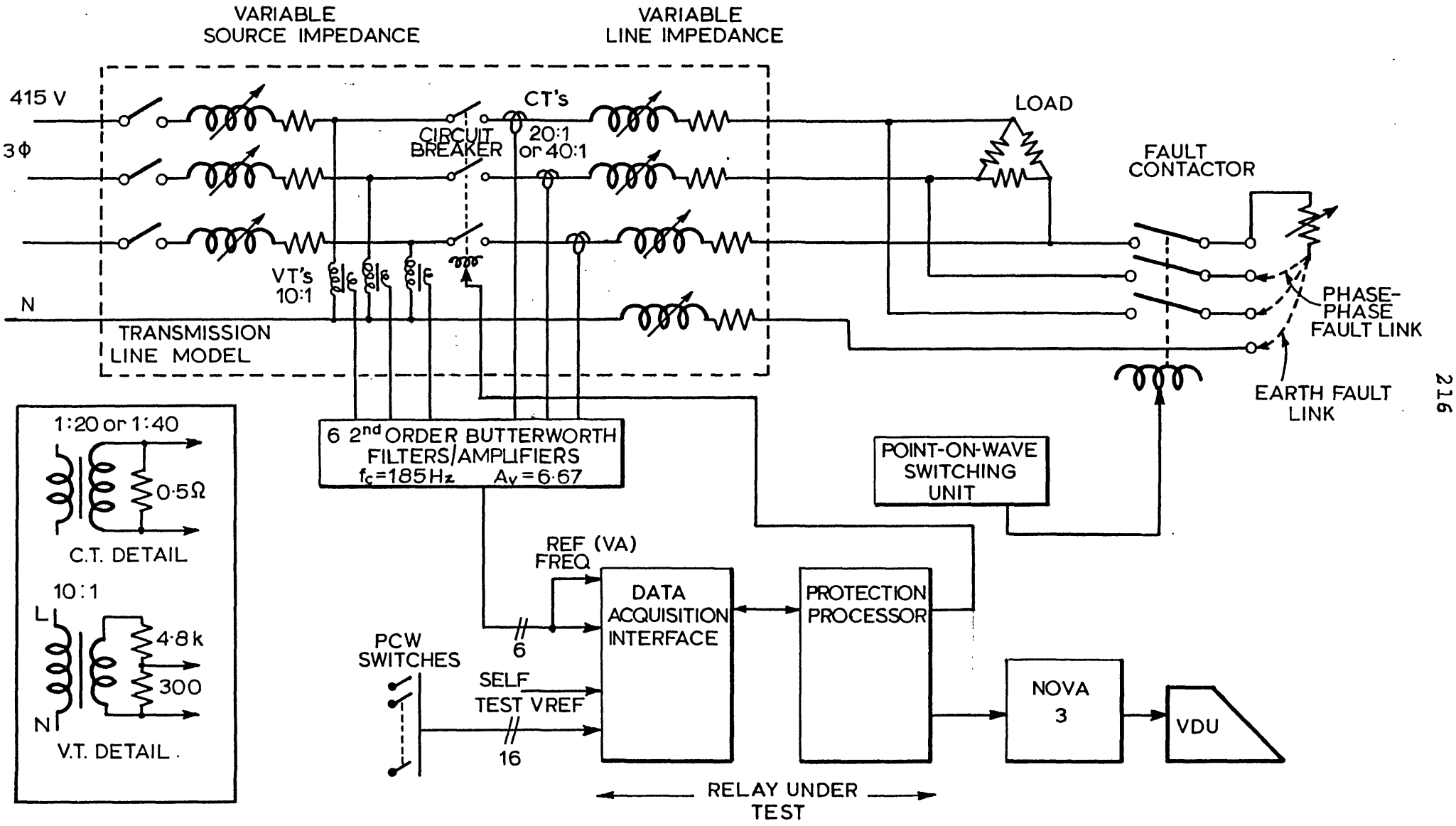


Fig.6.5 Overcurrent-earth fault relay test hardware

Fault conditions were applied to the 3 phase system by a contactor under the control of a"point on wave" switching unit. The system circuit breaker comprised a vacuum contactor with a 200A rating, controlled by the relay.

The relay communication port was connected to a mini-computer to facilitate performance monitoring and data recording.

6.4.2 Relay inputs

Current and voltage data were obtained using the transducers, filters and amplifiers designed and connected to the line model for distance protection research.

The relay inputs comprised:

i) 3 phase currents derived from the current transformers. These transformers, with an operating ratio of 20:1, were provided with a 0.5Ω resistive burden. Filtering and amplification of the current signals produced an overall transfer ratio of 0.167V/A.

ii) 3 phase-neutral voltages supplied by voltage transformers in the line model. A 1 p.u. phase-neutral voltage of 240V gave a relay input of 3.0V after attenuation and filtering.

iii) Sampling clock reference signal, obtained from the "A" phase voltage measurement channel of the equipment.

iv) Self test reference voltage.

v) Program control word.

The provision of inputs (iv) and (v) is described in Chapter 5.

6.5 Test Results

6.5.1 Earth fault tests

Earth faults were simulated by connecting a resistive load from one phase to neutral of the transmission line model. Adjustment of this load allowed the fault severity to be varied. Throughout the tests the line model supplied a balanced 15A (1p.u.) load.

Fig. 6.6 shows the computed RMS value of the imbalance current for a range of earth faults up to a maximum of 1.5 p.u. i.e. 22.5A. The values plotted in this figure were derived from a 50 sample averaging routine stored in the monitoring mini-computer. Inspection of Fig.6.6 indicates the presence of a 0.06 p.u. RMS value which exists for unfaulted conditions (i.e. 1 p.u. balanced load). The influence of this error, which is mainly attributable to offsets and gain mismatch in the analogue filter-amplifiers and non-linearities in the current transformers, is discussed in Section 6.3.6.

The transient response of the relay to suddenly applied earth faults is illustrated by Figs. 6.7 and 6.8. These figures were recorded for 0.5 p.u. and 2.0 p.u. earth faults respectively. The tripping functions of the

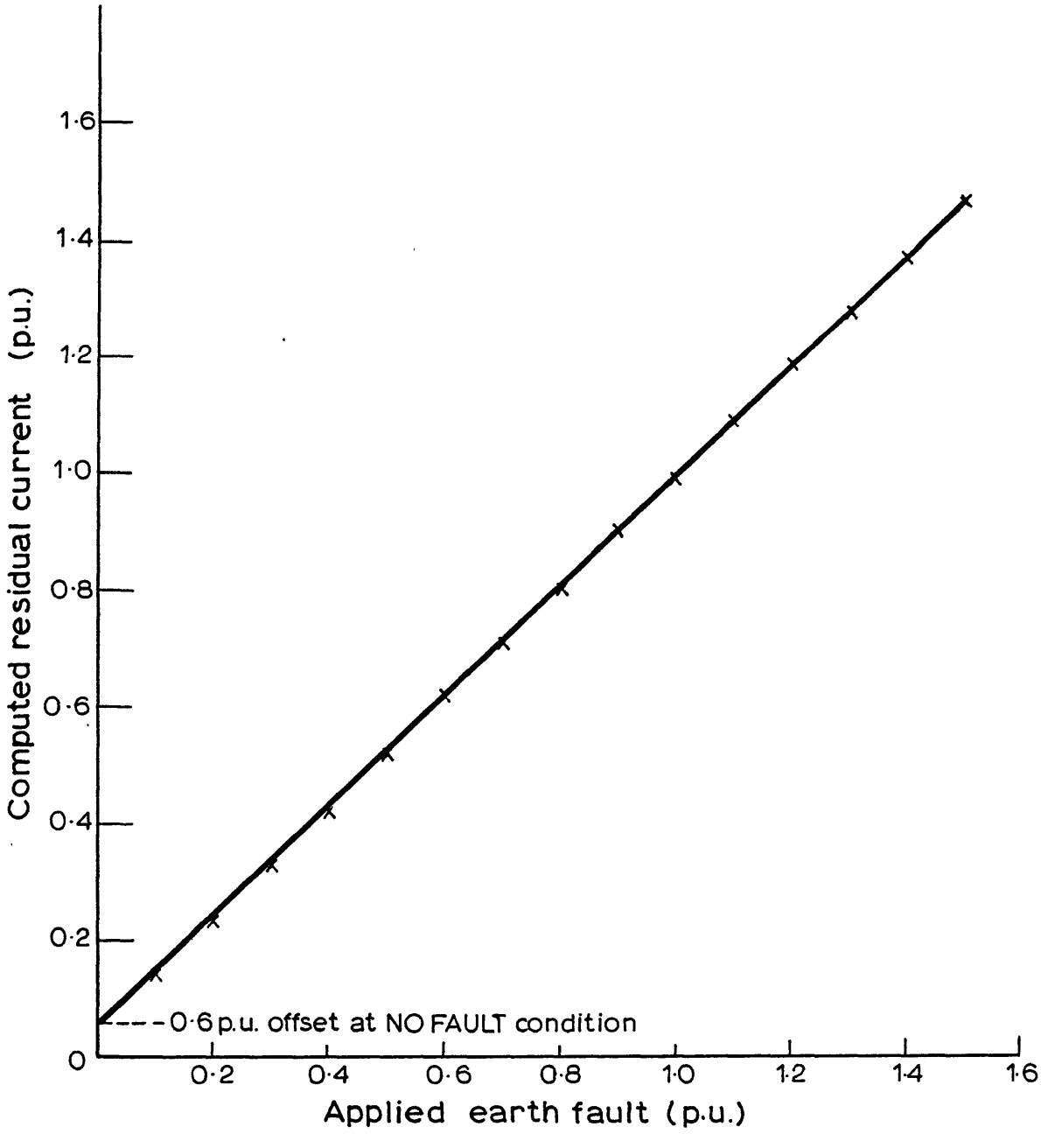
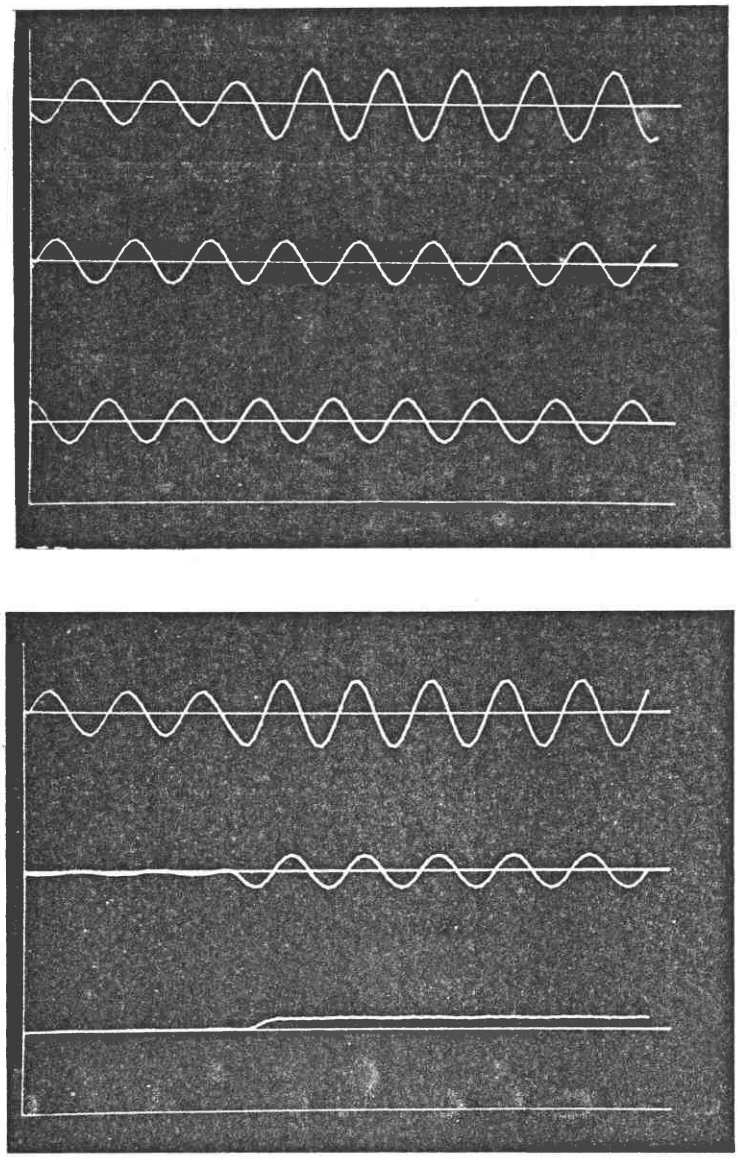


Fig. 6·6 Actual earth fault magnitude and computed residual current.



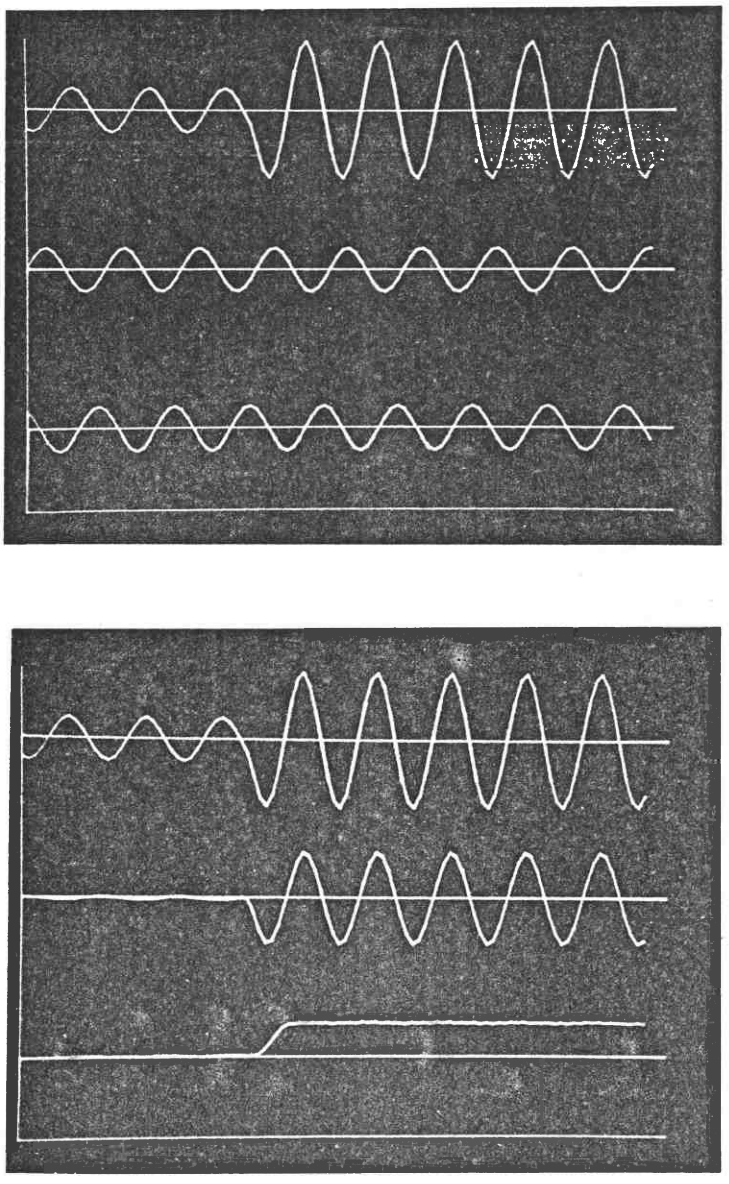
(a) Faulted primary current waveforms

(b) Faulted phase current

(c) Computed residual current

(d) RMS of residual current

Fig. 6·7 Transient response of the earth fault relay for a 0·5 p.u. fault.



(a) Faulted primary current waveforms
(b) Faulted phase current
(c) Computed residual current
(d) RMS of residual current

Fig. 6·8 Transient response of the earth fault relay for a 2·0 p.u. fault.

relay were inhibited during this test.

Figs. 6.9 and 6.10 show the tripping performance of the relay. In these figures trace (b) indicates the point of fault application and was derived from auxillary contacts on the fault contactor.

Fig.6.9 was recorded for a 0.5 p.u. fault magnitude. The extended clearance times involved when the relay invokes the 64 sample averaging portion of the earth fault software is clearly illustrated in Fig.6.10. In this case a 0.05 p.u. earth fault was applied.

From both of these examples it can be seen that the vacuum contactor "circuit breaker" requires approximately 16 cycles to operate.

Finally, Fig.6.11 demonstrates the performance of the earth fault section of the relay during a 2.0 p.u. phase-phase fault. Clearly the residual current routine is unaffected by this type of fault and the relay remains in a stable condition.

6.5.2 Overcurrent tests

The overcurrent sections of the relay were set to operate with a 1 p.u. primary current of 5A. To avoid saturation of the current transformers. (secondary rating = 1A) and the analogue filters during these tests, the current transformer ratio was increased to 40:1. P_S, the plug setting evaluated from expression 6.3.2 was thus:

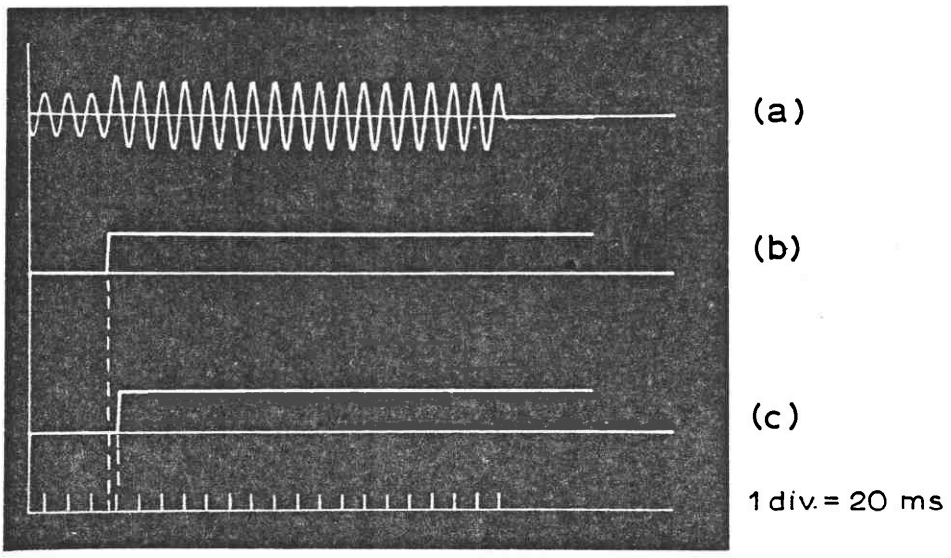


Fig. 6·9 Tripping performance for 0·5 p.u. earth fault.

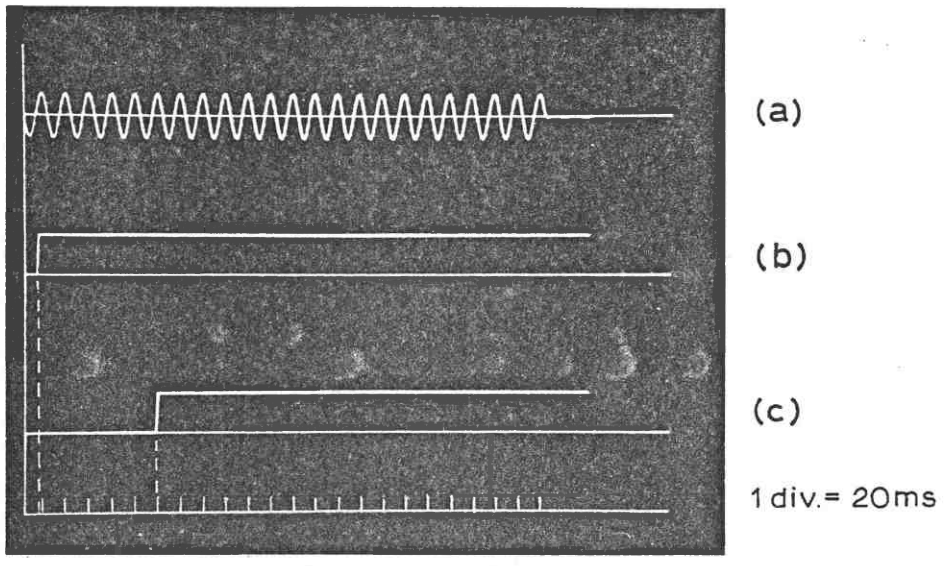
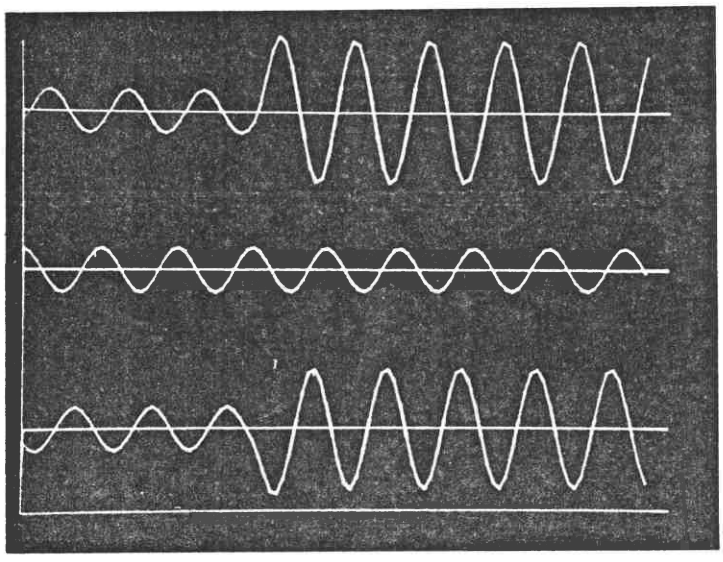
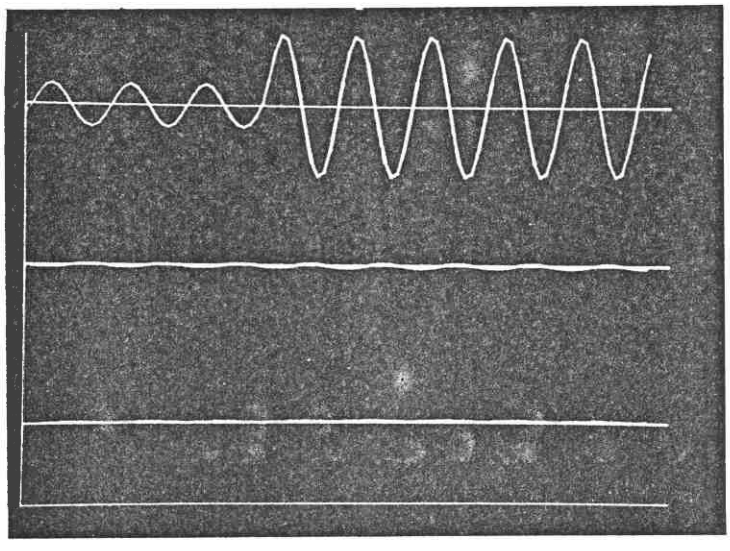


Fig. 6·10 Tripping performance for 0·05 p.u. earth fault.

(a) Faulted phase current
(b) Point of fault application
(c) Relay "trip command" output





(a) Faulted primary current waveforms
(b) One of the faulted phase currents
(c) Computed residual current
(d) RMS of residual current

Fig. 6·11 Earth fault relay performance during a 2·0 p.u. phase-phase fault.

$$P_S = \underset{I_S}{5.0} \times \underset{A_{CT}}{0.0125} \times \underset{A_F}{6.67} \times 51.2 = 21.33$$

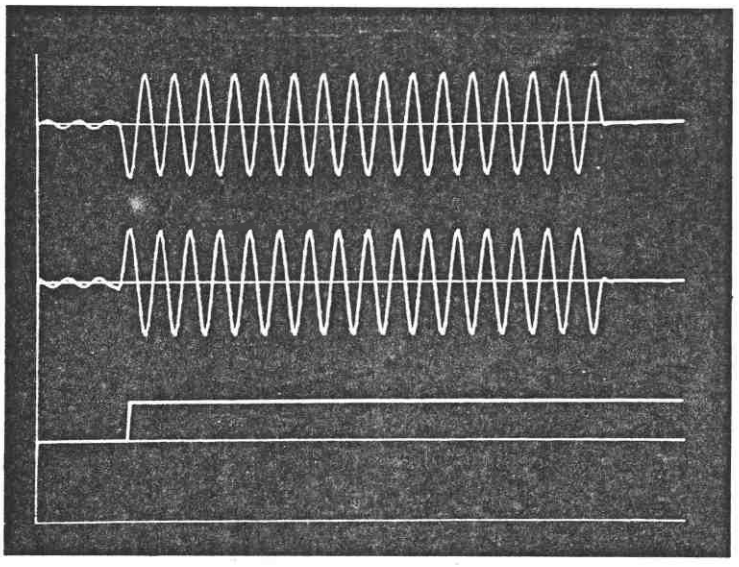
i.e $P_S = 21$ as an integer value

Fig. 6.12 shows the operation of the "high set" instantaneous O/C function for a 15 p.u. (75A) phase-phase fault. As previously described, a high set limit of 8p.u. was used, corresponding to a primary current of 40A with the above relay plug setting.

The measured IDMT relay tripping times are plotted in Figs. 6.13 and 6.14 for TMS settings of 1.0 (TMS sub-routine parameter = 100) and 0.5 (TMS parameter = 50) respectively. In both cases the appropriate standard IDMT characteristics are shown on the figures for comparison.

Fault currents were limited to 10p.u. (50A) during these tests. This restriction was imposed to protect both the transmission line model and its supply in view of the relatively long clearance times provided by the IDMT function.

Finally the operation of the relay with an extremely inverse characteristic (TMS = 1.0) is illustrated in Fig.6.15. Again a typical analogue relay characteristic is included to allow a comparison of performance.



(a),(b) Faulted phase current waveforms
I_{FAULT} = 75 A

(c) Relay trip command output

Fig. 6·12 High-set O/C relay operation for a 15 p.u. fault.

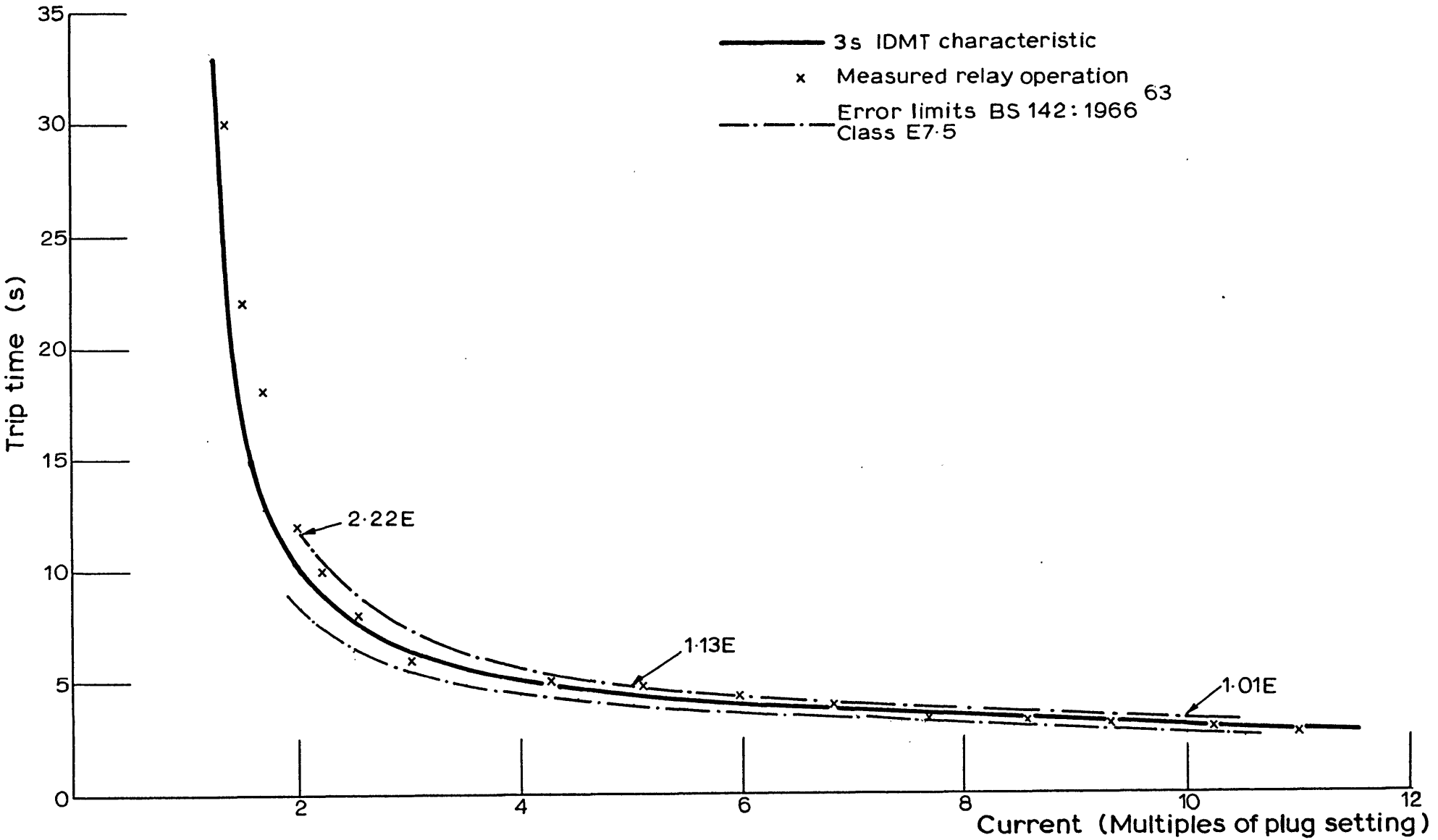


Fig. 6·13 O/C Relay performance for TMS = 1·0

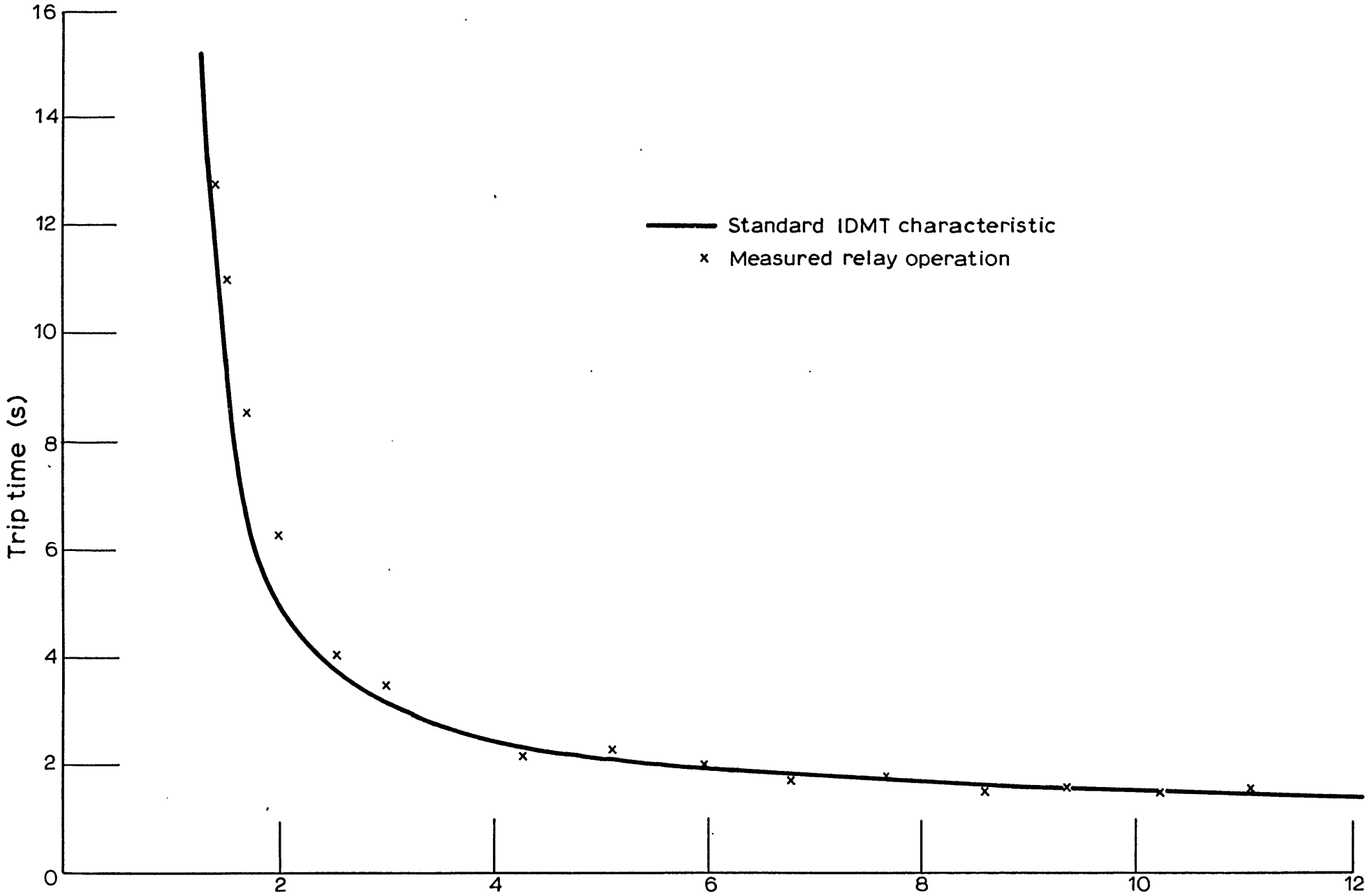


Fig. 6·14 O/C Relay performance for TMS = 0·5

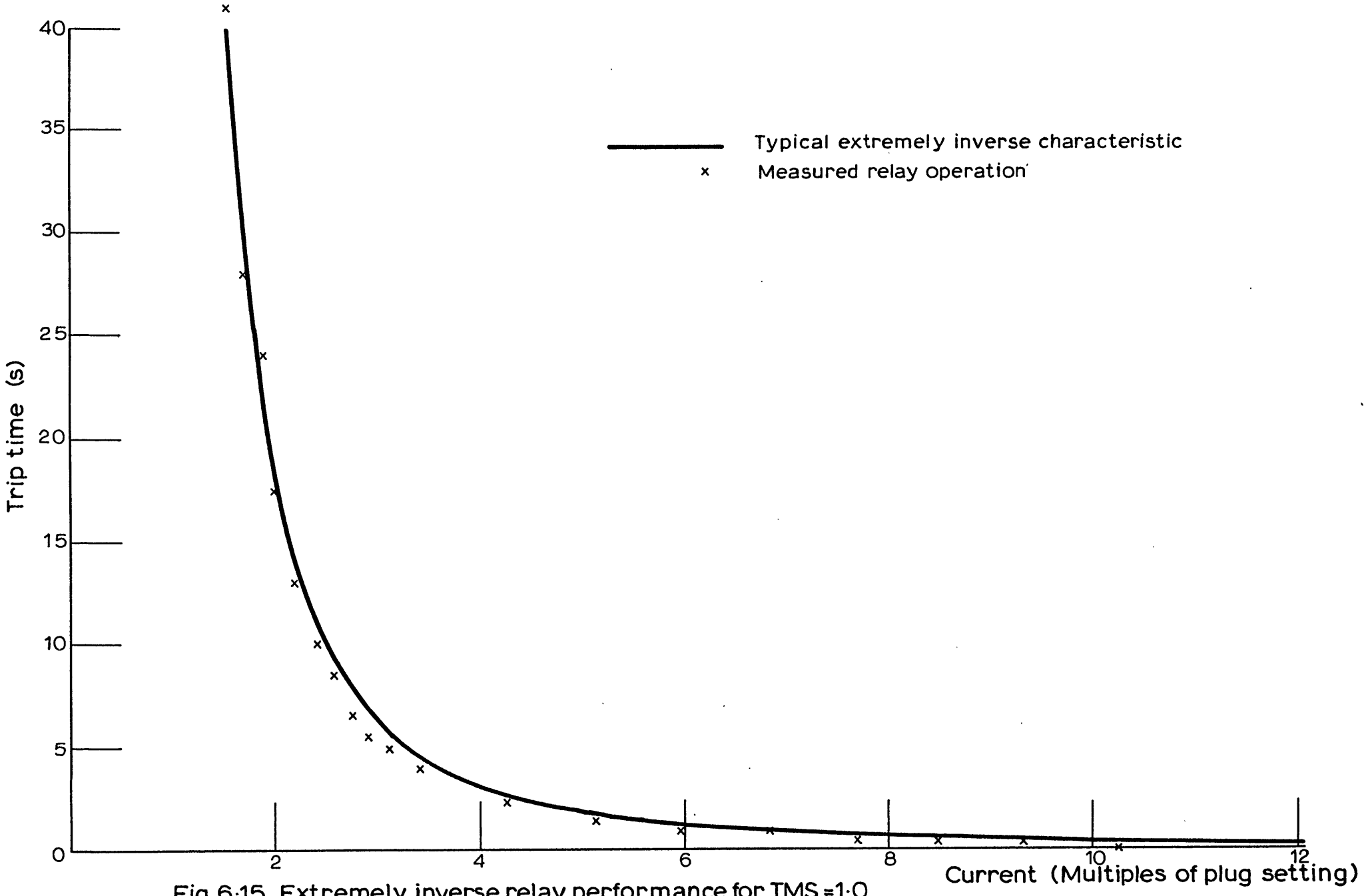


Fig. 6·15 Extremely inverse relay performance for TMS = 1·0

6.5.3 Auto-reclose test.

Circuit breaker operation during a typical auto-reclose cycle is shown in Fig.6.16 A 1.8 p.u. phase-phase fault was applied to initiate tripping via the O/C section of the relay with the IDMT (TMS = 1.0) characteristic selected. Tripping time in this example is approximately 15s.

6.6 CONCLUSIONS

The O/C, E/F application demonstrates the feasibility of combining two closely related relaying functions in a single digital equipment. By providing a range of pre-programmed, optional, tripping characteristics the flexibility of the relay is further enhanced.

From the test results it can be seen that the IDMT characteristic does not exactly match the conventional standard. However, since relay grading is of major importance in many O/C applications the precise repeatability of the digital characteristic is of great value. Alternative, and possibly optimum, characteristics derived by digital processing remain to be investigated. In other respects the digital relay is superior to its analogue counterparts. It produces a negligible overshoot, and has a minimal reset time. The relay "pick up" current may also be defined within close limits.

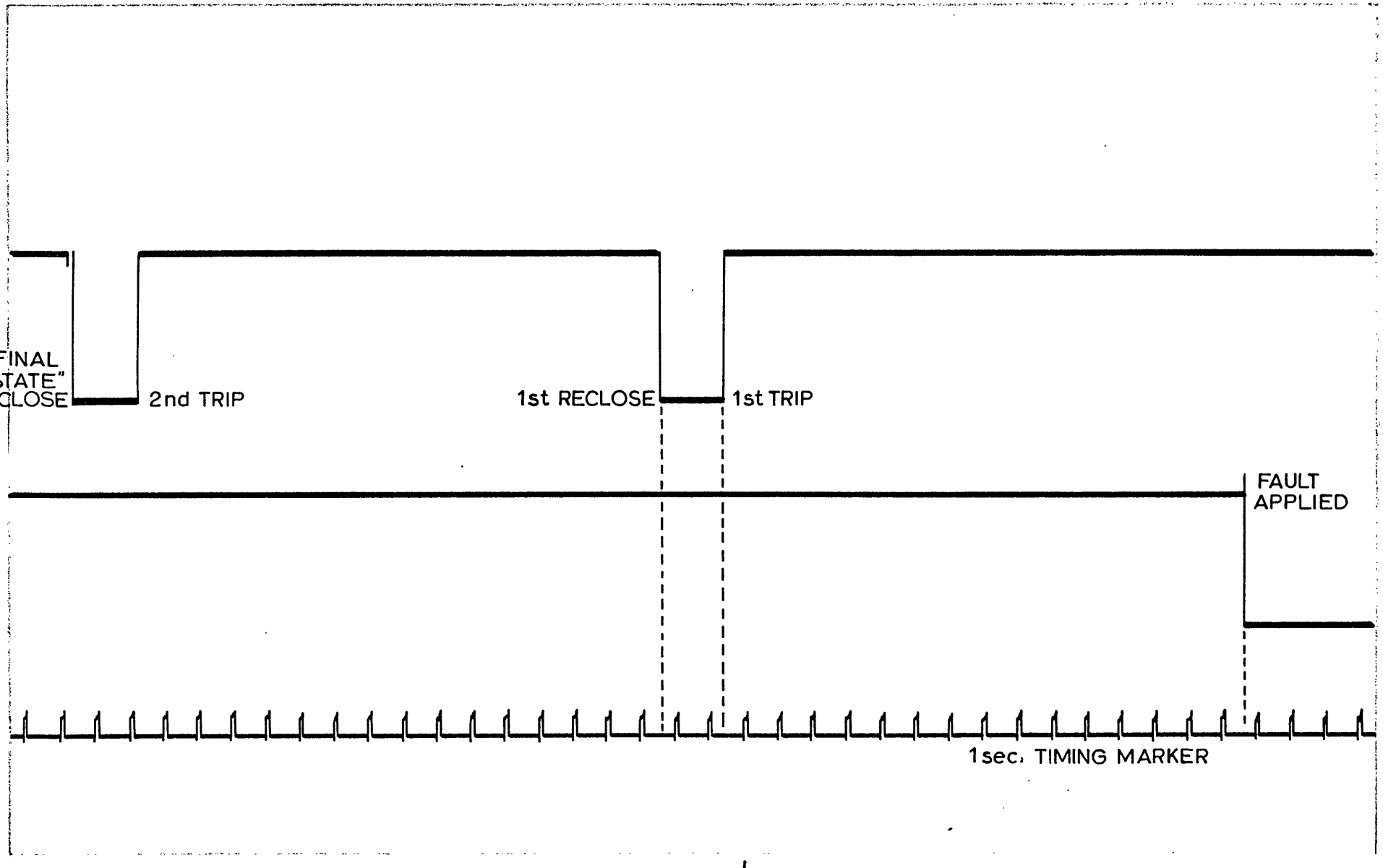


Fig. 6·16 Auto reclose operation. Relay tripped by O/C fault of approximate magnitude 1·8 p.u. IDMT characteristic TMS =1·0. *Note: Time scale from R to L.*

Clearly, additional logic associated with the auto-reclose function could be easily incorporated in the software.

The software structure described in this chapter provides an attractive starting point for the development of high level relay programming techniques. By replacing the sub-routine calls used in the main program with appropriate high level language statements, application programs for the equipment could be readily defined and understood. Several essential functions, e.g. self test and circuit breaker control, could be usefully pre-programmed for all relay applications. Further routines, applicable to specific tasks, would then be selected and programmed with the aid of an off-line "compiler". This area provides considerable scope for future development.

CHAPTER 7

CONCLUSIONS

7.1 General Conclusions.

This thesis has described a new approach to the construction of protective relays using digital electronic methods. Microprocessor based relays can provide an economically realistic solution for the practical application of digital power system protection techniques. In addition they offer many benifits, notably improved flexibility, when compared with existing static analogue equipments.

In Chapter 2 the major advantages and disadvantages of digital protection schemes were discussed. To fully utilise the investment in hardware, an "integrated protection" approach is generally adopted for mini-computer based schemes, in which several protection functions are performed by a single computer. It was argued that this approach places additional constraints on the protection performance which could prove unacceptable for widescale primary relaying applications.

To overcome these limitations an alternative scheme was proposed in which existing electromechanical and static analogue relays are replaced by dedicated digital equipments constructed with microprocessors. For maximum

flexibility the digital relay was designed to comprise two component units: i) the data acquisition interface and ii) the protection processor.

The design considerations and performance criteria of the relay data acquisition unit were examined in Chapter 3. This unit provides an interface between the relay and the power system transducers. The analogue signal sampling and conversion components used in the interface are a large portion of the total relay cost and must be carefully specified.

It was shown that a 10 bit A-D converter is adequate for protection tasks provided that:

i) a sample-hold device is included in each analogue signal channel to eliminate "rate of change" errors during conversion.

ii) the signal input to the converter is conditioned to prevent saturation occuring for high level inputs, or excessive signal/noise and resolution errors for low level signals. A design for a suitable signal ranging amplifier was given.

The use of conversion systems with higher accuracies (i.e. 12 or 14 bit word lengths) cannot be justified unless primary transducer accuracies are improved.

A simple communication structure combined with a programmable controller and modular construction techniques,

were employed to enable the interface to economically satisfy a wide range of input requirements.

In Chapter 4 the design of the protection processor was described.

From studies of previous digital protection research, it was possible to identify a suitable specification for the data processing portion of the relay. It was concluded that a mini-computer word length i.e. 12-18 bits and instruction execution time i.e. 1-3μs are desirable characteristics for the processor. Further, a program storage capacity of 2K instructions and approximately 200 data memory locations were indicated by the study.

It was found that the word length and speed requirements could only be satisfied by bipolar bit-slice micro processors. A design for the protection processor employing bit-slice components was presented. To cater for the special demands of the protection task, the design featured:

i) 16 bit word length, obtained by the use of 8, 2 bit-slice devices.

ii) specialised architecture in which program and data memory addressing is seperated.

iii) extensive use of pipeline registers, which, in conjunction with (ii) above, enabled a nominal

instruction execution time of 1μs to be achieved.

iv) microprogrammable control giving great flexibility of instruction set design.

v) a maximum program capacity of 2K instructions, stored in high density, low cost, non-volatile semiconductor EPROMS.

vi) a data/constant memory structure providing storage for 512 data and 256 constant words i.e. a maximum of 768 locations.

vii) input and output ports to provide circuit breaker control and communication with external equipment.

viii) typical mini-computer assembler level instruction set with several additions related to the specialised functions of the equipment.

ix) reliable and predictable restart following power supply failure.

The design considerations which gave rise to the above features were discussed for each part of the processor.

The application examples described in Chapters 5 and 6 were undertaken to verify and demonstrate the design philosophy of the overall relay.

In Chapter 5 a new and straightforward method was presented for unbalanced load protection of generating plant.

The digital implementation which provides all the facilities associated with existing analogue unbalanced load relays was shown to be well within the design capabilities of equipment. In several areas, the performance of the protection is improved. Notably, the digital relay offers: i) a detection accuracy which is unaffected by normal variations in the power system frequency, ii) more comprehensive alarm reporting facilities and iii) continuous internal self testing.

Chapter 6 described an overcurrent and earth fault protection application.

The relay software for this application was designed on a modular basis with routines to provide i) overcurrent detection ii) earth fault detection iii) instantaneous (high set), IDMT, and EI tripping characteristics iv) self test v) fault reporting and vi) simple auto-reclose. External control data was then used to select these pre-programmed routines for specific relay functions.

It is proposed that this approach to application software design is adopted since it enhances the flexibility of the digital relay. The initial specification and set-up of the relay are simplified, as are subsequent modifications if the protection task requirements change.

7.2 Original Contributions.

The following original contributions are presented by this thesis:

i) A critical review of mini-computer based "integrated" protection schemes and the identification of their limitations.

ii) The proposal of an alternative approach using dedicated microprocessor based relays which overcomes these limitations for practical applications.

iii) A detailed analysis of the performance parameters required in the data acquisition interface of a dedicated relay.

iv) The architectural design of a specialised, high performance bit-slice microprocessor, capable of providing the computational needs of a digital relay.

v) The development of an algorithm for unbalanced load protection of generators, and the on-line testing of the dedicated relay equipment for this task.

vi) The development of digital methods for overcurrent and earth fault protection. A software structure is demonstrated in this application which enables relay manufacturers and users to readily specify a wide range of optional functions.

7.3 Further Research.

The most apparent area for further research using the digital relay which has been constructed concerns the

implementation and testing of various protection functions. Transmission line, bus zone and transformer protection algorithms which have been developed during previous research are an obvious starting point for these applications. The facilities offered by the digital relay also allow new methods to be investigated. For example, the ability to store pre-fault measurements provides new possibilities for the development of distance relays which satisfy the requirements of networks involving "Teed" feeders. An advanced generator protection relay which exploits stockastic techniques to detect obscure internal machine faults is a further area worthy of study.

In a wider field, the application of the relay as part of a combined integrated-dedicated protection scheme and the overall structure of such a scheme gives rise to many research topics. These include inter-processor communication methods and a study of the complete system timing requirements.

The development of application programs for the relay would be greatly simplified by the provision of more extensive off-line software support aids. A comprehensive error detecting assembler, text editor and relay simulator package would be very valuable.

Additional benefits would be gained if the relay programs could be described in a high level task oriented language. For example, operations such as data filtering,

circuit breaker control and relay timers might usefully be implemented as micro-program routines.

From a hardware view-point, relay testing has, to date, been performed in a laboratory environment. An evaluation of the equipment performance under realsitic power system conditions is required.

Further development of the protection processer, using "single chip" 16 bit bipolar microprocessor components, is possible when these devices become available. This approach would undoubtedly reduce the complexity and hence cost of the relay. However, it must be noted that the pre-defined instruction set of the single chip micro-processor does not allow the software flexibility which is available with the bit-slice design presented.

REFERENCES

1. Warrington,A.R.van C.: "Protective Relays.-Their Theory and Practice". Vol.1 Chapman & Hall, 1968.

2. G.E.C.Ltd.: "Protective Relays Application Guide". G.E.C. Measurements Ltd., 1975.

3. Seymour, C. : "Statistics of C.E.G.B. protection performance - 1968-73". I.E.E. Conference on developments in power system protection. Conference publication No.125, March 1975.

4. Crook, D.W.E. and Harris, J.H. : "The use of on-line computing facilities at the national control centre of the C.E.G.B.". Symposium on Implementation of Real-Time Control by Digital Computer. Imperial College, London, 1973.

5. Patterson, D.: "Application of a computerised alarm-analysis system to a nuclear power station". Proc. IEE, Vol.115, December 1968.

6. Naylor, J.H.: "Trends in automation in HV and EHV substations". Electra No.52, May 1977.

7. Allison, P.B and Lomas, T.H.: "The application of a microprocessor system to automatic switching and reclosing on power supply networks". I.E.E. Conference on developments in power system protection, March 1975.

8. Rockefeller, G.D.: "Fault protection with a digital computer". IEEE transactions Vol. PAS-88, No.4, April 1969. Paper presented at IEEE Summer Power Meeting, June 1968.

9. Ranjbar, A.M.: "Computer protection of high voltage transmission lines". Ph.D Thesis. Imperial College, London, 1975.

10. Mann, B.J. and Morrison I.F.: "Digital calculation of impedance for transmission line protection". IEEE transactions Vol. PAS-90, No.1, Feburary 1971.

11. Ranjbar, A.M. and Cory, B.J.: "An improved method for the digital protection of high voltage transmission lines". IEEE transactions, Vol. PAS-94, No.2, March/April 1975.

12. Cory, B.J. and Mount,J.F.: "Application of digital computers to busbar protection". IEE Conference, Bournemouth, 1970.

13. Ong, P.K.S.: "Digital protection of transformers". M.Sc. thesis, Imperial College, London, 1974.

14. Sykes, J.A. and Morrison, I.F. : "A proposed method of harmonic restraint differential protection of transformers by digital computer". IEEE transactions, Vol.PAS-91, No.3, May/June 1972.

15. Hope, G.S., Dash, P.K. and Malik, O.P.: "Digital differential protection of a generating unit: scheme and real time test results". IEEE transactions, Vol. PAS-96, No.2, March/April 1977.

16. Connelly, K., Scott, A.J.M., Campain, A.J. and Morrison,I.F.: "Integrated protection and control in substations". CIGRE paper 34-07, August 1974.

17. Edgley, R.K.: "Back-up protection and overall protection of power systems networks". Electra No.34, August 1974.

18. Traca A.T.C.: "Digital computer switching of high voltage substations". Ph.D. thesis, Imperial College, London 1977.

19. Gonzales, G.E.: "Data validation and reliability calculations in digital protection systems". Ph.D thesis, Imperial College, London 1976.

20. Horne,E. and Cory, B.J.: "Digital processors for substation switching and control". IEE Conference on developments in power system protection, March 1975.

21. Electricity Council: "Power system protection", Vol.1, 2,3. Macdonald & Co.Ltd. 1969.

22. Possner, O.G.: "New trends in the arrangement, accommodation and interconnection of protective relays and associated devices". Electra No.54, October 1977.

23. Cory, B.J., Dromey, G. and Murray, B.E.: "Digital systems for protection". CIGRE paper 34-08, August 1976.

24. C.E.G.B.: "Operation diagram Kingsnorth 400 kV sub-station". C.E.G.B. Transmission department. 1967.

25. Lomas, T: "Digital equipment in electrically hostile environments". Post experience course on digital protection of power systems. Imperial College, June 1976.

26. Manchur, G., Harvey, S.M. and Iwanusiw, O.W.: "Development of line protection with solid state and digital techniques". I.E.E. Conference on developments in power system protection. March 1975.

27. Dromey, G.: Personal Communication.

28. Tou, J.T.: "Digital and Sampled-data Control Systems". McGraw-Hill 1959.

29. Hybrid Systems Corporation.: "S/H725 Series, Sample Hold". Device data sheet. Hybrid Systems Corporation. Bulletin No. 725-2-3. June 1973.

30. Bruck, D.B.: "Data Conversion Handbook". Hybrid Systems Corporation. 1974.

31. Burr-Brown.: "C-Mos Analog Multiplexers". Device data sheet and application note. Burr-Brown Research Corporation. Note No. PDS-315A. June 1975.

32. Hoeschele, D.F.: "Analog to Digital/Digital to Analog Conversion Techniques ". Wiley 1968.

33. Uzunoglu, V.: "Analysis and design of digital systems". Gordon and Breach 1975.

34. Sachdev, M.S. and Hunchuk, D.G.: "Performance of analog to digital converters used in computer relaying". IEEE A76 154-5. Winter power meeting. January 1976.

35. British Standards Institution.: "Current Transformers" B.S. 3938, 1973. "Voltage Transformers" B.S. 3941, 1975.

36. Burr-Brown.: "Model ADC 85 Hybrid A/D Converters". Device data sheet and application note. Burr-Brown Research Corporation. Note No. PDS-320. January 1975.

37. Springer, J.: "Application of First-In First Out Memories". Advanced Micro Devices, application note. September 1974.

38. Jones, J.L. and Greco, L.S.: "An application of mini-computers to the control of E.H.V. substations. Unpublished paper.

39. Gardner, F.M.: "Phase lock Techniques". Wiley 1966.

40. Viterbi, A.J.: "Principles of Coherent Communication." McGraw-Hill 1966.

41. Grebene, A.B.: "The monolithic phase-locked loop - a versatile building block". IEEE Spectrum. March 1971.

42. Luckett, R.G., Munday, P.J. and Murray, B.E.: "A substation based computer for control and protection". I.E.E. Conference on developments in power system protection. March 1975.

43. Mann, B.J. and Morrison, I.F.: "Relaying a three phase transmission line with a digital computer". IEEE transactions Vol. PAS - 90, No.2, March/April 1971.

44. Penney, B.K.: "The implications of microprocessor architecture on speed, programming and memory size". The Radio and Electronic Engineer. Vol.47, No.11, November 1977.

45. Hoff, M.E.: "Designing central processors with bipolar microcomputer components". Intel Corporation, application note. Presented at the National Computer Conference, 1975.

46. Colin, A.: "Intel 3000 and AMD 2900 microprocessors - a comparison". Microprocessors, Vol.1, No.5, June 1977.

47. Intel Corporation.: "3002 Central Processing Element". Device data sheet. Intel Corporation. 1975.

48. Intel Corporation.: "3003 Look-Ahead Carry Generator". Device data sheet. Intel Corporation. 1975.

49. Wilkes, M.V. and Stringer, J.B.: "Microprogramming and the design of the control circuits in an electronic digital computer". Proceedings of Cambridge Phil.Soc., Vol. 49, part 2. April 1953.

50. Texas Instruments Ltd.: "Series 54/74, 54S/74S Programmable Read-Only Memories". Device data sheet. Texas Instruments bulletin, No. DL-S7512258. May 1975.

51. Intel Corporation. "3001 Microprogram Control Unit". Device data sheet. Intel Corporation. 1975.

52. Texas Instruments Ltd.: "TMS2708JL 1024- Word 8-Bit Erasable Programmable Read-Only Memories". Device data sheet. Texas Instruments. January 1977.

53. Texas Instruments Ltd.: "TMS4036 JL, NL, 64-Word By 8-Bit Static Random-Access Memories". Device data sheet. Texas Instruments bulletin, No. DL-S7512277. May 1975.

54. Fielland, G.: "Series 3000 System Timing Considerations". Application note. Intel Corporation. 1975.

55. Brown, P.G.: "Generator I_2^2t Requirements for System Faults". IEEE transactions Vol.PAS-92, July/August 1973.

56. Arnold, J.J.: "The protection of generators against negative sequence current". I.E.E. Conference on developments in power system protection. March 1975.

57. Gschwind, H.W.: "Design of Digital Computers". Springer-Verlag 1967.

58. Ackroyd, M.M.: "Digital Filters". Butterworth. 1973.

59. Knowles, J.B. and Olcayto, E.M.: "Coefficient accuracy and digital filter response". IEEE transactions Vol. CT-15, No.1. March 1968.

60. Clarke, E.: "Circuit analysis of a-c power systems". Wiley 1943.

61. Hamming, R.W.: "Numerical Methods For Scientists And Engineers". McGraw-Hill 1962.

62. Minisey, S.M.: "Integrated digital protection and control of high voltage substations". Power System Report, No.94. Electrical Engineering Dept., Imperial College. September 1973.

63. British Standards Institution: "Electrical Protective Relays". B.S. 142, 1966.

64. Texas Instruments Ltd.: "The TTL Data Book for Design Engineers". Texas Instruments. 1976.

Introduction to appendices 1,2.

Appendices 1, 2 give detailed circuit diagrams and relevant hardware information for the units of the prototype digital relay. Appendix 1 relates to the data acquisition interface whilst Appendix 2 covers the protection processor.

Since the main operational features of the relay modules are described in the appropriate chapters of this thesis no further description is included in these appendices. The following notes are applicable to both appendices:

i) Data sources for major integrated devices used in the design are listed as references in the main chapters.

ii) Other 74 series TTL devices are fully described in reference 64.

iii) For clarity the Ω symbol has been omitted from resistor values. The notation employed is:

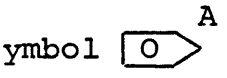
100	value in ohms.
4.7K	value in ohms x 10^3
3.9M	value in ohms x 10^6

All fixed resistors are metal oxide 5% 0.33W unless otherwise indicated.

iv) Capacitors are identified by a dual number notation of the form A/B, where A- value (micro-farads),

B - working voltage (volts). In some cases the suffix p is appended to the A value e.g. 100p/50 to indicated that this quantity is in picofarads.

v) The symbol  indicates a printed circuit board edge connector.

vi) The symbol  indicates a wire link from the printed circuit board.

vii) A positive logic convention is used throughout the circuit diagrams. "Active low" signals are thus labelled:

SIGNAL NAME

APPENDIX 1

Data acquisition system.

Drawings and details.

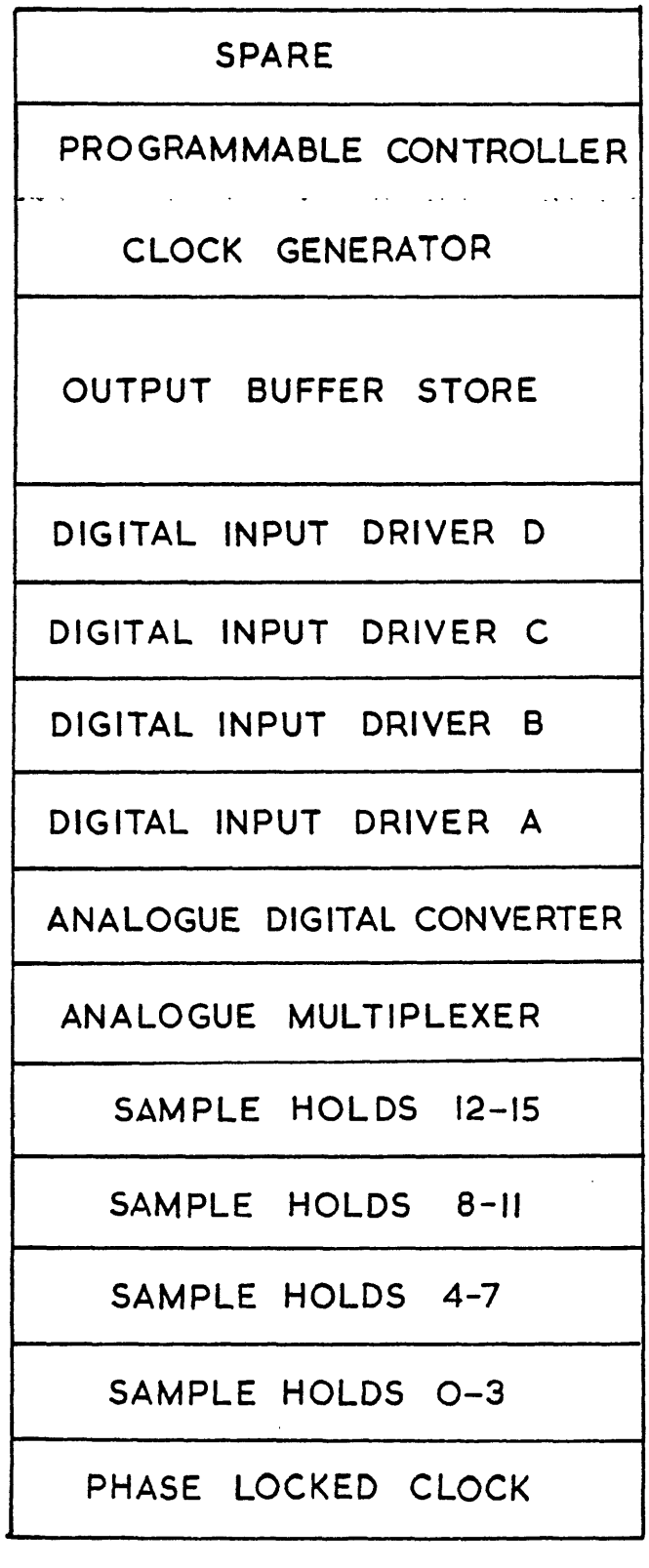


FIG A1.1 ACQUISITION SYSTEM RACK MODULE POSITIONS

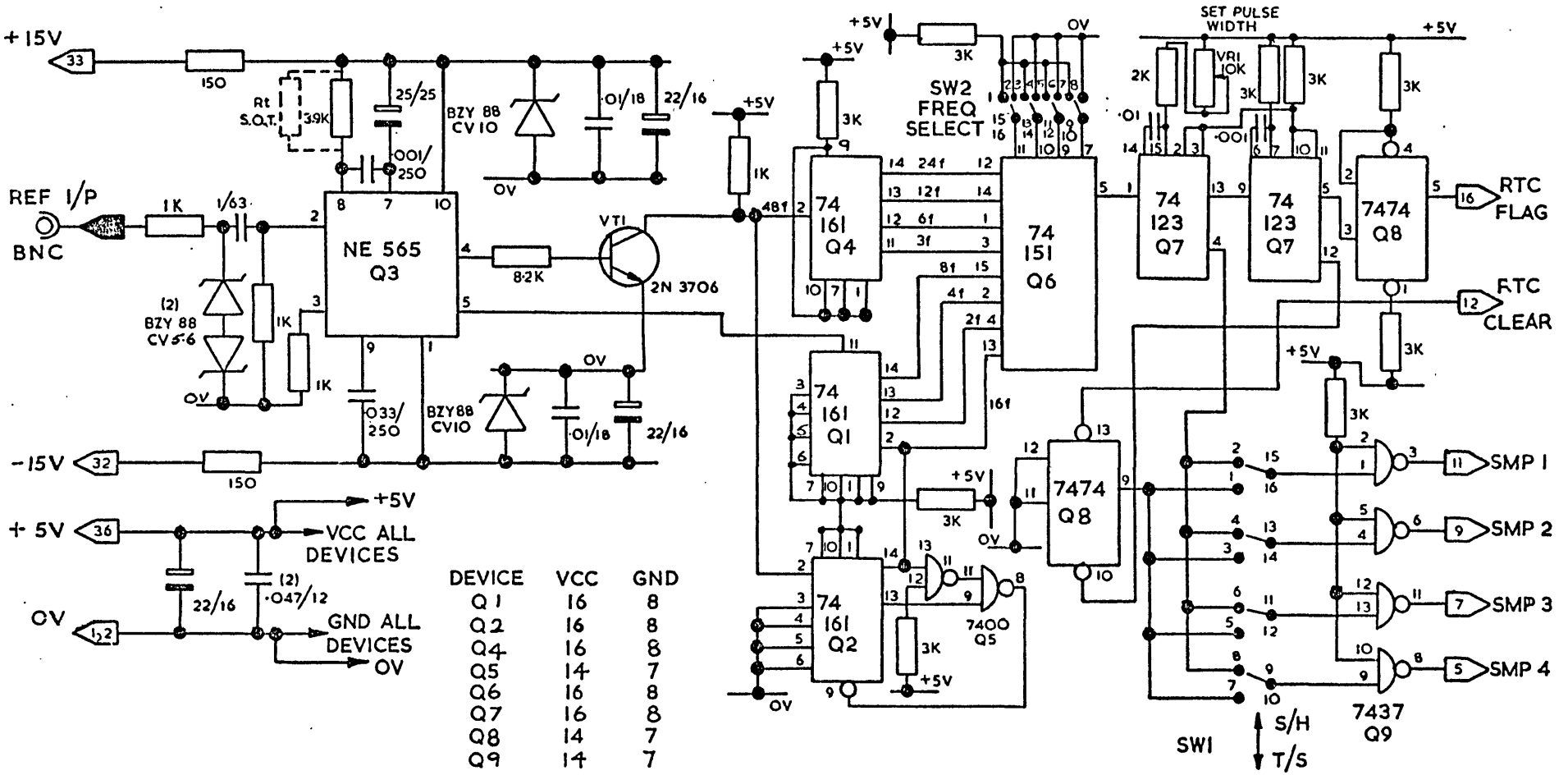


FIG A1,2 PHASE LOCKED SAMPLING CLOCK CIRCUIT

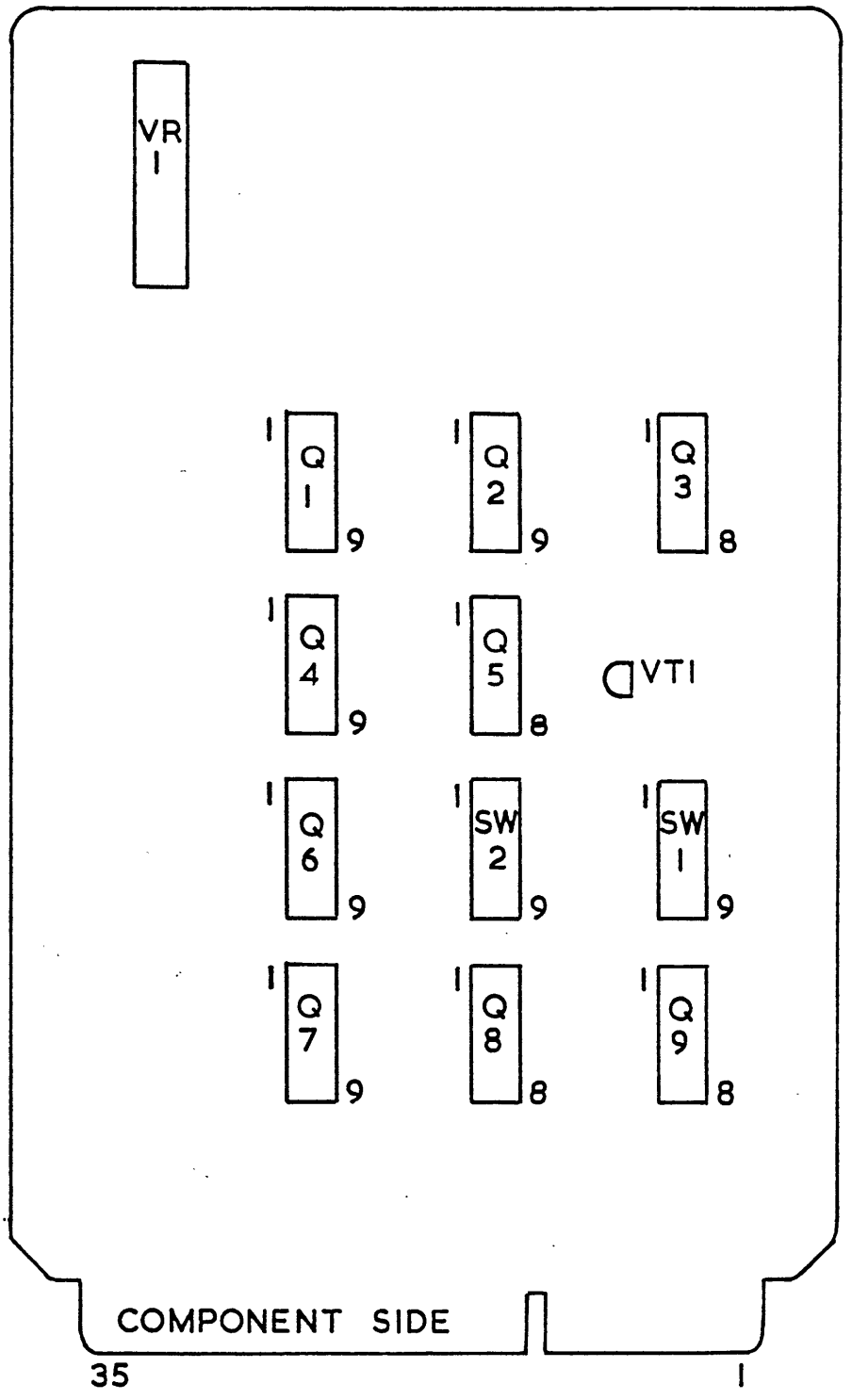
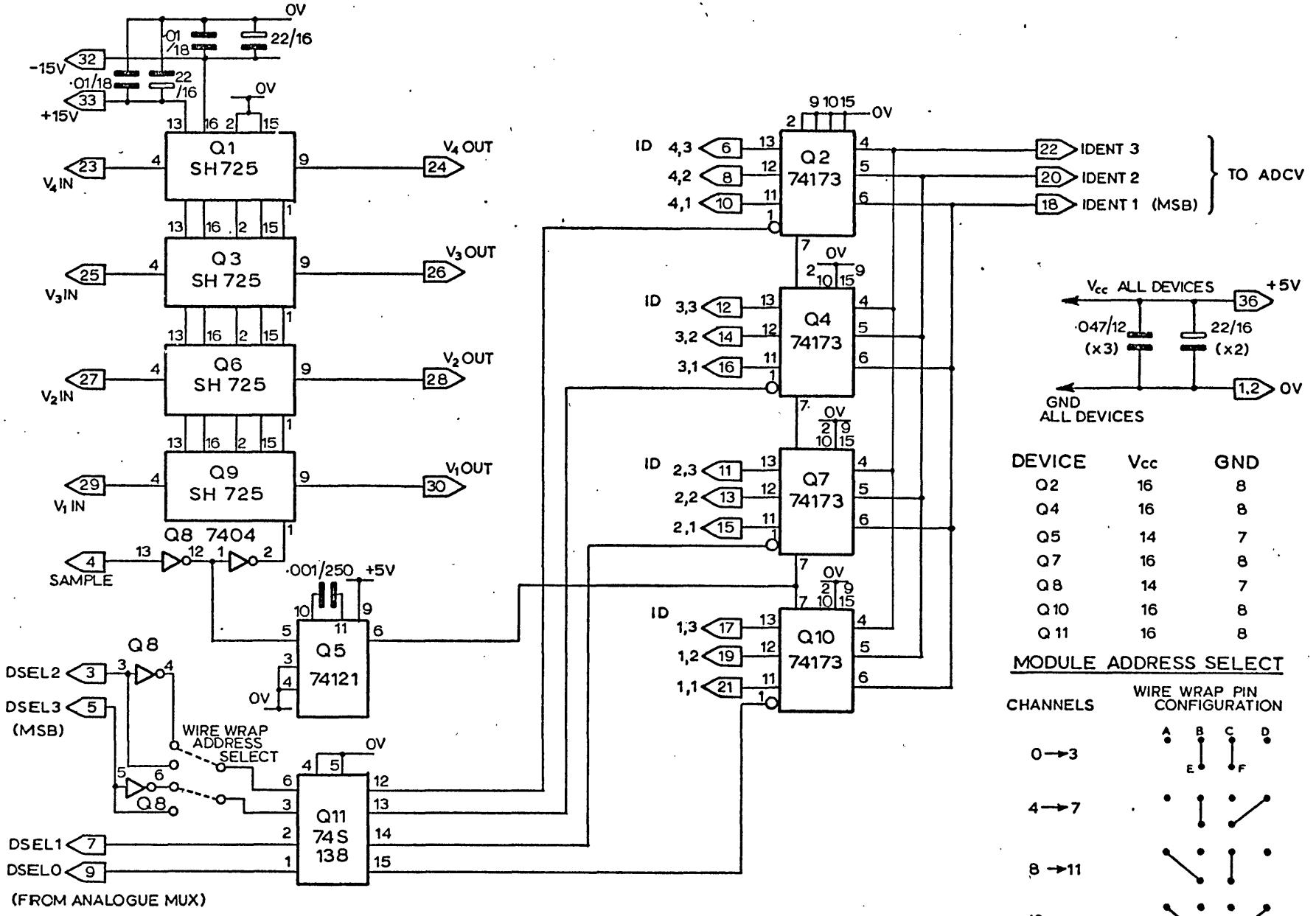


FIG A1.3 PHASE LOCKED SAMPLING CLOCK DEVICE IDENTIFICATION



DEVICE	Vcc	GND
Q2	16	8
Q4	16	8
Q5	14	7
Q7	16	8
Q8	14	7
Q10	16	8
Q11	16	8

MODULE ADDRESS SELECT

CHANNELS	WIRE WRAP PIN CONFIGURATION
0 → 3	
4 → 7	
8 → 11	
12 → 15	

Fig. A1·4 SAMPLE HOLD CIRCUIT

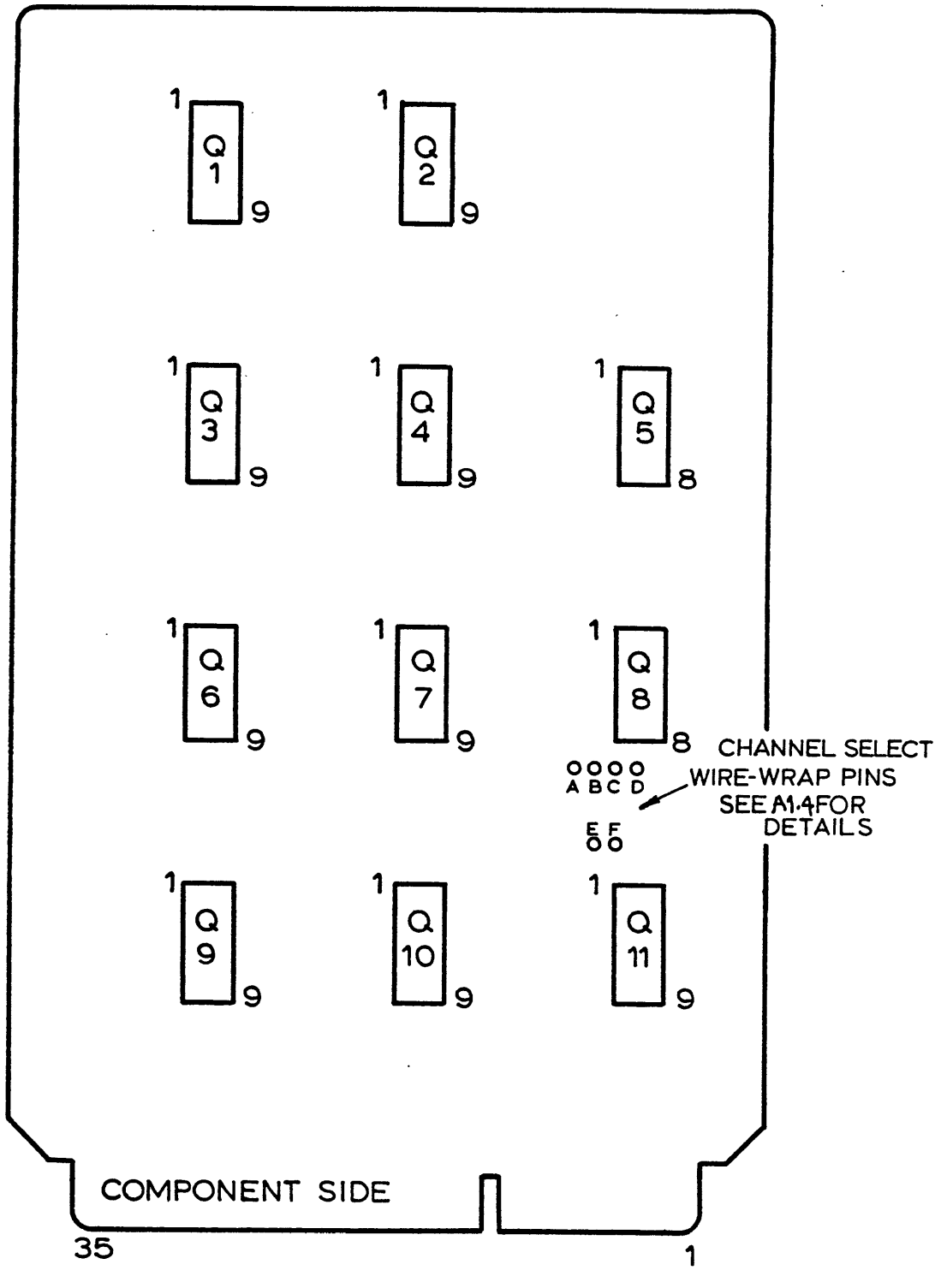
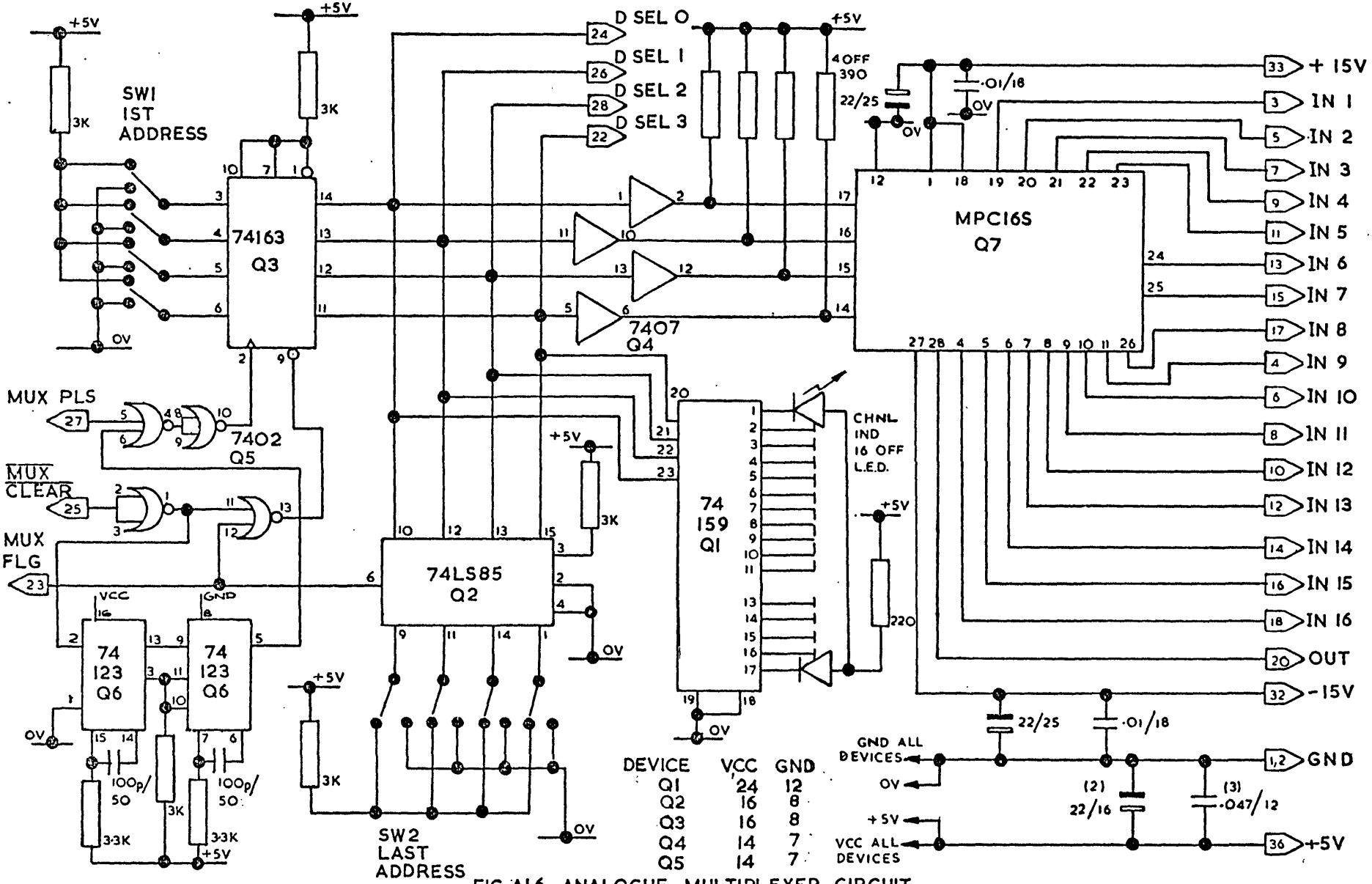


FIG. A1.5 SAMPLE-HOLD DEVICE IDENTIFICATION



DEVICE	VCC	GND
Q1	24	12
Q2	16	8
Q3	16	8
Q4	14	7
Q5	14	7

FIG A1,6 ANALOGUE MULTIPLEXER CIRCUIT

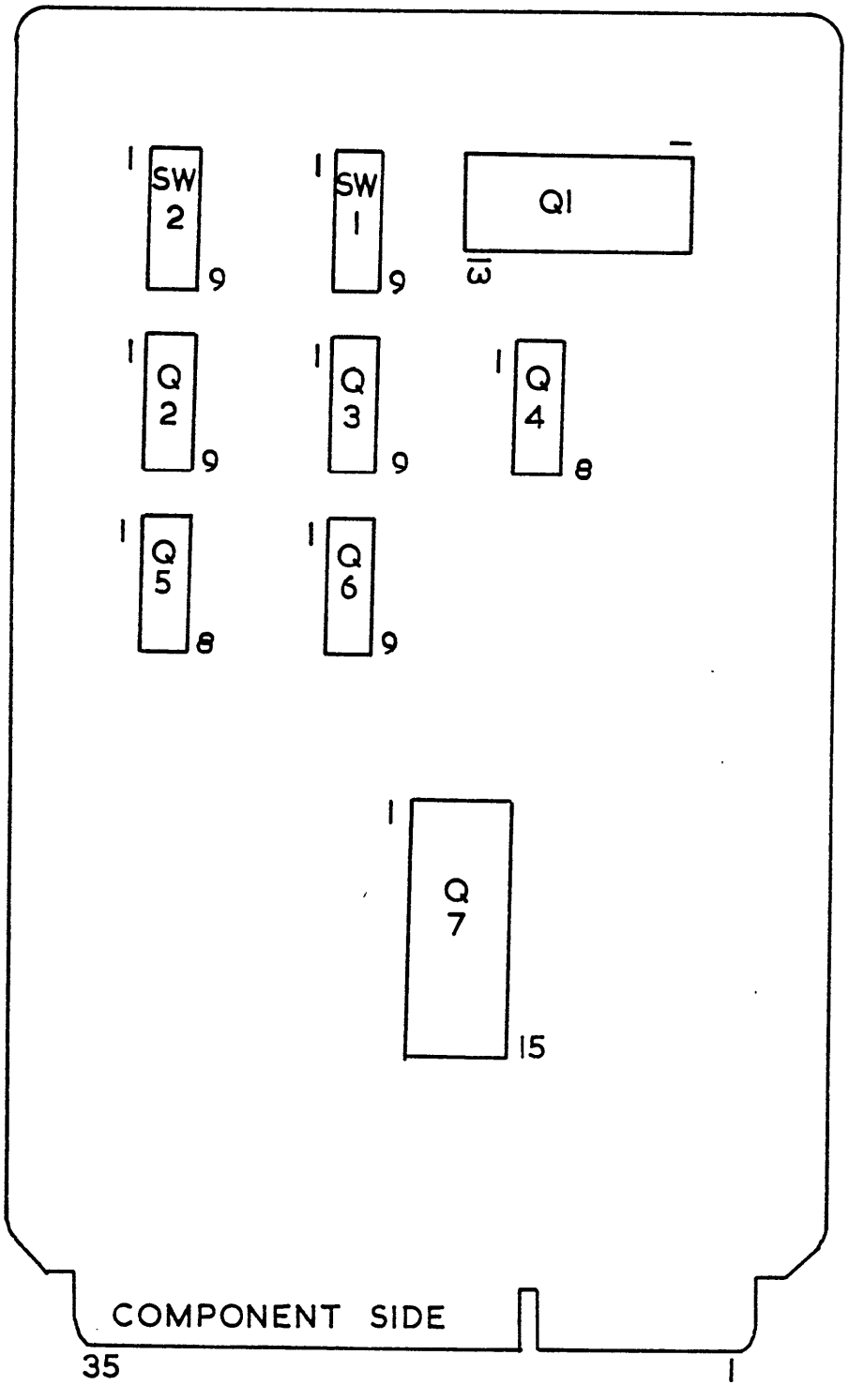
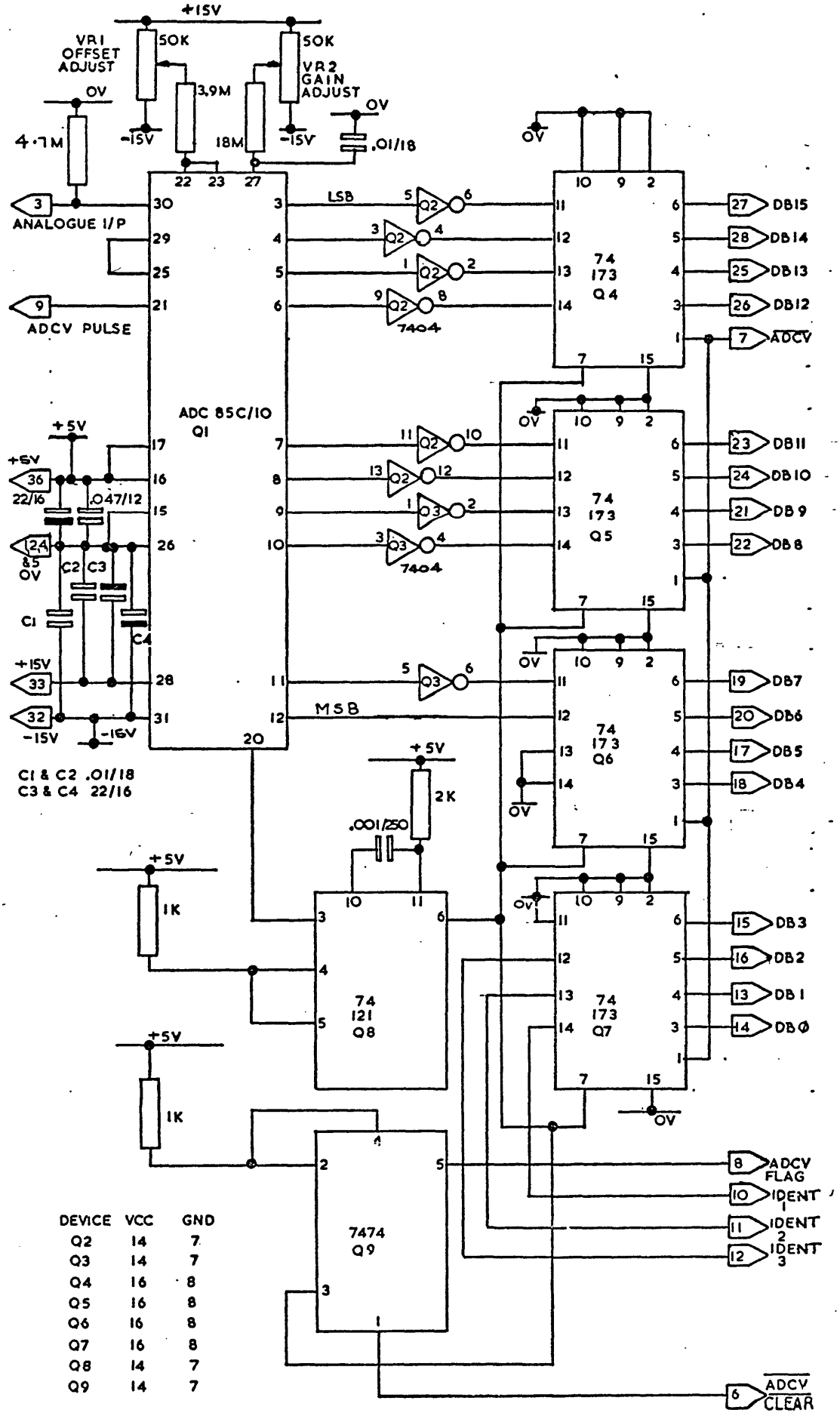
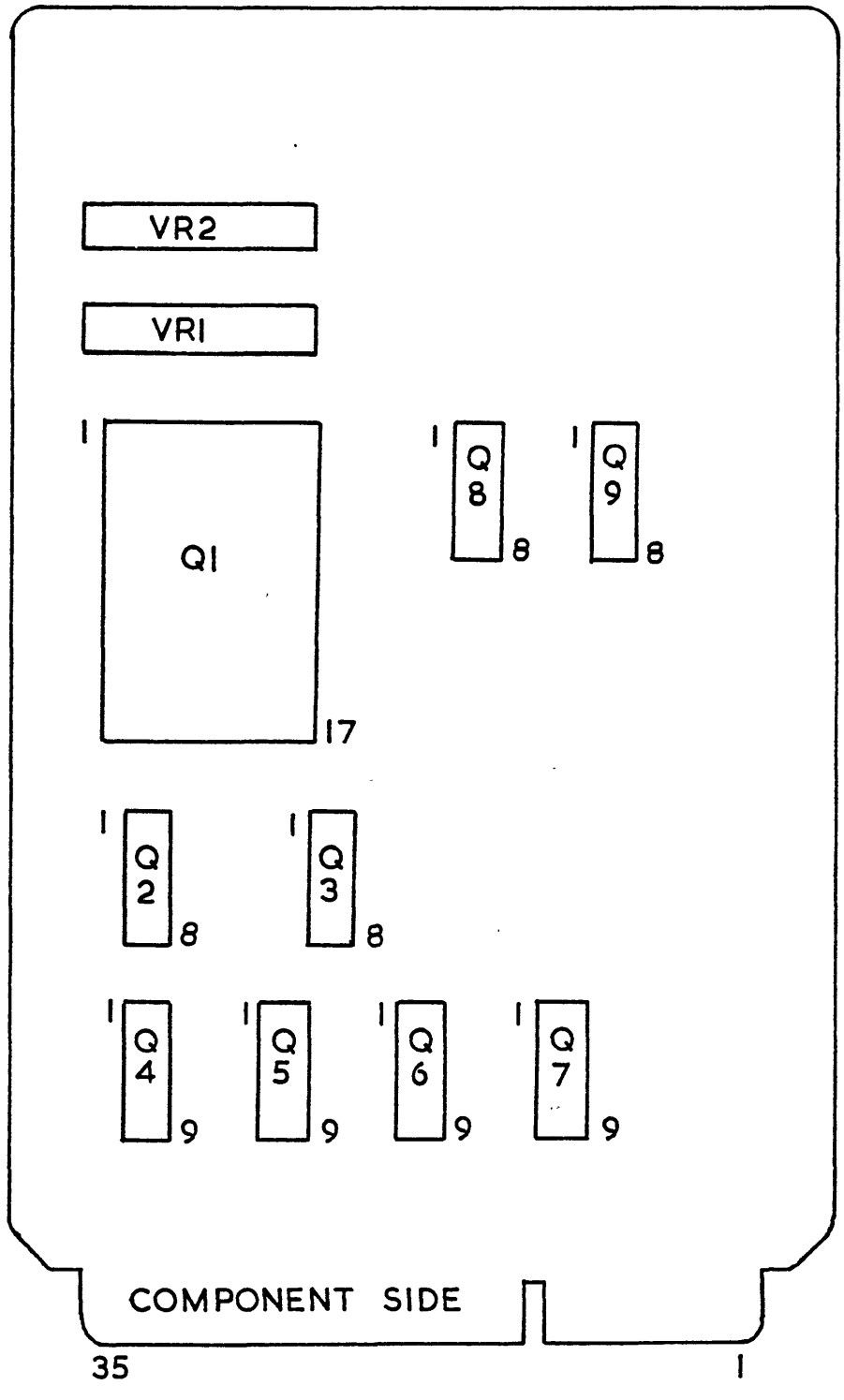


FIG A1.7 ANALOGUE MULTIPLEXER DEVICE IDENTIFICATION

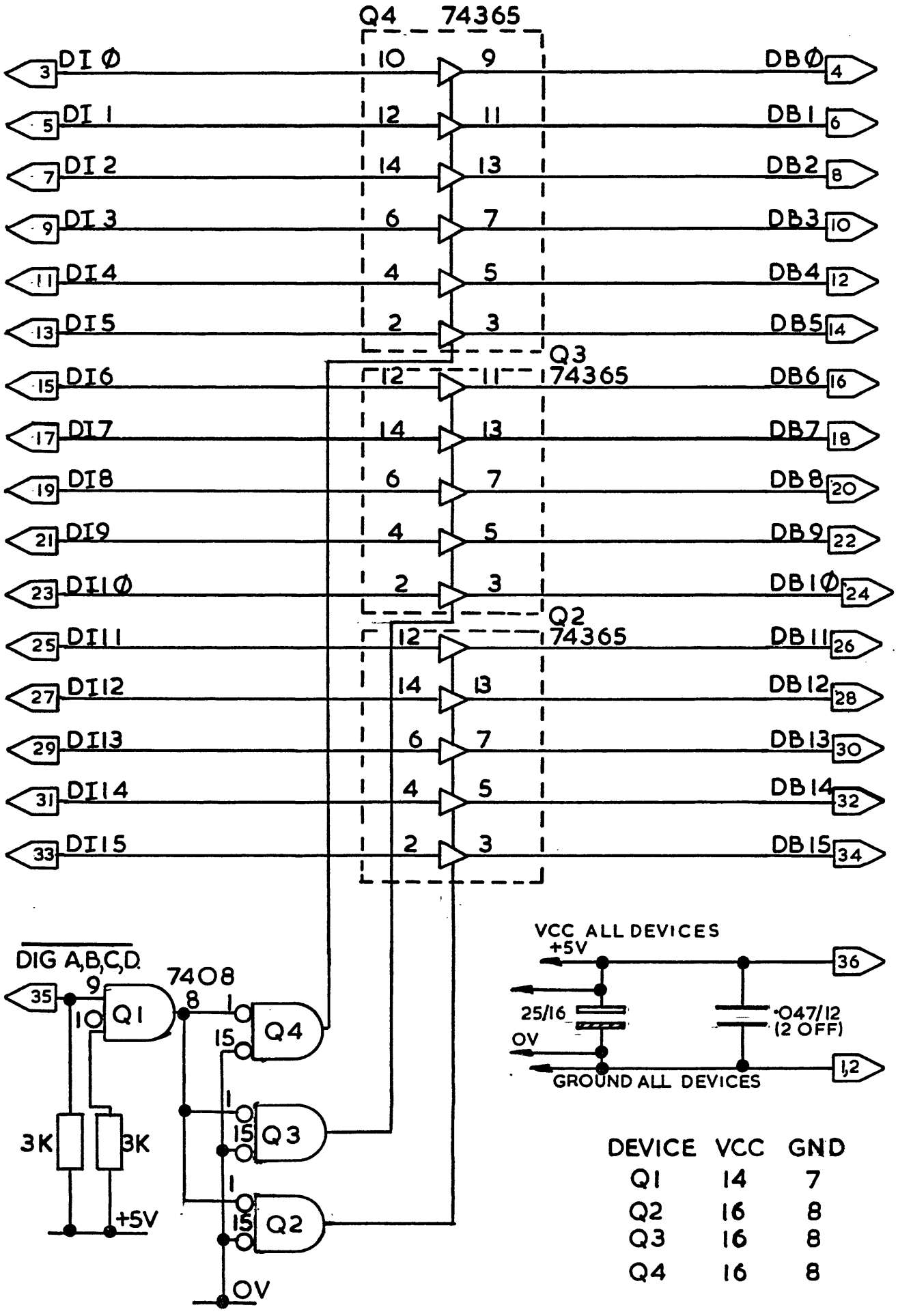


DEVICE	VCC	GND
Q2	14	7
Q3	14	7
Q4	16	8
Q5	16	8
Q6	16	8
Q7	16	8
Q8	14	7
Q9	14	7

FIG.A1.8 ANALOGUE - DIGITAL CONVERTER CIRCUIT.



FIGAI,9ANALOGUE DIGITAL CONVERTER
DEVICE IDENTIFICATION



DEVICE	VCC	GND
Q1	14	7
Q2	16	8
Q3	16	8
Q4	16	8

FIG.A1.10 DIGITAL INPUT BUS DRIVERS CIRCUIT

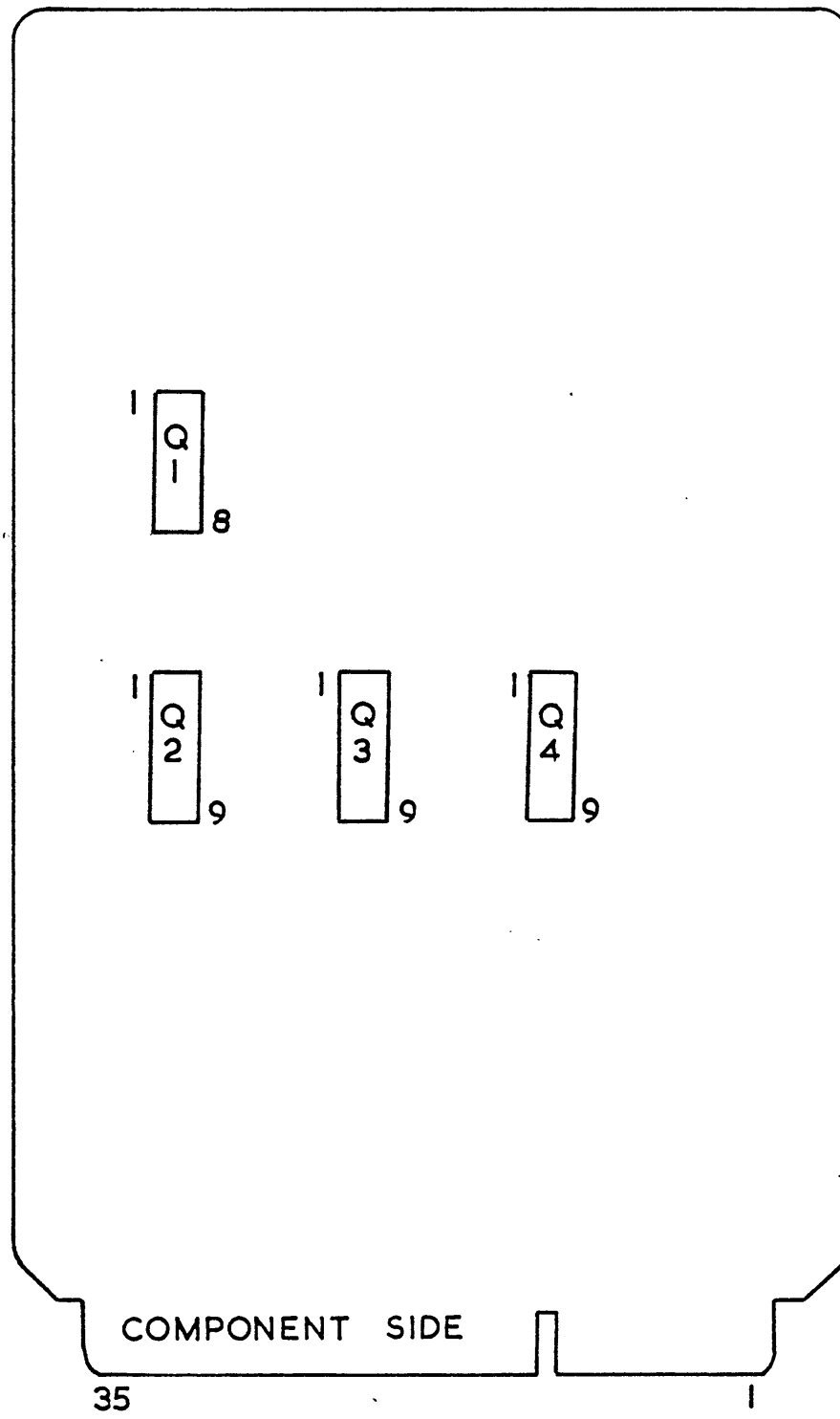


FIG A1.11 DIGITAL INPUT BUS DRIVERS
DEVICE IDENTIFICATION

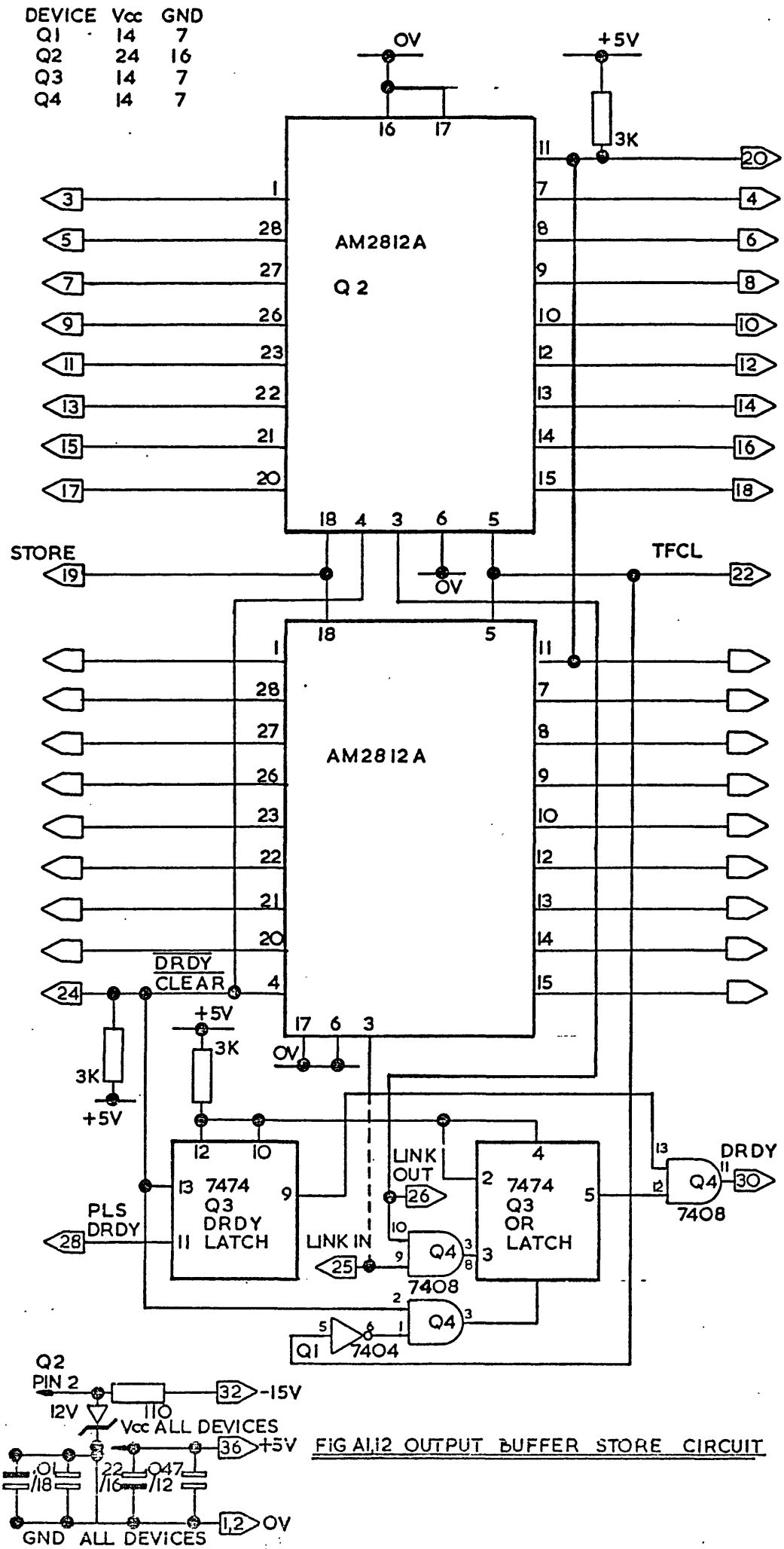


FIG A1,12 OUTPUT BUFFER STORE CIRCUIT

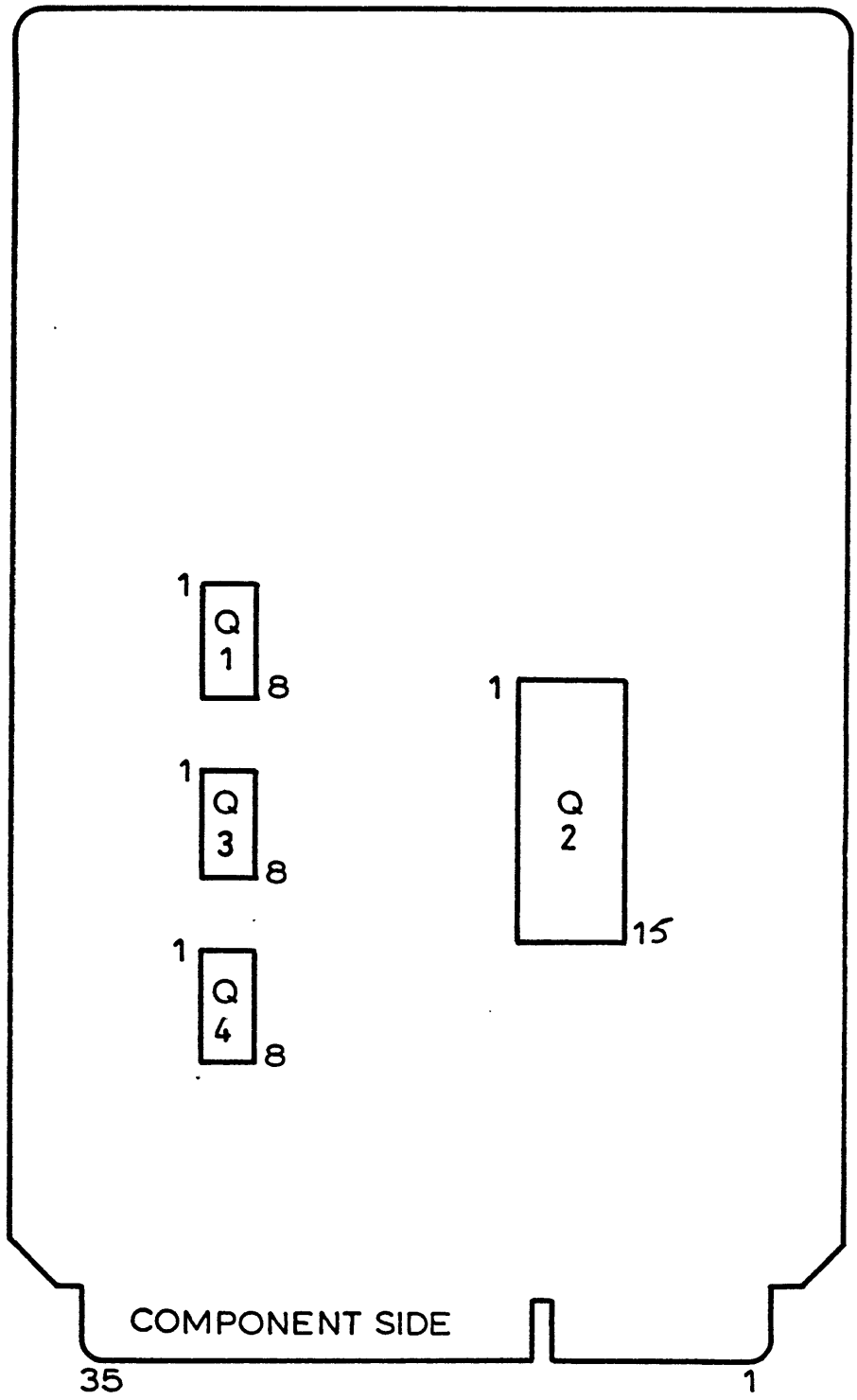
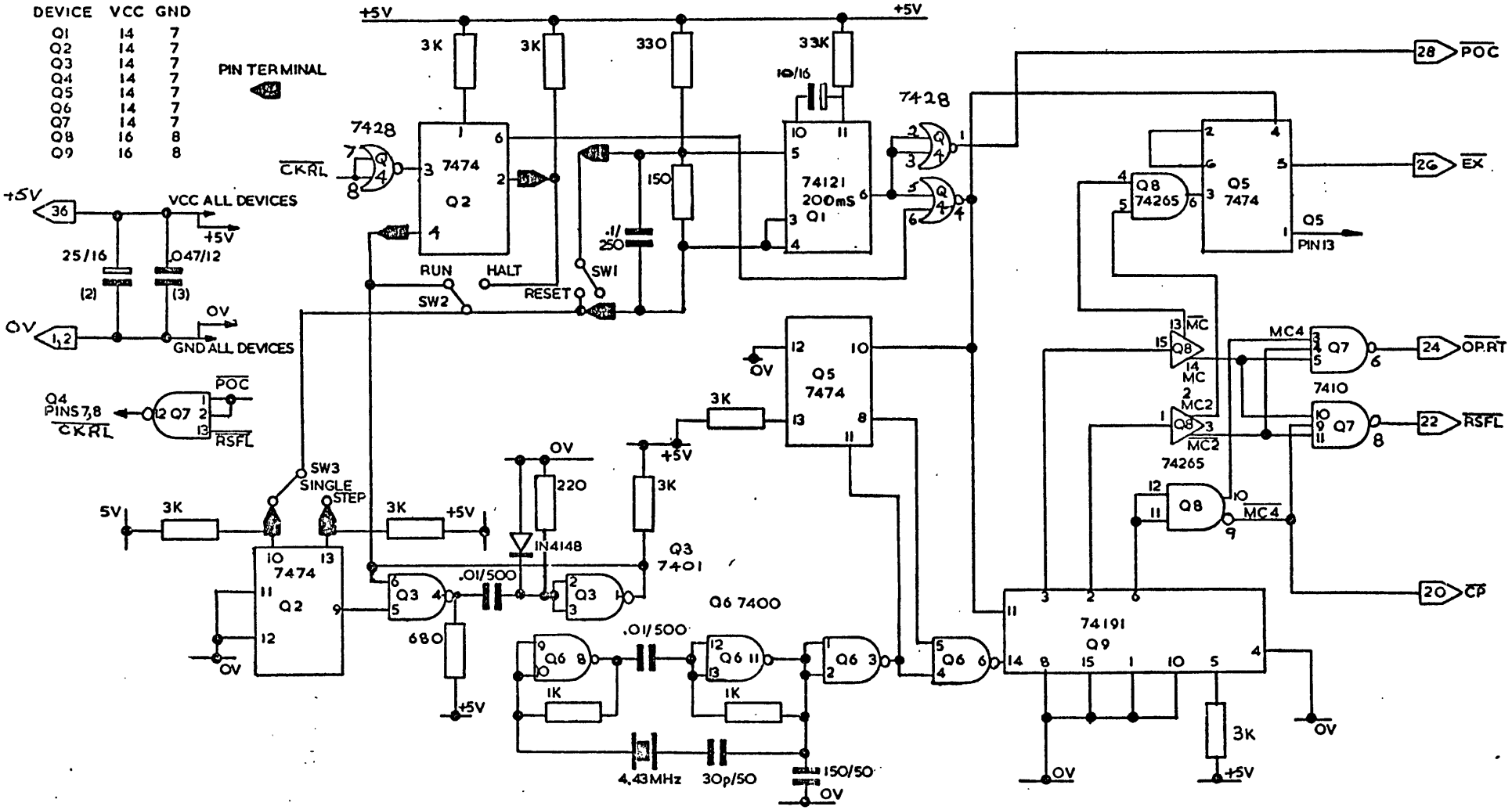


FIG. A1,13 OUTPUT BUFFER DEVICE IDENTIFICATION



FIGAI.14 ACQUISITION SYSTEM CLOCK GENERATOR CIRCUIT

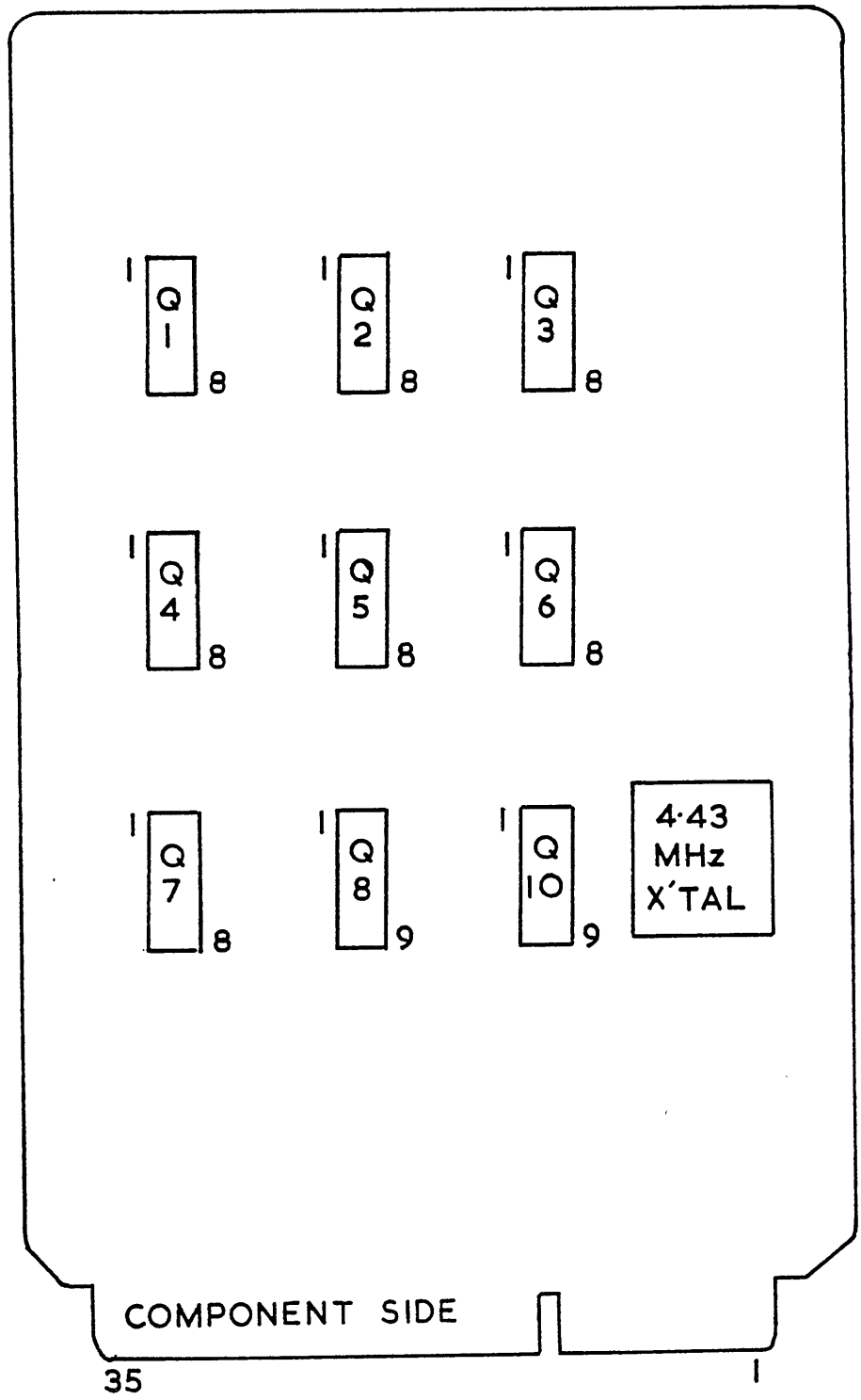
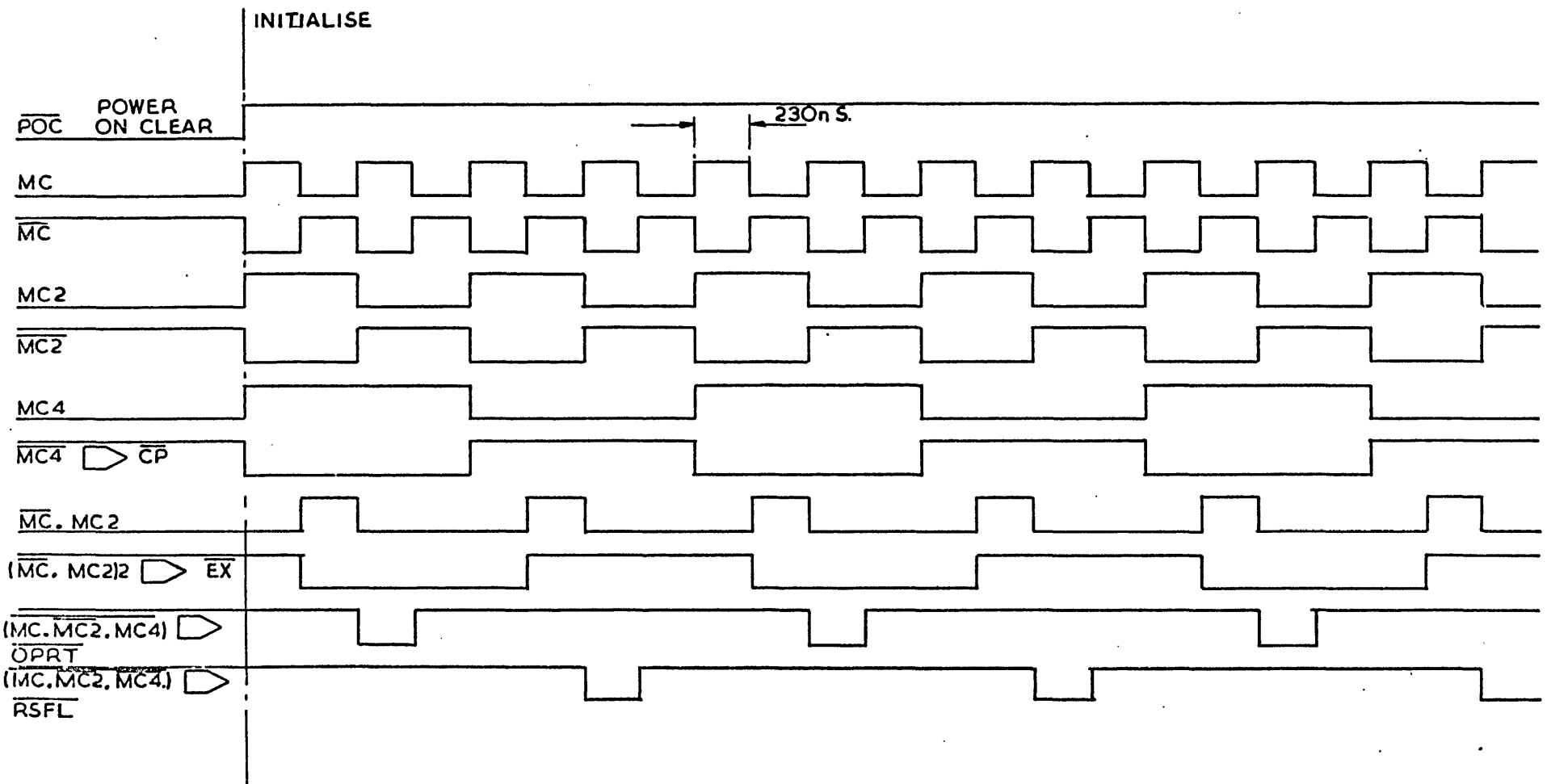
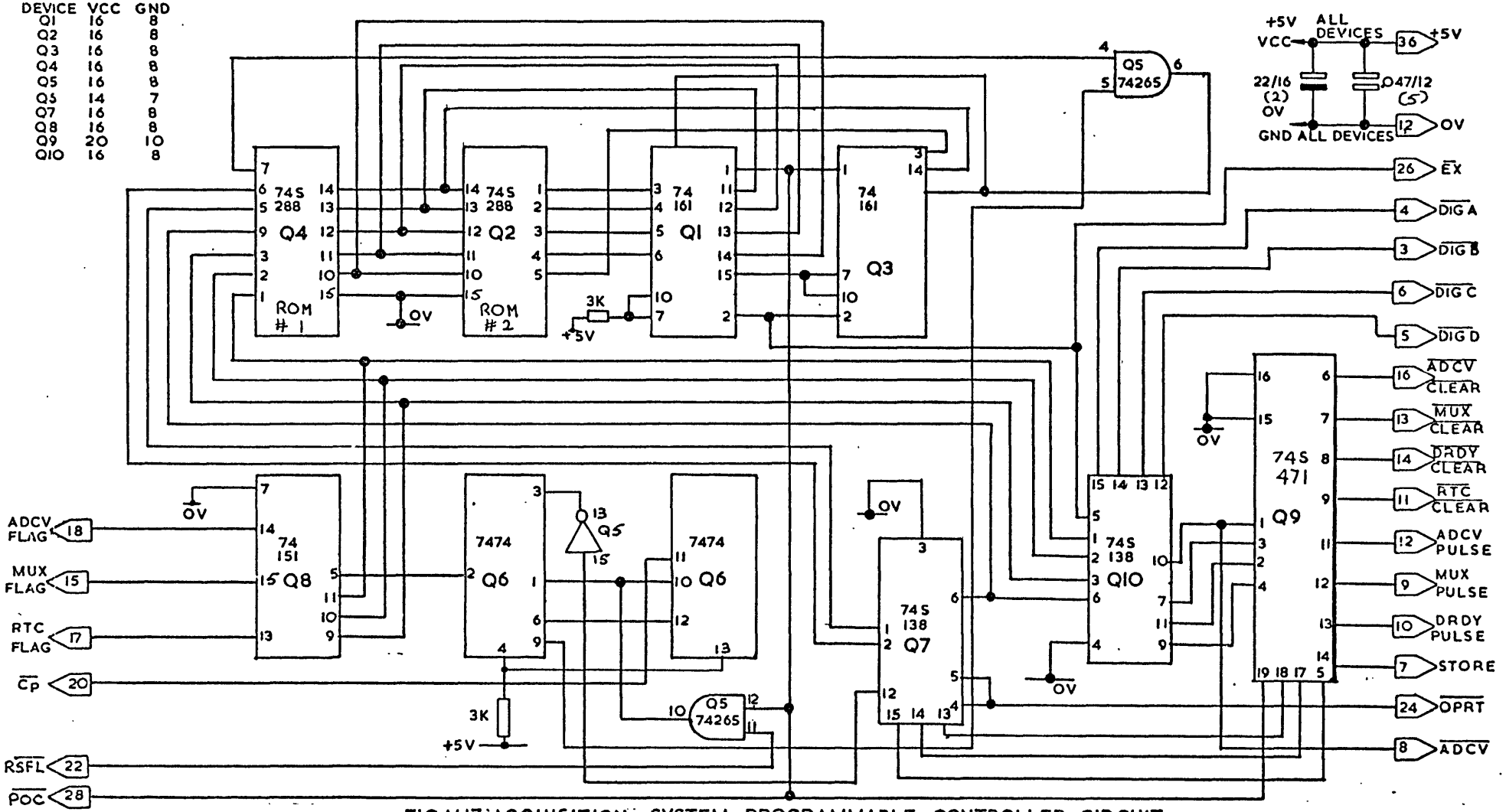


FIG AI,15 ACQUISITION SYSTEM CLOCK GENERATOR DEVICE IDENTIFICATION



FIGAI.16 ACQUISITION SYSTEM TIMING

DEVICE	VCC	GND
Q1	16	8
Q2	16	8
Q3	16	8
Q4	16	8
Q5	16	8
Q6	14	7
Q7	16	8
Q8	16	8
Q9	20	10
Q10	16	8



FIGAI.17 ACQUISITION SYSTEM PROGRAMMABLE CONTROLLER CIRCUIT

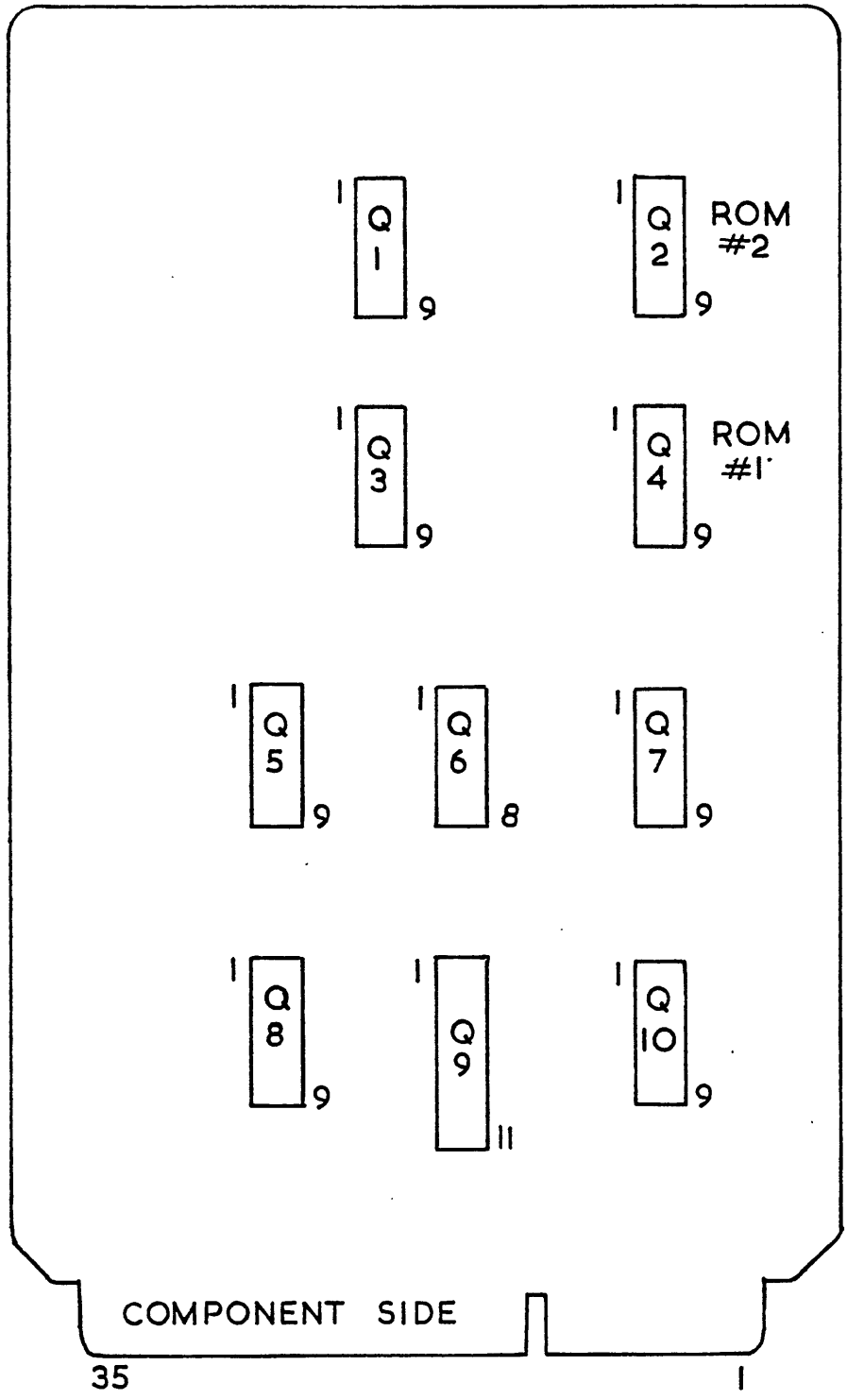
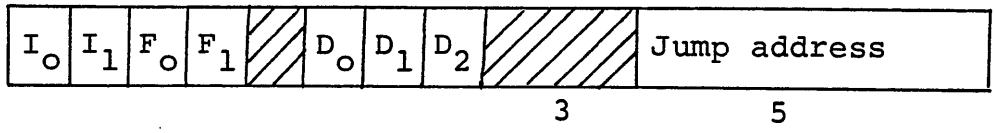


FIG A1.18 ACQUISITION SYSTEM PROGRAMMEABLE CONTROLLER DEVICE IDENTIFICATION

ADDRESS (INPUTS)								DATA (CONTROL OUTPUTS)								FUNCTION
H	G	F	E	D	C	B	A	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Power on clear Gives ADCV CLR
						REPEATED TO										
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	$\overline{\text{MUX CLR}}$ $\overline{\text{RTC CLR}}$ $\overline{\text{DRDY CLR}}$
1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	Invalid inputs. Outputs inactive.
						REPEATED TO										
1	0	1	1	1	1	0	1	0	0	0	0	1	1	1	1	
1	0	1	1	1	1	1	0	1	0	0	0	1	1	1	1	STORE with ADCV selected
1	0	1	1	1	1	1	1	1	0	0	0	1	1	1	1	STORE digital inputs.
1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	Invalid inputs Outputs inactive.
						REPEATED TO										
1	1	0	1	1	0	1	0	0	0	0	0	1	1	1	1	
1	1	0	1	1	0	1	1	0	1	0	0	1	1	1	1	DRDY PULSE
1	1	0	1	1	1	0	0	0	0	0	0	1	1	1	1	Invalid input
1	1	0	1	1	1	0	1	0	0	1	0	1	1	1	1	MUX PULSE
1	1	0	1	1	1	1	1	0	0	0	1	1	1	1	1	ADCV PULSE
1	1	0	1	1	1	1	1	0	0	0	0	1	1	1	1	Invalid inputs Outputs inactive.
						REPEATED TO										
1	1	1	0	0	1	1	0	0	0	0	0	1	1	1	1	
1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	1	$\overline{\text{RTC CLR}}$
1	1	1	0	1	0	0	0	0	0	0	0	1	1	1	1	Invalid inputs Outputs inactive
1	1	1	0	1	0	0	1	0	0	0	0	1	1	1	1	
1	1	1	0	1	0	1	0	0	0	0	0	1	1	1	1	
1	1	1	0	1	0	1	1	0	0	0	0	1	0	1	1	$\overline{\text{DRDY CLR}}$
1	1	1	0	1	1	0	0	0	0	0	0	1	1	1	1	Invalid input
1	1	1	0	1	1	0	1	0	0	0	0	1	1	0	1	$\overline{\text{MUX CLR}}$
1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	$\overline{\text{ADCV CLR}}$
1	1	1	0	1	1	1	1	0	0	0	0	1	1	1	1	Invalid inputs. Outputs inactive.
						REPEATED TO										
1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	

TABLE A1.1 Programmable controller output decoding ROM (Q9) contents map.



Device address group;

Device	Mnemonic	D_o	D_1	D_2
Digital input port 1	DIG A	0	0	0
" " " 2	DIG B	0	0	1
" " " 3	DIG C	0	1	0
" " " 4	DIG D	0	1	1
Analogue Multiplexer	MUX	1	0	0
Analogue-Digital conv	ADCV	1	0	1
Phase locked clock	RTC	1	1	0
Data ready flag	DRDY	1	1	1

Function group:

Function	Mnemonic	I_o	I_1	F_o	F_1
Program jump	JMP	0	0	0	0
Clear device or flag	CLR	1	1	0	0
Clock device	PLS	1	1	0	1
Transfer data	STR	1	1	1	0
Test device status	FLG	1	1	1	1

TABLE A1.2 Interface controller instruction formats.

Interface programming examples.

The following programs demonstrate the application of the interface instruction set in two cases:

i) acquisition of digital and analogue data,

ii) analogue data only

The flow chart, Fig. A1.19 is applicable to both programs, and the corresponding controller program R.O.M. maps are illustrated in Fig.A1.20, A1.21.

Control program identification

The pairs of P.R.O.M's which contain the controller programs are identified by four character codes:

X	X	X	1(2)
Digital data	Analogue data	Number of samples per block	R.O.M number

i) Digital data

The first character of the code indicates the digital input channels used, if any, and is an alphabetic sequence from A to P

e.g. A - input A only

K - inputs ABC

P - inputs ABCD

The letter X indicates that the digital inputs are unused.

ii) Analogue data

The second character is the letter A if analogue channels are included in the data block, otherwise the letter X is used. The channel numbers in use will depend on the multiplexer switch settings and thus can not be indicated in the program code.

iii) Number of samples per block

This figure is an indication of the number of samples stored in the output buffer prior to the setting of the data ready flag. In most applications it is unlikely to exceed 1.

iv) R.O.M. number

The final digit i.e. -1 or -2 corresponds to the numbered program memory sockets on the controller module. ROM - 1 contains the function and device address groups of the instructions, whilst ROM -2 holds jump address information.

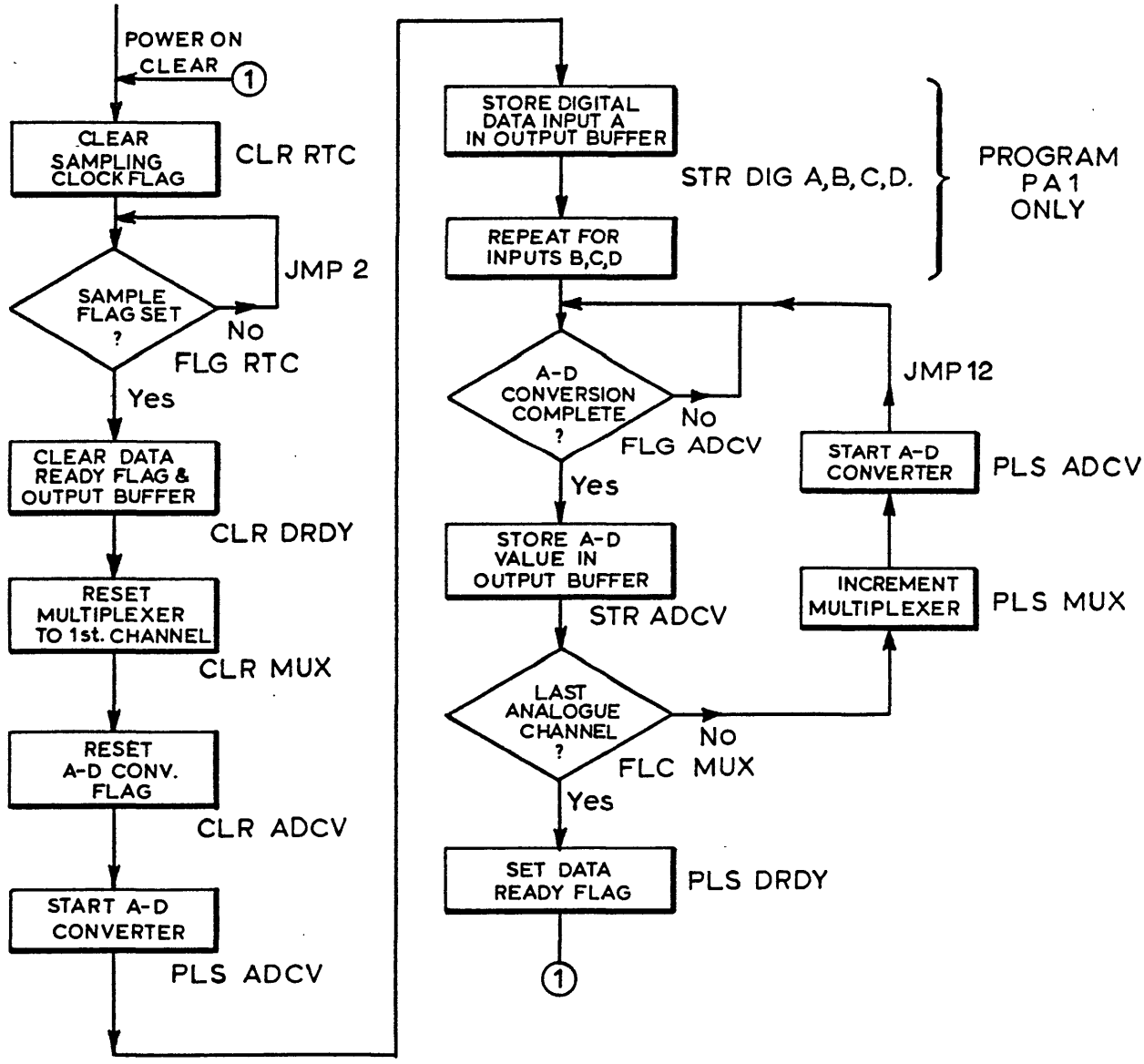


Fig. A1.19 Program flow chart for data acquisition interface controller.

ADDRESS						DATA															INSTRUCTION
DECIMAL	BINARY					ROM#1								ROM #2							
	A4	A3	A2	A1	A0	O8	O7	O6	O5	O4	O3	O2	O1	O8	O7	O6	O5	O4	O3	O2	O1	
0	0	0	0	0	0	0	0	0	0	X	0	0	0	X	X	X	0	0	0	0	1	JMP 1
1	0	0	0	0	1	1	1	0	0	X	1	1	0	X	X	X	0	0	0	0	0	CLR RTC
2	0	0	0	1	0	1	1	1	1	X	1	1	0	X	X	X	0	0	1	0	0	FLG RTC
3	0	0	0	1	1	0	0	0	0	X	0	0	0	X	X	X	0	0	0	1	0	JMP 2
4	0	0	1	0	0	1	1	0	0	X	1	1	1	X	X	X	0	0	0	0	0	CLR DRDY
5	0	0	1	0	1	1	1	0	0	X	1	0	0	X	X	X	0	0	0	0	0	CLR MUX
6	0	0	1	1	0	1	1	0	0	X	1	0	1	X	X	X	0	0	0	0	0	CLR ADCV
7	0	0	1	1	1	1	1	0	1	X	1	0	1	X	X	X	0	0	0	0	0	PLS ADCV
8	0	1	0	0	0	1	1	1	0	X	0	0	0	X	X	X	0	0	0	0	0	STR DIG A
9	0	1	0	0	1	1	1	1	0	X	0	0	1	X	X	X	0	0	0	0	0	STR DIG B
10	0	1	0	1	0	1	1	1	0	X	0	1	0	X	X	X	0	0	0	0	0	STR DIG C
11	0	1	0	1	1	1	1	1	0	X	0	1	1	X	X	X	0	0	0	0	0	STR DIG D
12	0	1	1	0	0	1	1	1	1	X	1	0	1	X	X	X	0	1	1	1	0	FLG ADCV
13	0	1	1	0	1	0	0	0	0	X	0	0	0	X	X	X	0	1	1	0	0	JMP 12
14	0	1	1	1	0	1	1	1	0	X	1	0	1	X	X	X	0	0	0	0	0	STR ADCV
15	0	1	1	1	1	1	1	1	1	X	1	0	0	X	X	X	1	0	1	0	0	FLG MUX
16	1	0	0	0	0	1	1	0	1	X	1	0	0	X	X	X	0	0	0	0	0	PLS MUX
17	1	0	0	0	1	1	1	0	0	X	1	0	1	X	X	X	0	0	0	0	0	CLR ADCV
18	1	0	0	1	0	1	1	0	1	X	1	0	1	X	X	X	0	0	0	0	0	PLS ADCV
19	1	0	0	1	1	0	0	0	0	X	0	0	0	X	X	X	0	1	1	0	0	JMP 12
20	1	0	1	0	0	1	1	0	1	X	1	1	1	X	X	X	0	0	0	0	0	PLS DRDY
21	1	0	1	0	1	0	0	0	0	X	0	0	0	X	X	X	0	0	0	0	1	JMP 1

FIG. A1.20 ACQUISITION SYSTEM PROGRAM PAI. ROM. MAP

ADDRESS						DATA															INSTRUCTION
DECIMAL	BINARY					ROM #1								ROM #2							
	A4	A3	A2	A1	A0	O8	O7	O6	O5	O4	O3	O2	O1	O8	O7	O6	O5	O4	O3	O2	O1	
0	0	0	0	0	0	0	0	0	0	X	0	0	0	X	X	X	0	0	0	0	1	JMP 1
1	0	0	0	0	1	1	1	0	0	X	1	1	0	X	X	X	0	0	0	0	0	CLR RTC
2	0	0	0	1	0	1	1	1	1	X	1	1	0	X	X	X	0	0	1	0	0	FLG RTC
3	0	0	0	1	1	0	0	0	0	X	0	0	0	X	X	X	0	0	0	1	0	JMP 2
4	0	0	1	0	0	1	1	0	0	X	1	1	1	X	X	X	0	0	0	0	0	CLR DRDY
5	0	0	1	0	1	1	1	0	0	X	1	0	0	X	X	X	0	0	0	0	0	CLR MUX
6	0	0	1	1	0	1	1	0	0	X	1	0	1	X	X	X	0	0	0	0	0	CLR ADCV
7	0	0	1	1	1	1	1	0	1	X	1	0	1	X	X	X	0	0	0	0	0	PLS ADCV
8	0	1	0	0	0	1	1	1	1	X	1	0	1	X	X	X	0	1	0	1	0	FLG ADCV
9	0	1	0	0	1	0	0	0	0	X	0	0	0	X	X	X	0	1	0	0	0	JMP 8
10	0	1	0	1	0	1	1	1	0	X	1	0	1	X	X	X	0	0	0	0	0	STR ADCV
11	0	1	0	1	1	1	1	1	1	X	1	0	0	X	X	X	0	1	1	1	0	FLG MUX
12	0	1	1	0	0	1	1	0	1	X	1	0	0	X	X	X	0	0	0	0	0	PLS MUX
13	0	1	1	0	1	0	0	0	0	X	0	0	0	X	X	X	0	0	1	1	0	JMP 6
14	0	1	1	1	0	1	1	0	1	X	1	1	1	X	X	X	0	0	0	0	0	PLS DRDY
15	0	1	1	1	1	0	0	0	0	X	0	0	0	X	X	X	0	0	0	0	1	JMP 1

FIG A1.21 ACQUISITION SYSTEM PROGRAM XA1
ROM MAP

INPUT SOCKETS (McMURDO RED-RANGE 16 WAY).

PIN NO	DIGITAL INPUTS	ANALOGUE INPUTS
1	DATA IN 15 MSB	ANALOGUE IN 1
2	" 14	IDENT DATA 2 (MSB)
3	" 13	" 1
4	12	" 0 (LSB)
5	11	ANALOGUE IN 2
6	10	IDENT DATA 2
7	9	" 1
8	8	" 0
9	7	ANALOGUE IN 3
10	6	IDENT DATA 2
11	5	" 1
12	4	" 0
13	3	ANALOGUE IN 4
14	2	IDENT DATA 2
15	1	" 1
16	0 (LSB)	" 0

TABLE A1.3 Acquisition interface input plug details.

OUTPUT SOCKET (McMURDO RED-RANGE 24 WAY).

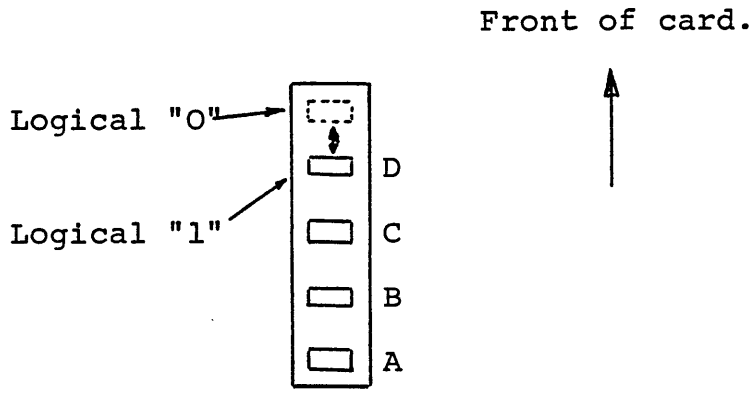
PIN NO	FUNCTION	PIN NO	FUNCTION
1	DATA OUT 0(LSB)	13	DATA OUT 12
2	" 1	14	" 13
3	" 2	15	" 14
4	" 3	16	" 15(MSB)
5	" 4	17	NC
6	" 5	18	DATA READY FLAG
7	" 6	19	NC
8	" 7	20	DATA TRANSFER CLOCK
9	" 8	21	NC
10	" 9	22	OUTPUT ENABLE
11	" 10	23	NC
12	" 11	24	POWER-ON CLEAR (O/P)

NC = NOT CONNECTED.

POWER PLUG (McMURDO RED-RANGE 8 WAY).

PIN NO	FUNCTION
1 & 5	- 15V
2 & 6	+ 15V
3 & 7	+ 5V
4 & 8	OV (GND)

TABLE A1.4 Acquisition interface output socket and power plug details.



First (SW1)/Last (SW2) channel address switches.

First/Last channel.	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

TABLE A1.5 Analogue multiplexer channel control switches.

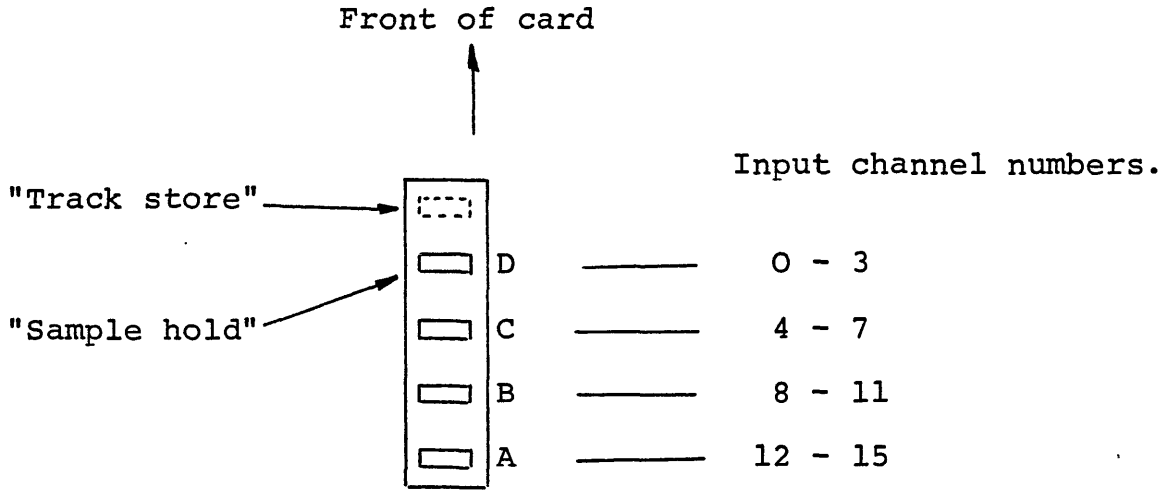
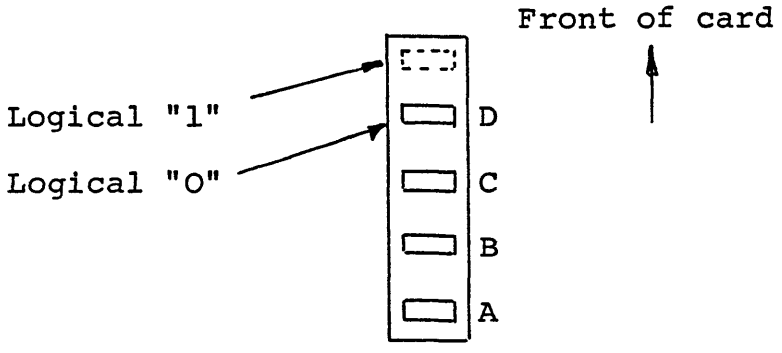


TABLE A1.6 Phase locked sampling clock, mode select switches (SW1).



Sampling rate (s.c)	D	C	B	A
Clock off	X	X	X	1
2	0	0	0	0
3	1	0	0	0
4	0	1	0	0
6	1	1	0	0
8	0	0	1	0
12	1	0	1	0
16	0	1	1	0
24	1	1	1	0

TABLE A1.7 Phase locked sampling clock, clock rate select switches (SW2).

APPENDIX 2

Protection processor.

Drawings and details.

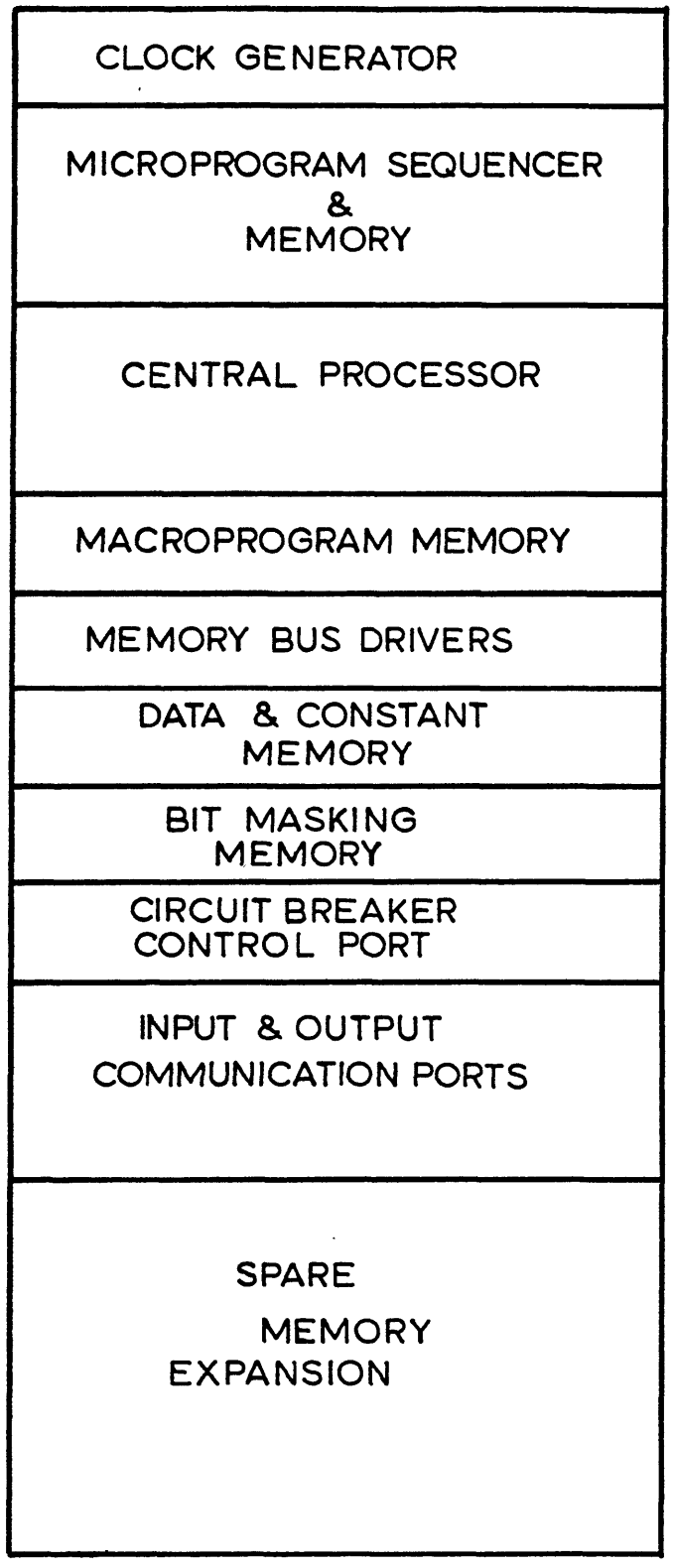


Fig. A2·1 PROTECTION PROCESSOR RACK MODULE POSITIONS.

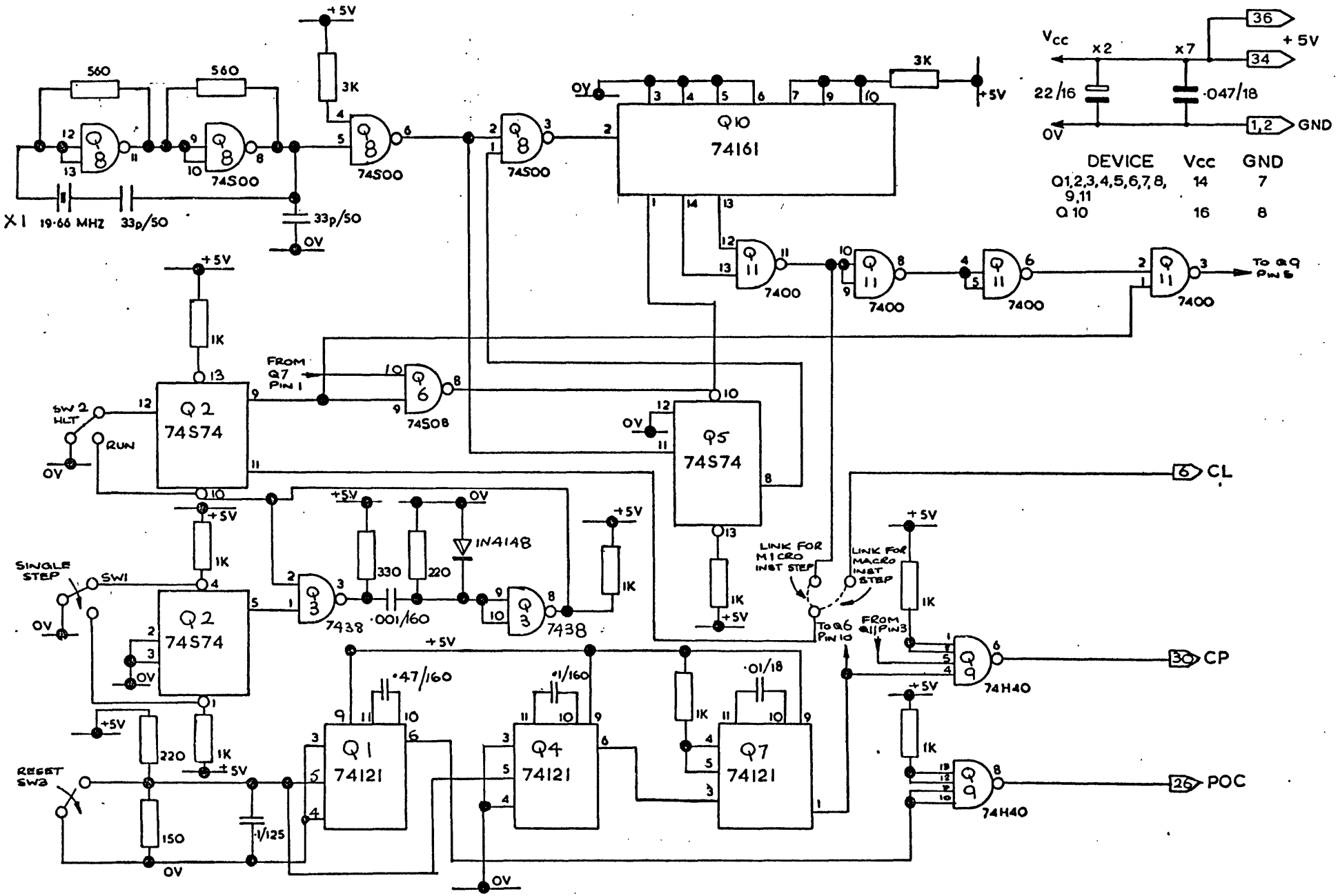


Fig. A2·2 PROTECTION PROCESSOR CLOCK GENERATOR CIRCUIT

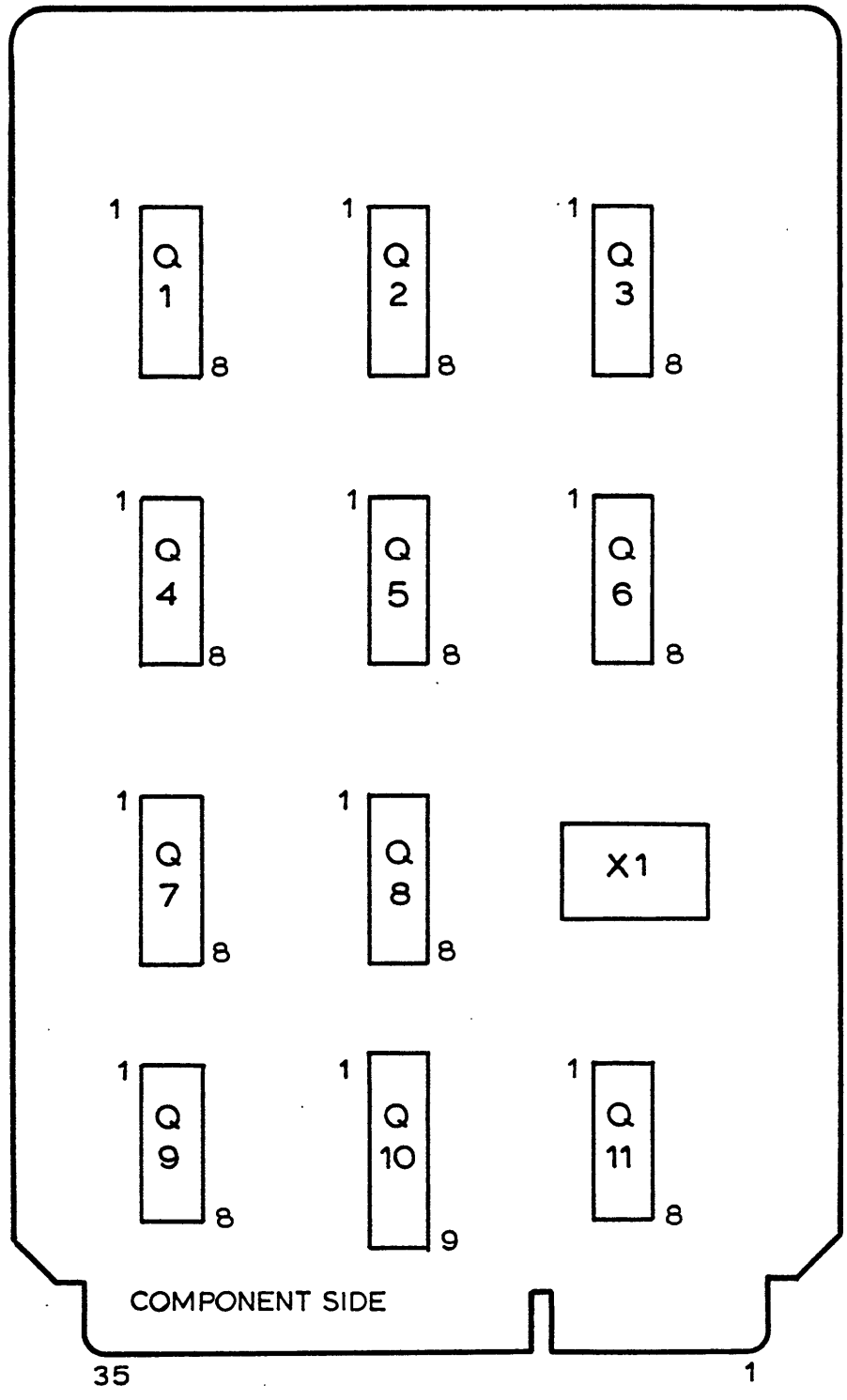


Fig. A2·3 PROTECTION PROCESSOR CLOCK GENERATOR DEVICE IDENTIFICATION.

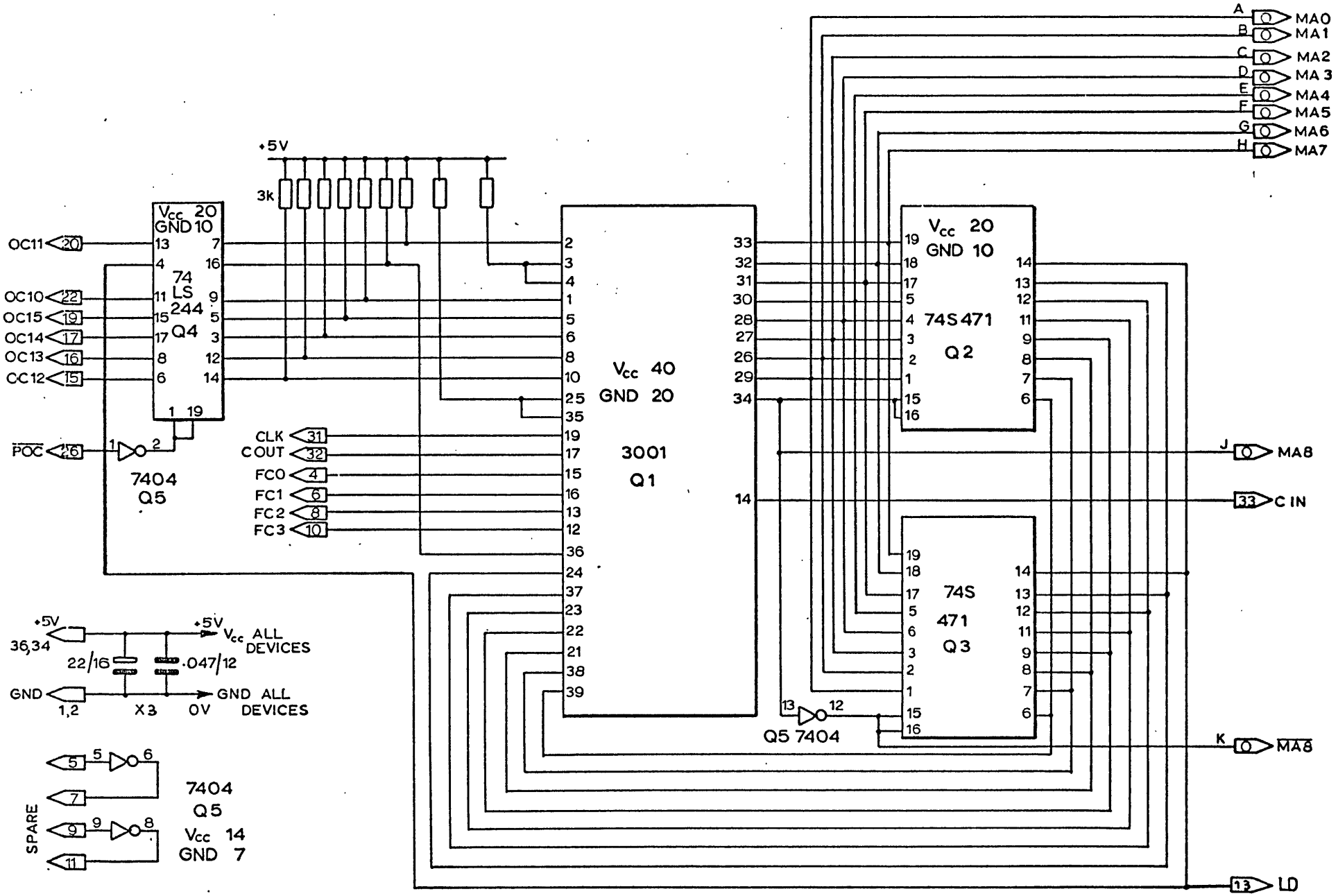


Fig. A2.4 MICROPROGRAM SEQUENCER CIRCUIT

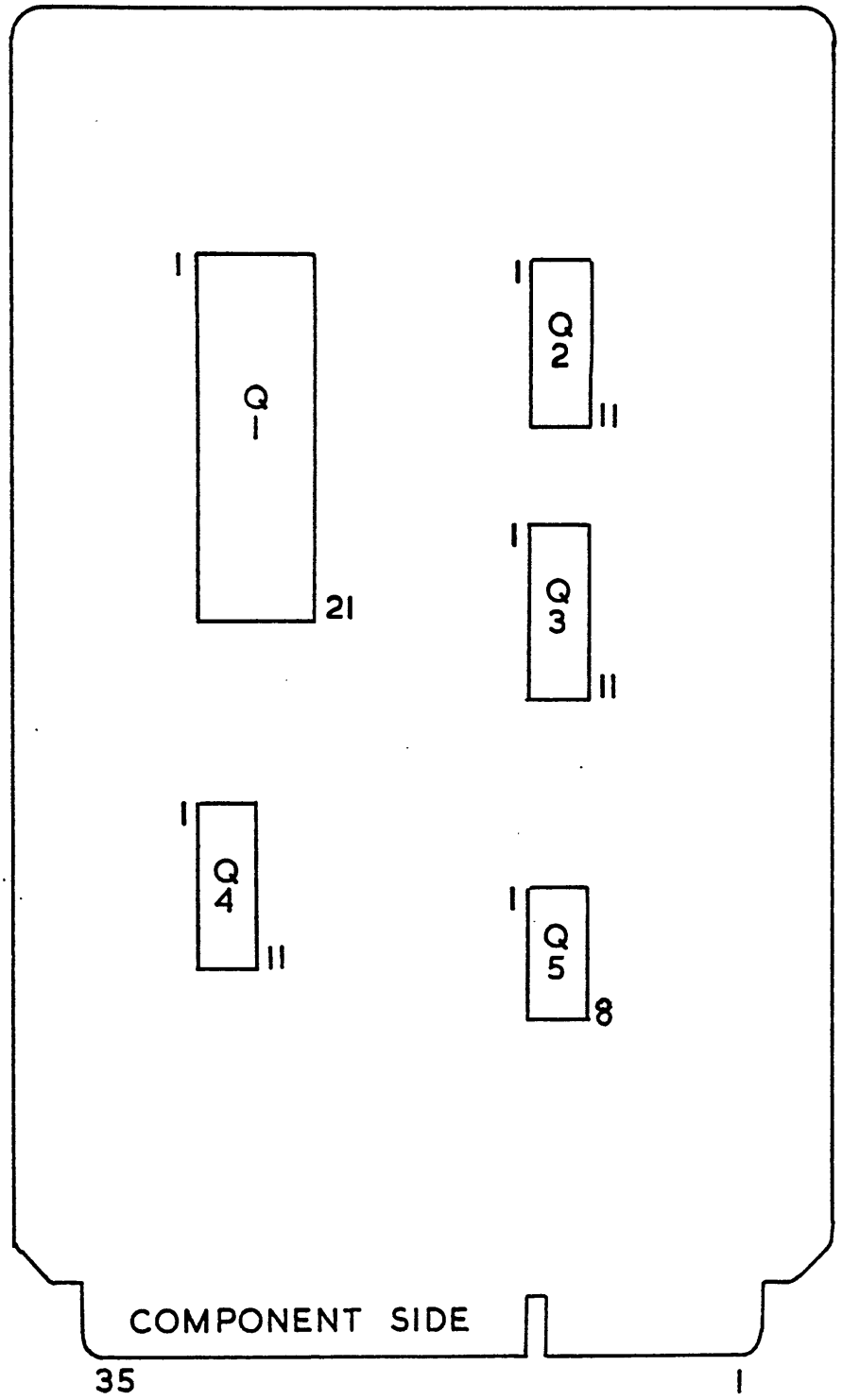
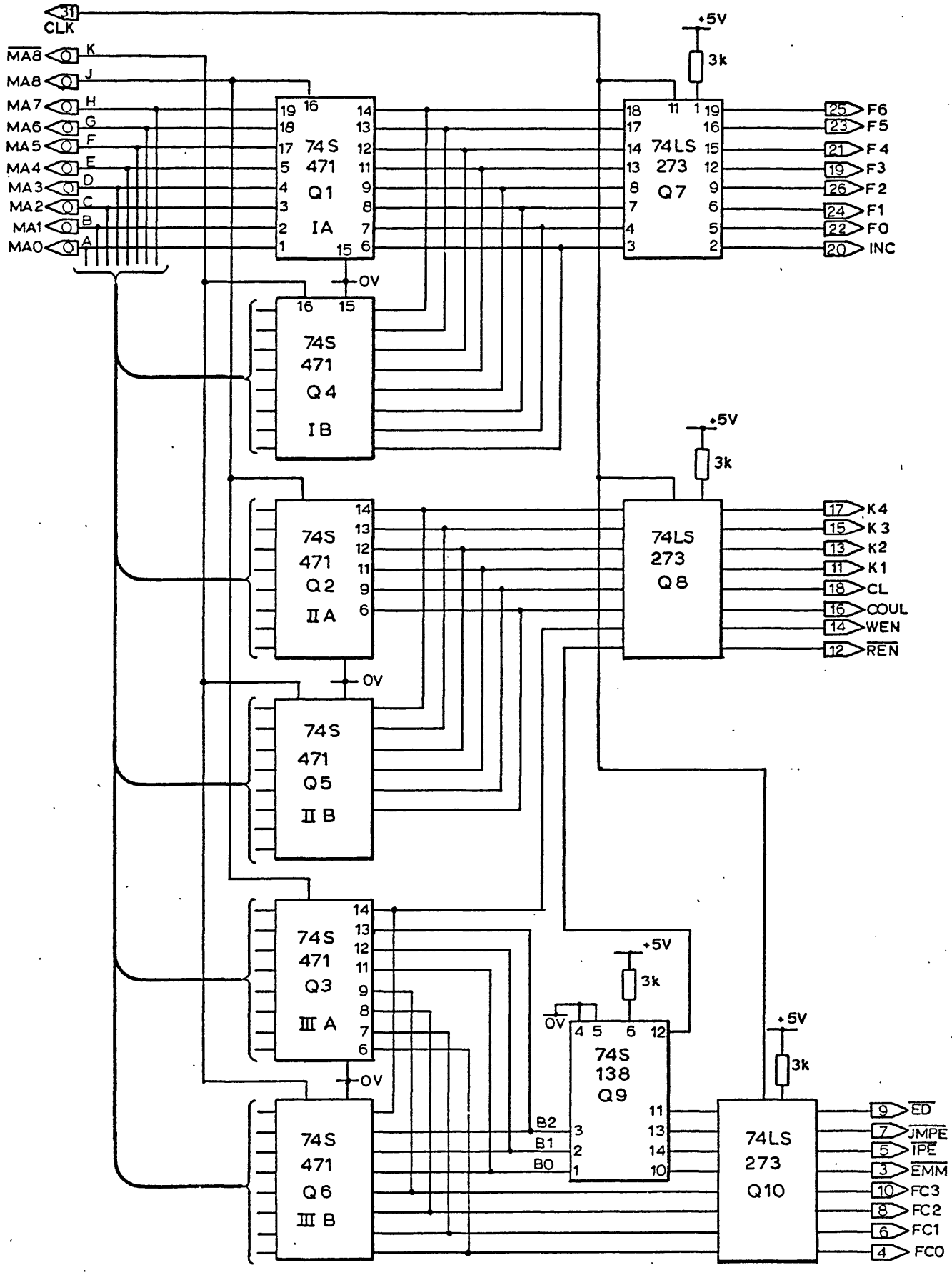
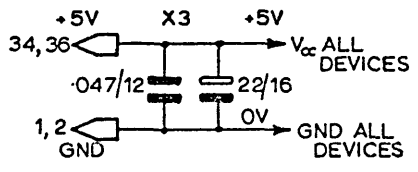


FIG. A2.5 MICROPROGRAM SEQUENCER DEVICE IDENTIFICATION





DEVICE	V_{cc}	GND
Q 1,2,3,4,5, 6,7,8,9,10	20	10
Q 9	16	8

Fig. A2.6 MICROMEMORY CIRCUIT

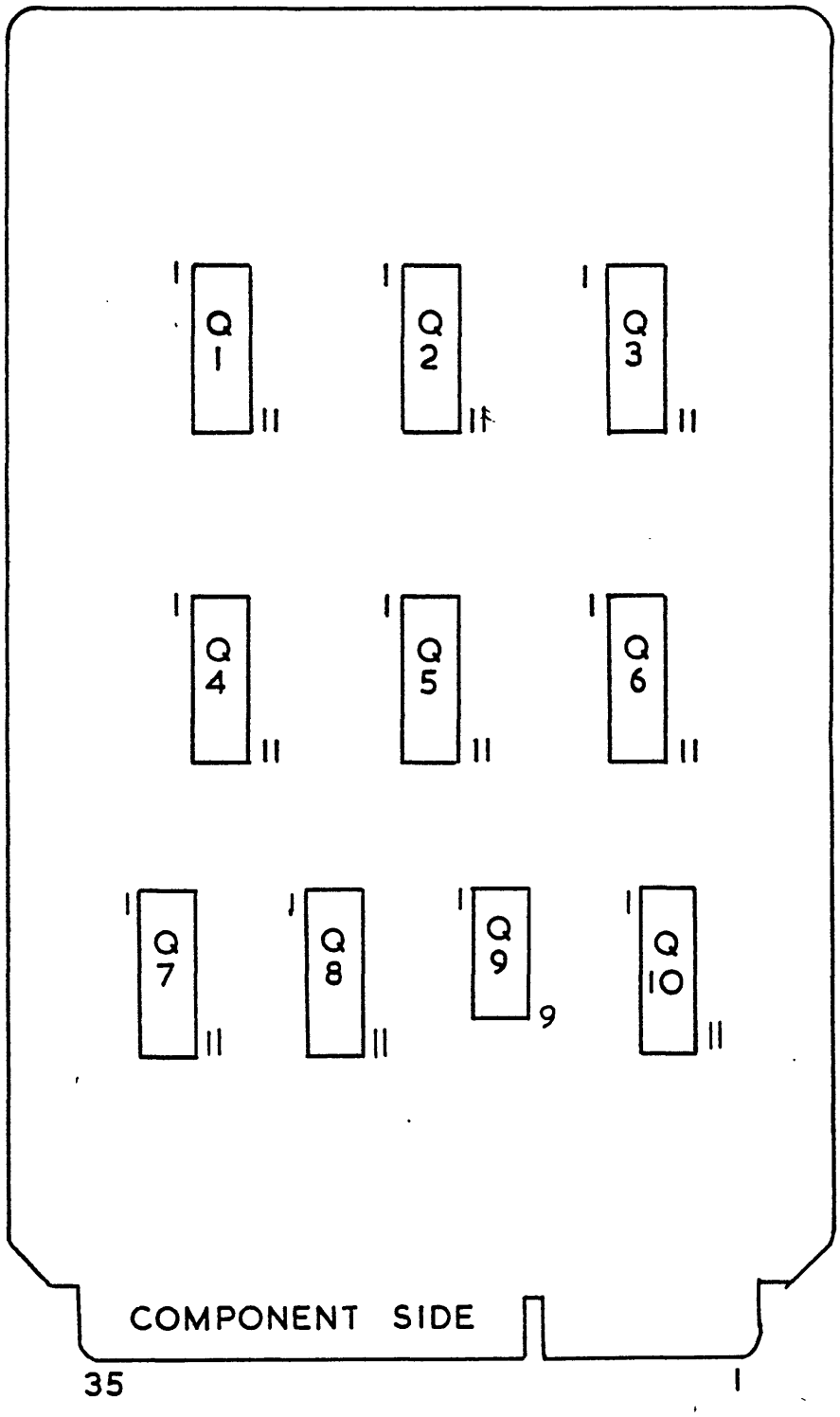


FIG A2,7 MICRO MEMORY DEVICE IDENTIFICATION

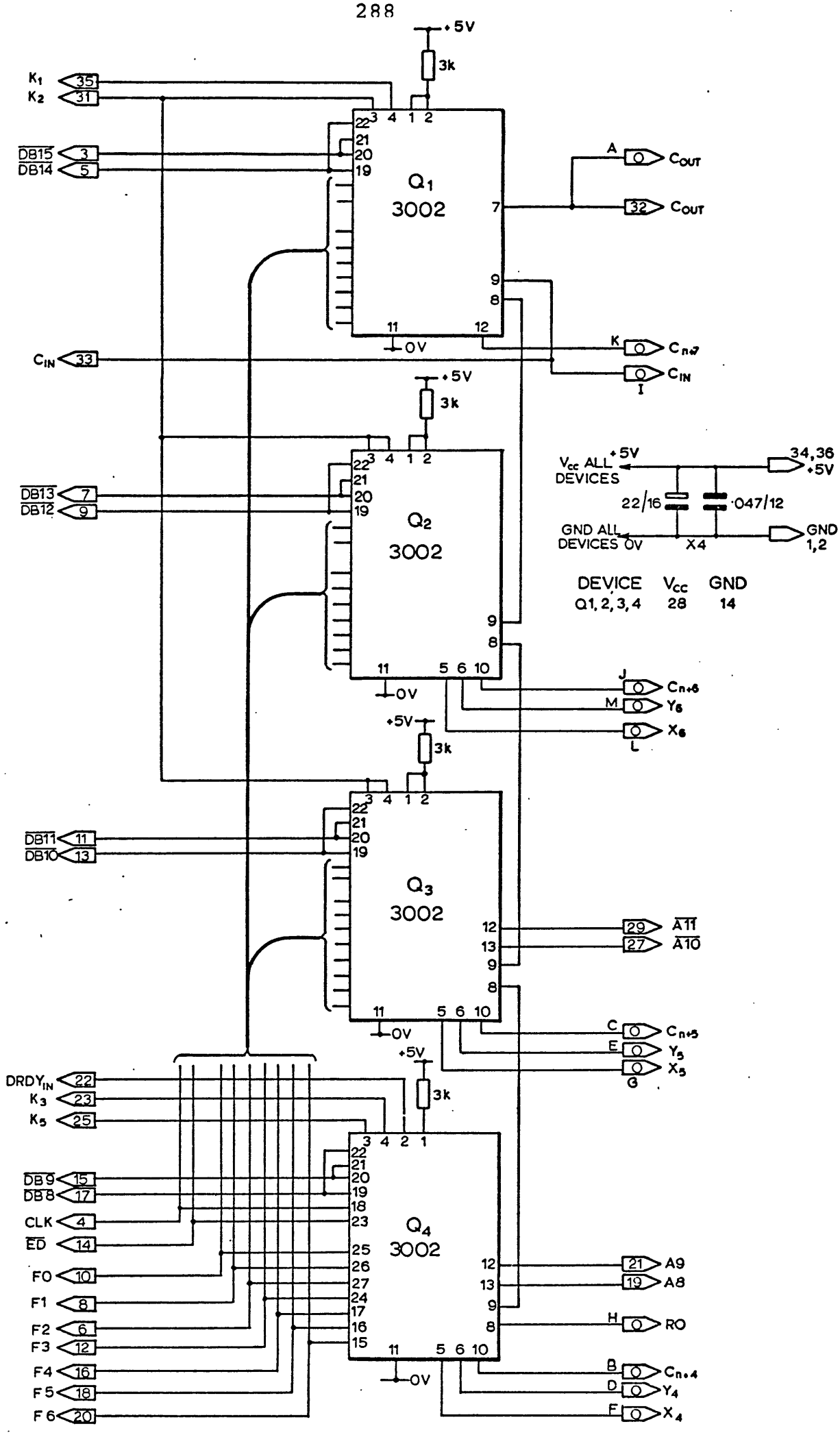


Fig. A2.8 CPU HIGH BYTE CIRCUIT

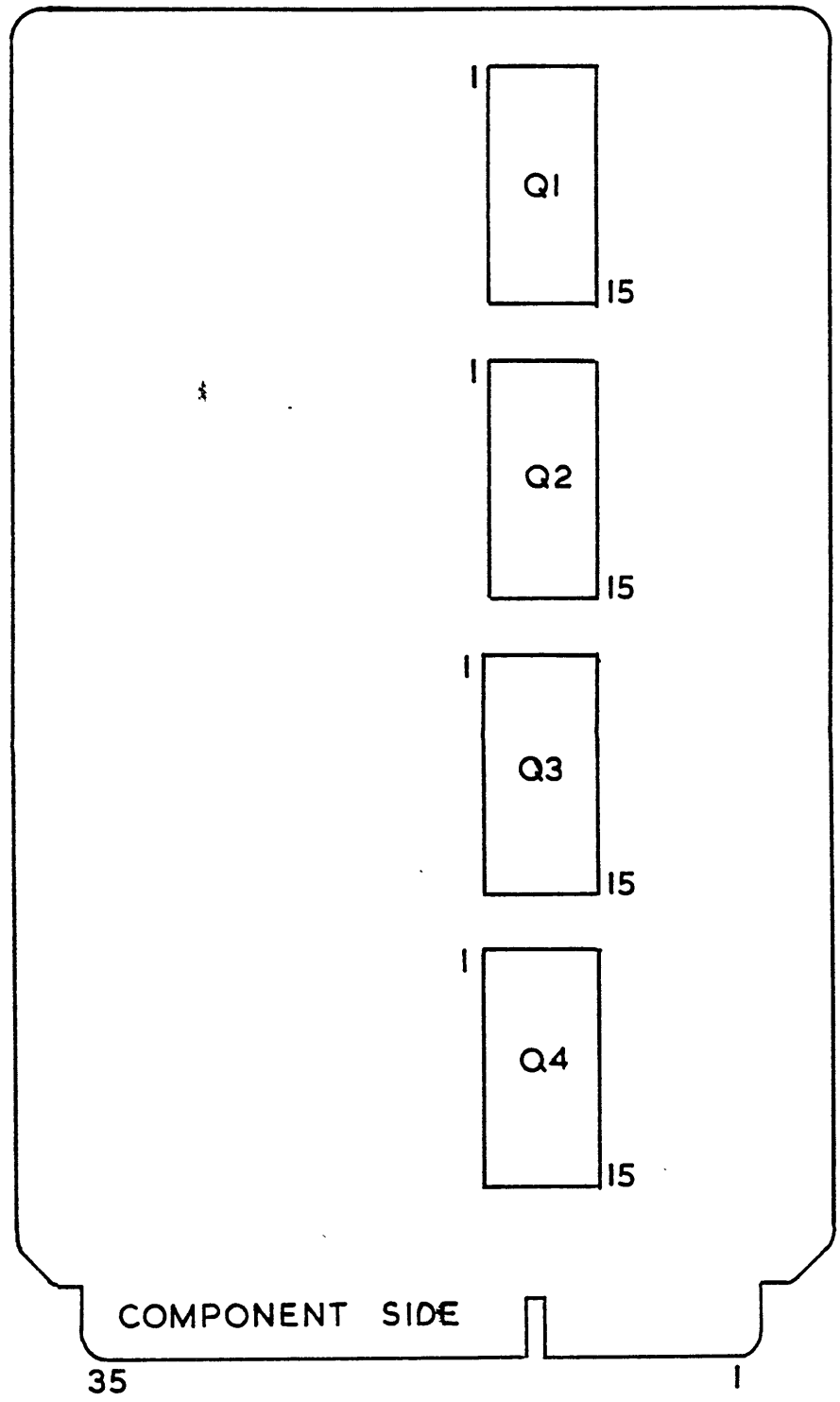


FIG. A2.9 C.P.U. (HIGH BYTE) DEVICE IDENTIFICATION

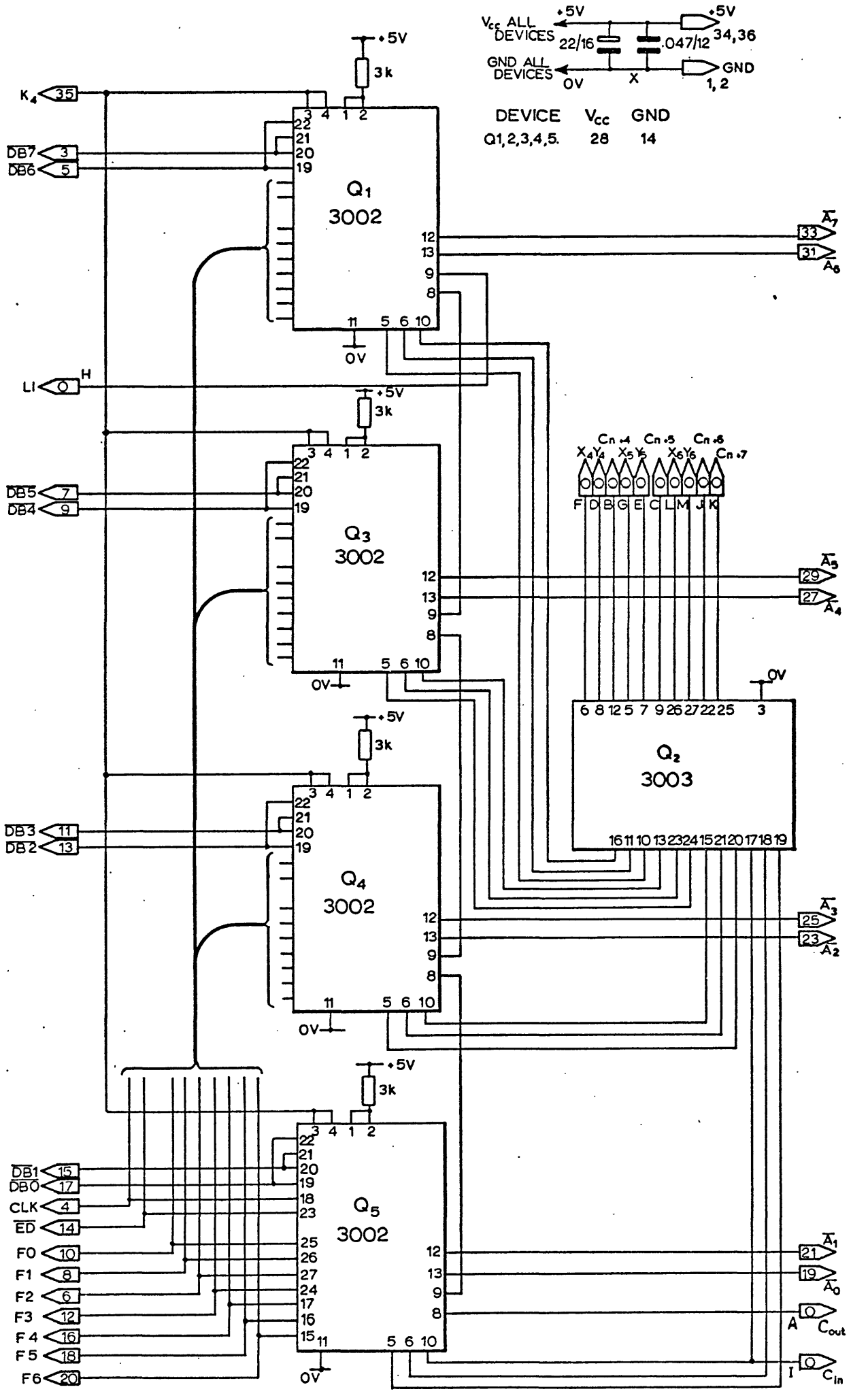


Fig. A.2.10 CPU LOW BYTE CIRCUIT

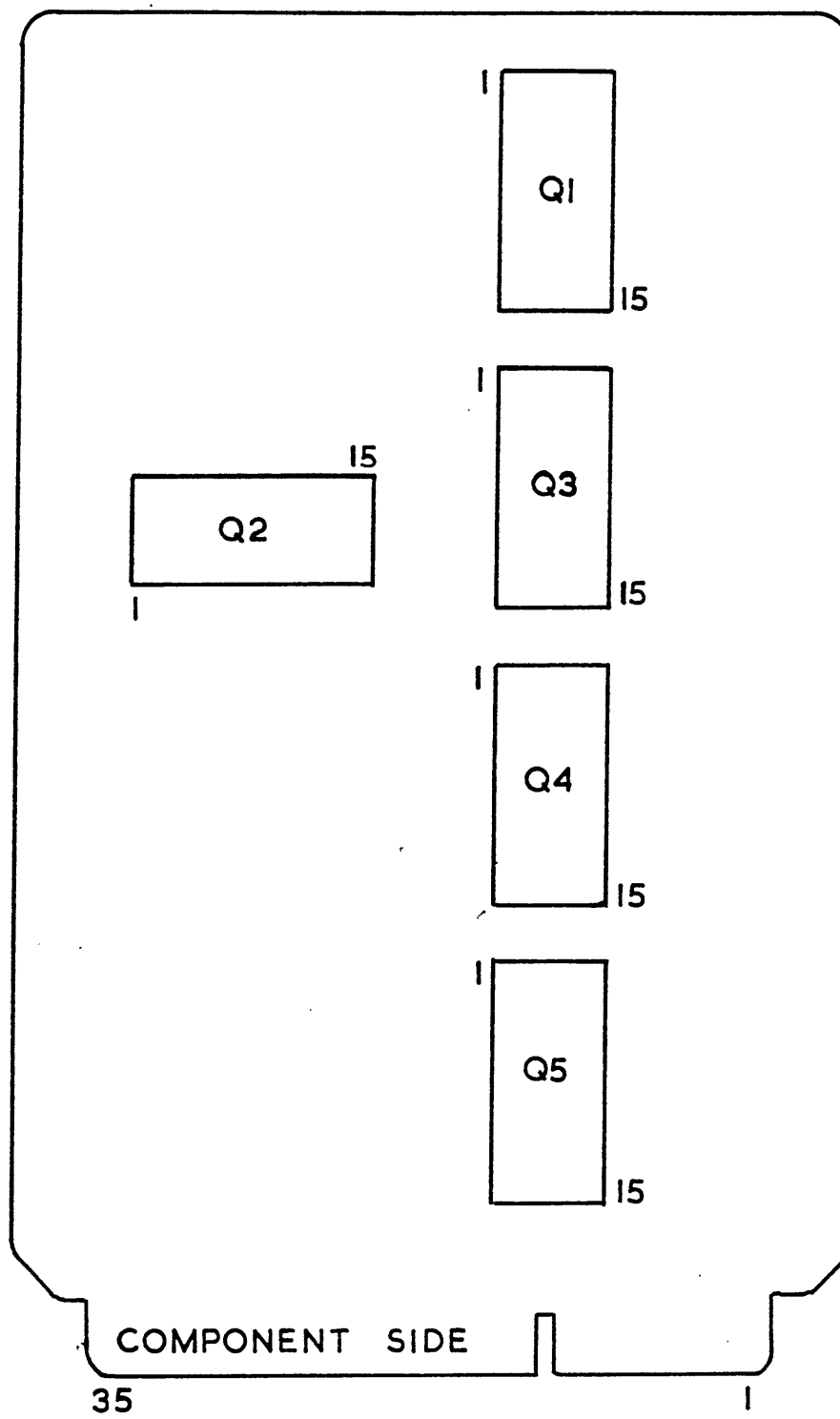


FIG. A2.11 C.P.U. (LOW BYTE) DEVICE IDENTIFICATION

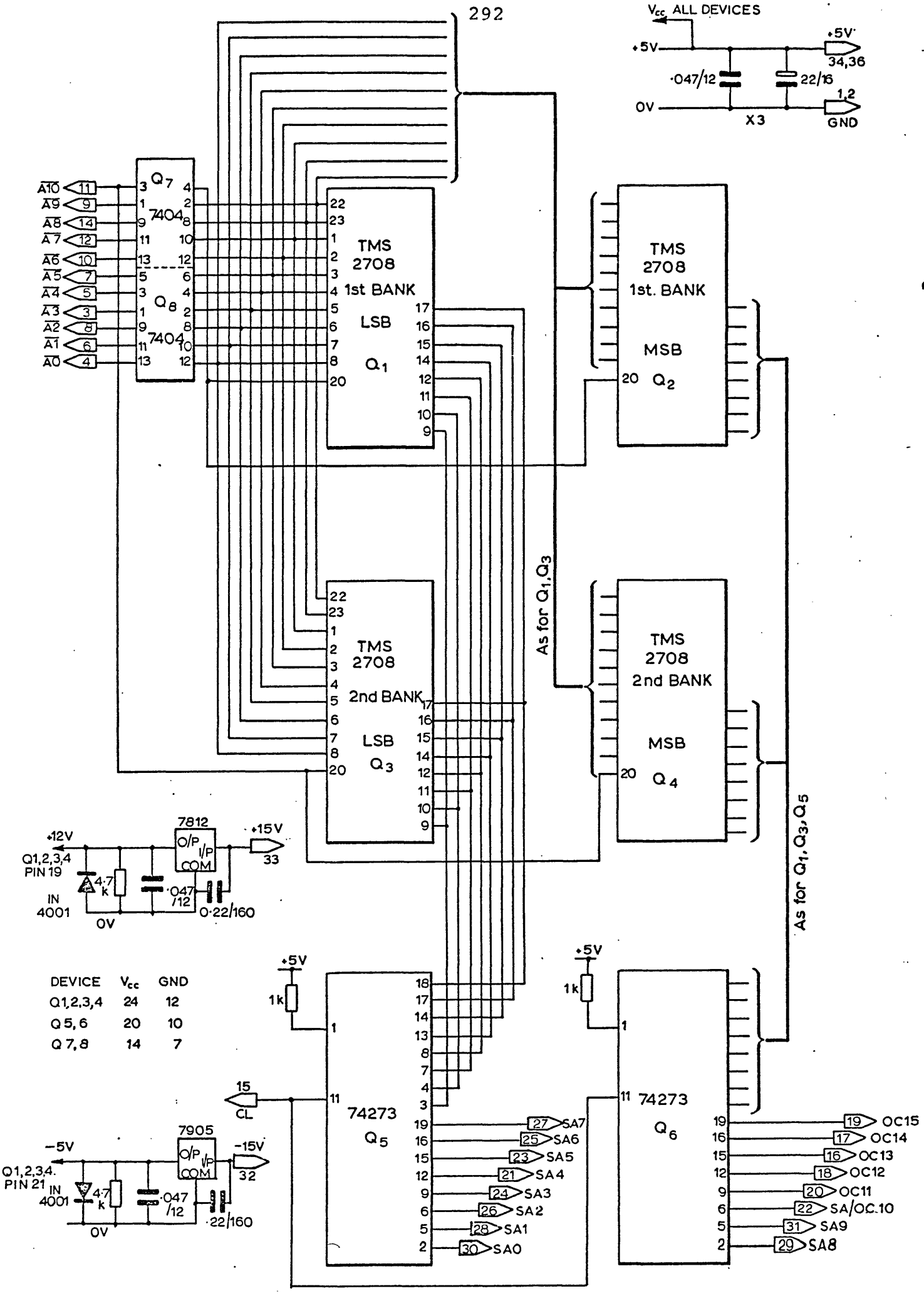


Fig A2.12 MACROMEMORY CIRCUIT

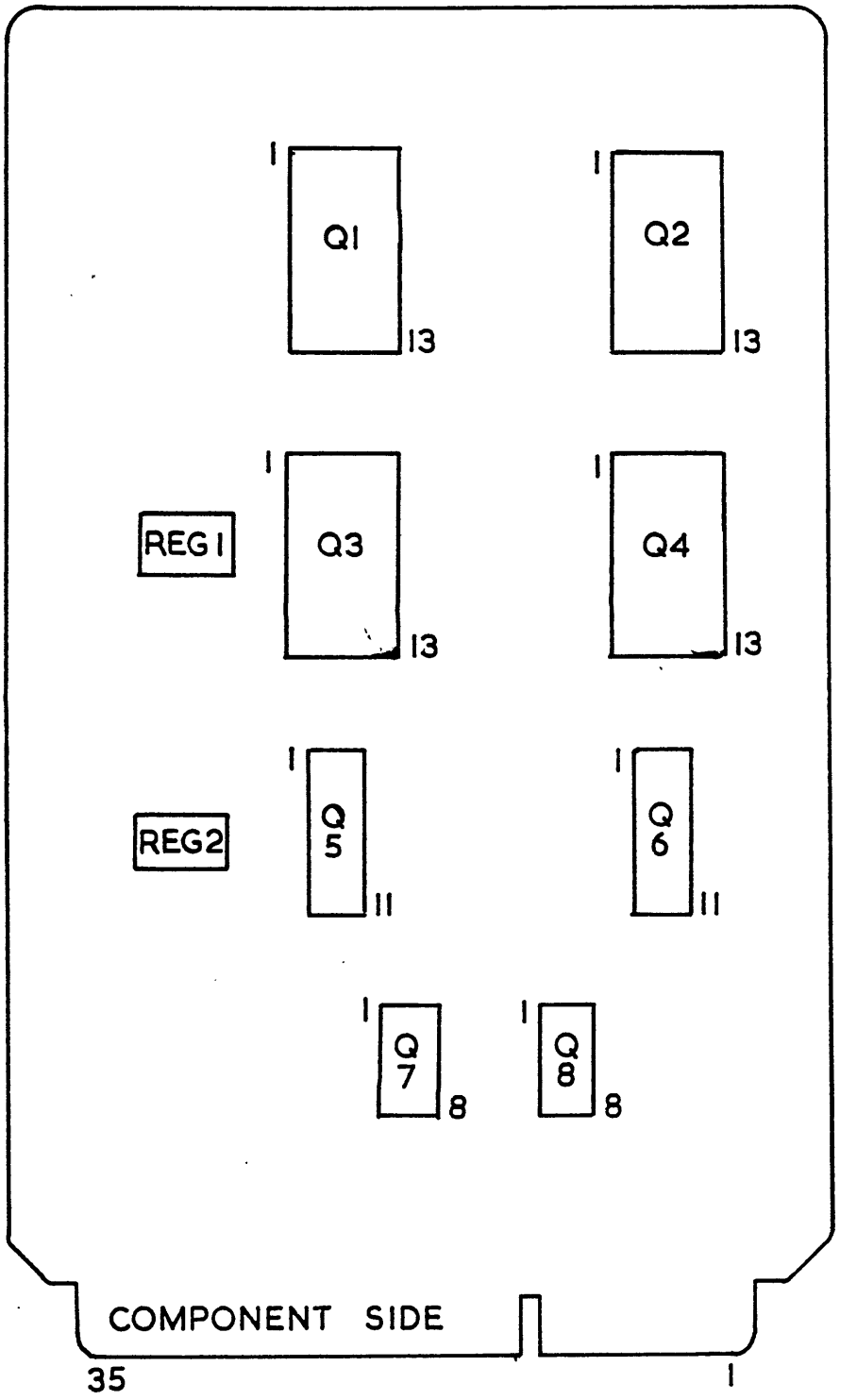


FIG. A2.13 MACRO MEMORY DEVICE IDENTIFICATION

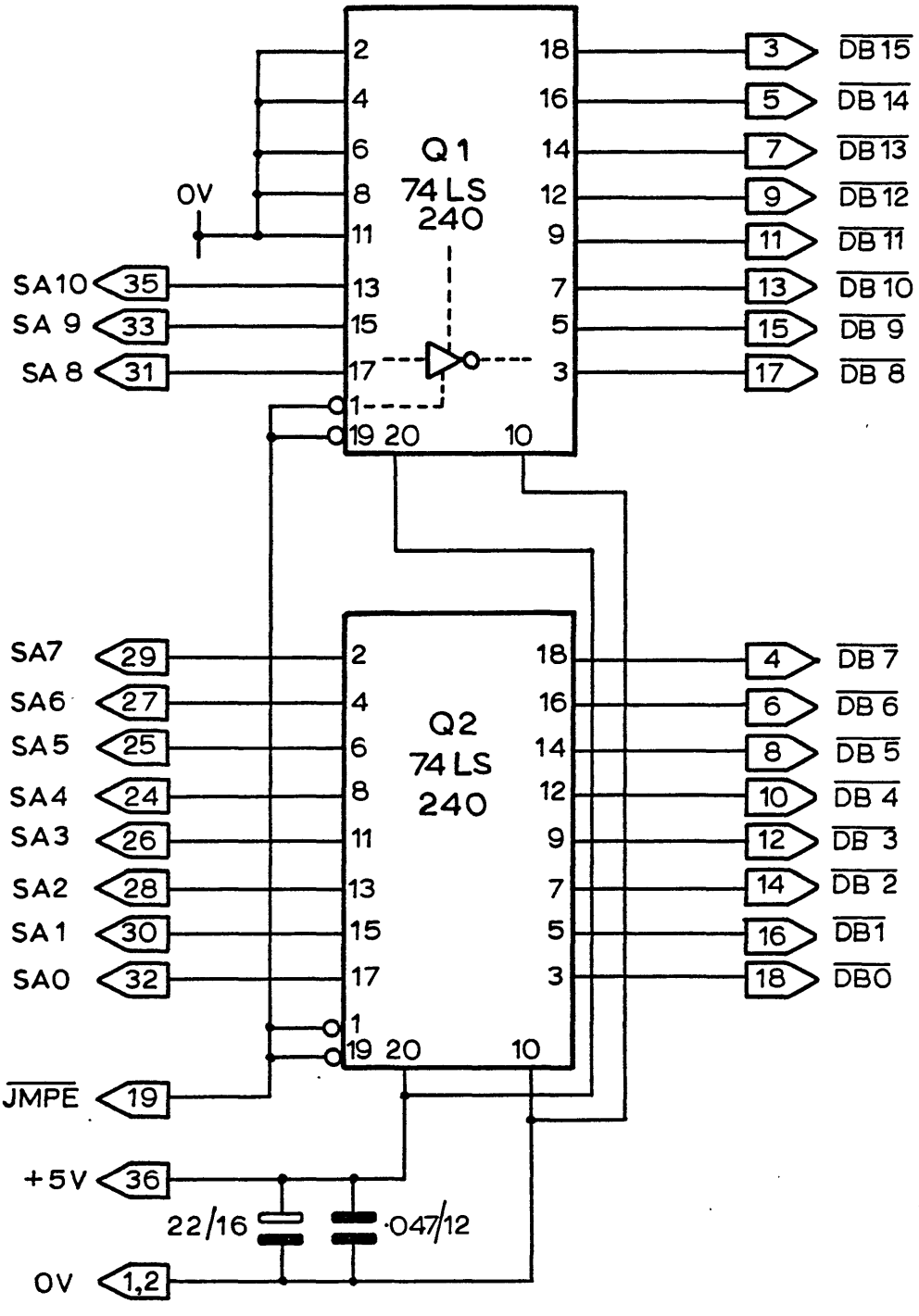


Fig. A2·14 MACROMEMORY BUS DRIVERS CIRCUIT

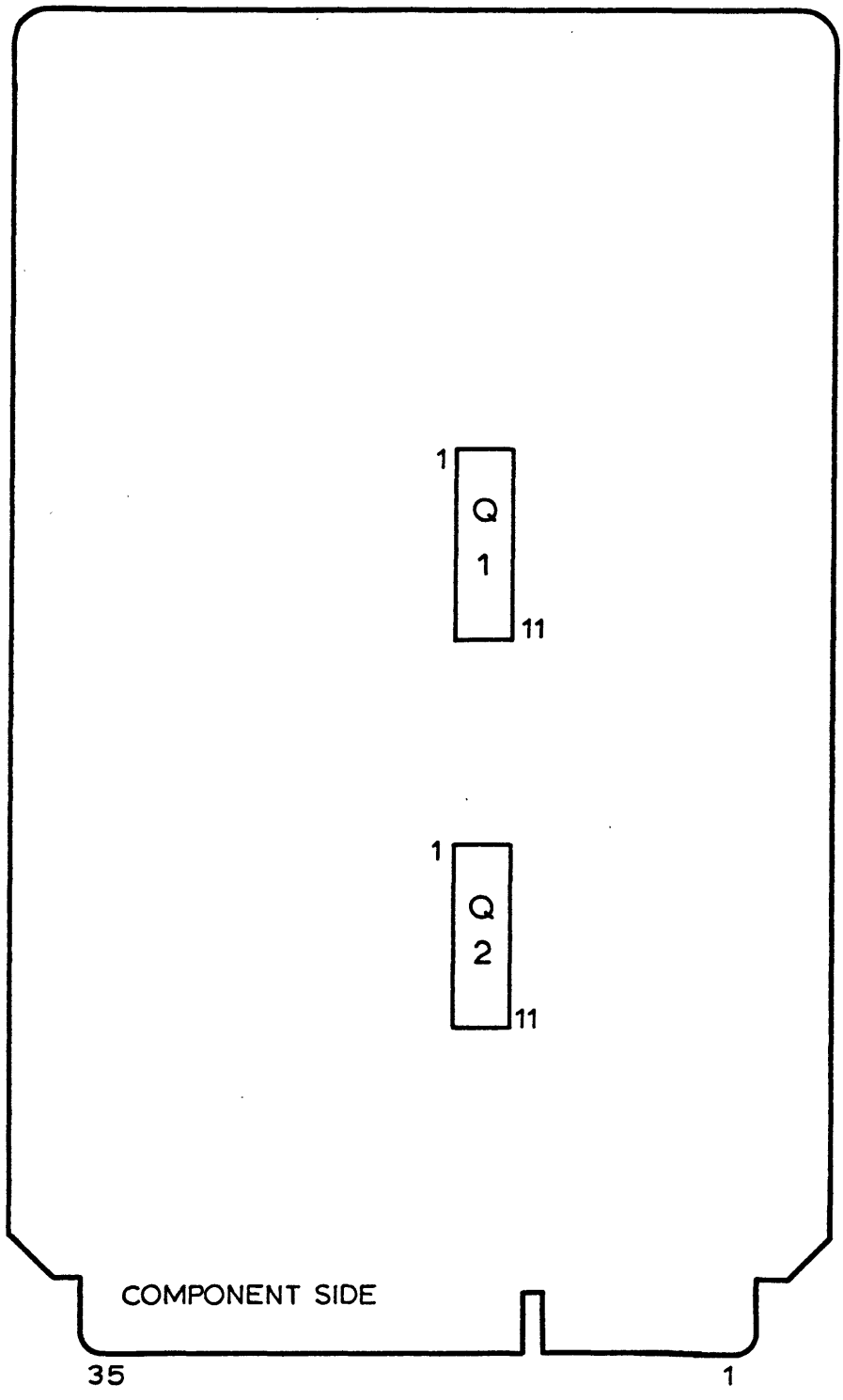


Fig. A2·15 MACROMEMORY BUS DRIVERS DEVICE IDENTIFICATION.

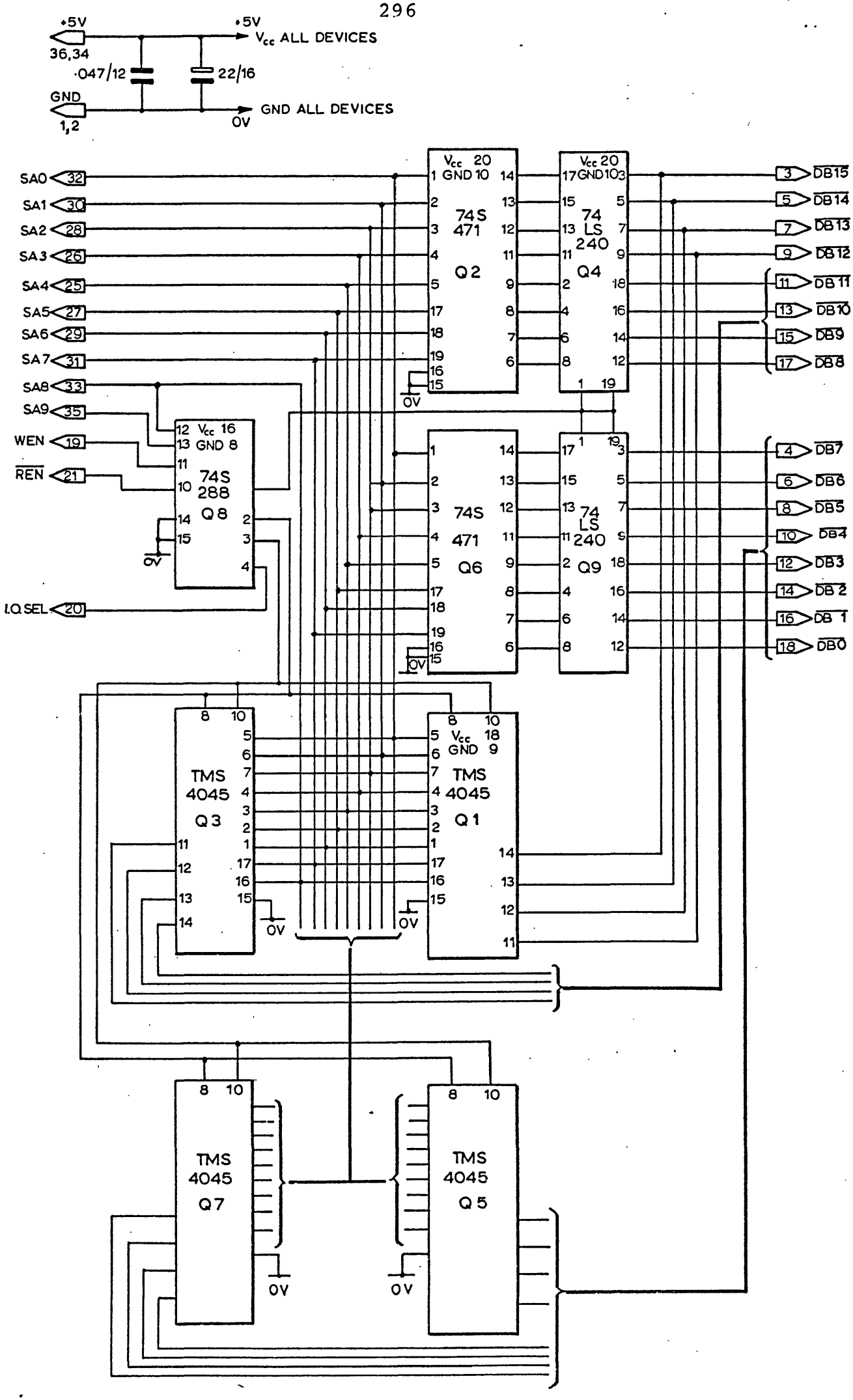


Fig. A2·16 DATA-CONSTANT MEMORY CIRCUIT

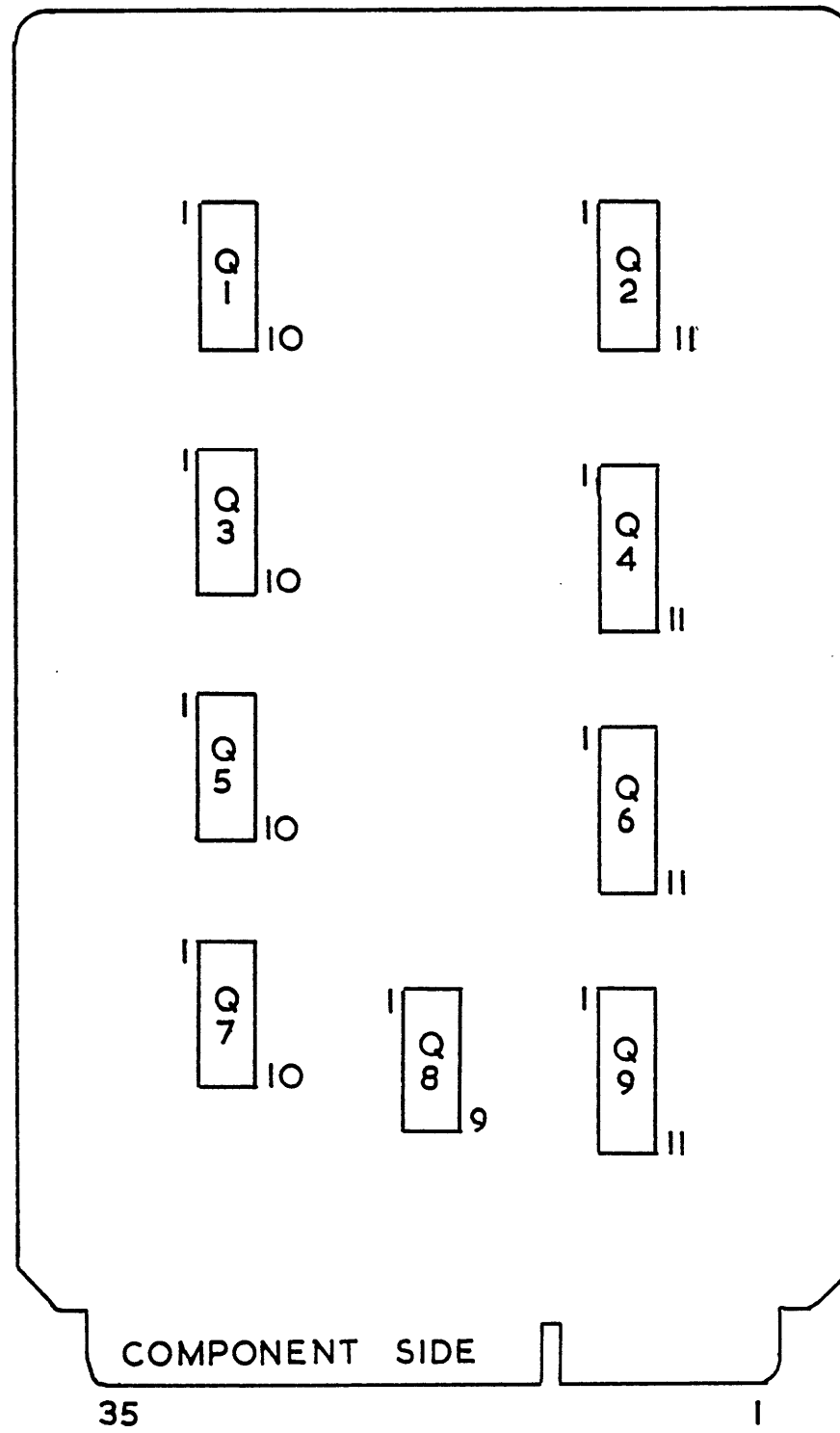


FIG. A2.17 DATA CONSTANT MEMORY DEVICE IDENTIFICATION

ADDRESS

A = $\overline{\text{REN}}$, B = WEN

C = SA8, D = SA9

E not used set to "0"

DATA

1 = $\overline{\text{ROM EN}}$, 2 = $\overline{\text{CE}}$

3 = $\overline{\text{WE}}$, 4 = I/O SEL.

5,6,7,8 not used.

D	C	B	A	4	3	2	1	
0	0	0	0	0	1	0	1	Ram read
0	0	0	1	0	1	1	1	
0	0	1	0	0	1	1	1	
0	0	1	1	0	0	0	1	Ram write
0	1	0	0	0	1	0	1	Ram read
0	1	0	1	0	1	1	1	
0	1	1	0	0	1	1	1	
0	1	1	1	0	0	0	1	Ram write
1	0	0	0	0	1	1	0	Rom read
1	0	0	1	0	1	1	1	
1	0	1	0	0	1	1	1	
1	0	1	1	0	1	1	1	
1	1	0	0	1	1	1	1	I/O
1	1	0	1	1	1	1	1	select
1	1	1	0	1	1	1	1	
1	1	1	1	1	1	1	1	

Table A2.1 Data-constant memory address/control decoding ROM(Q8) map.

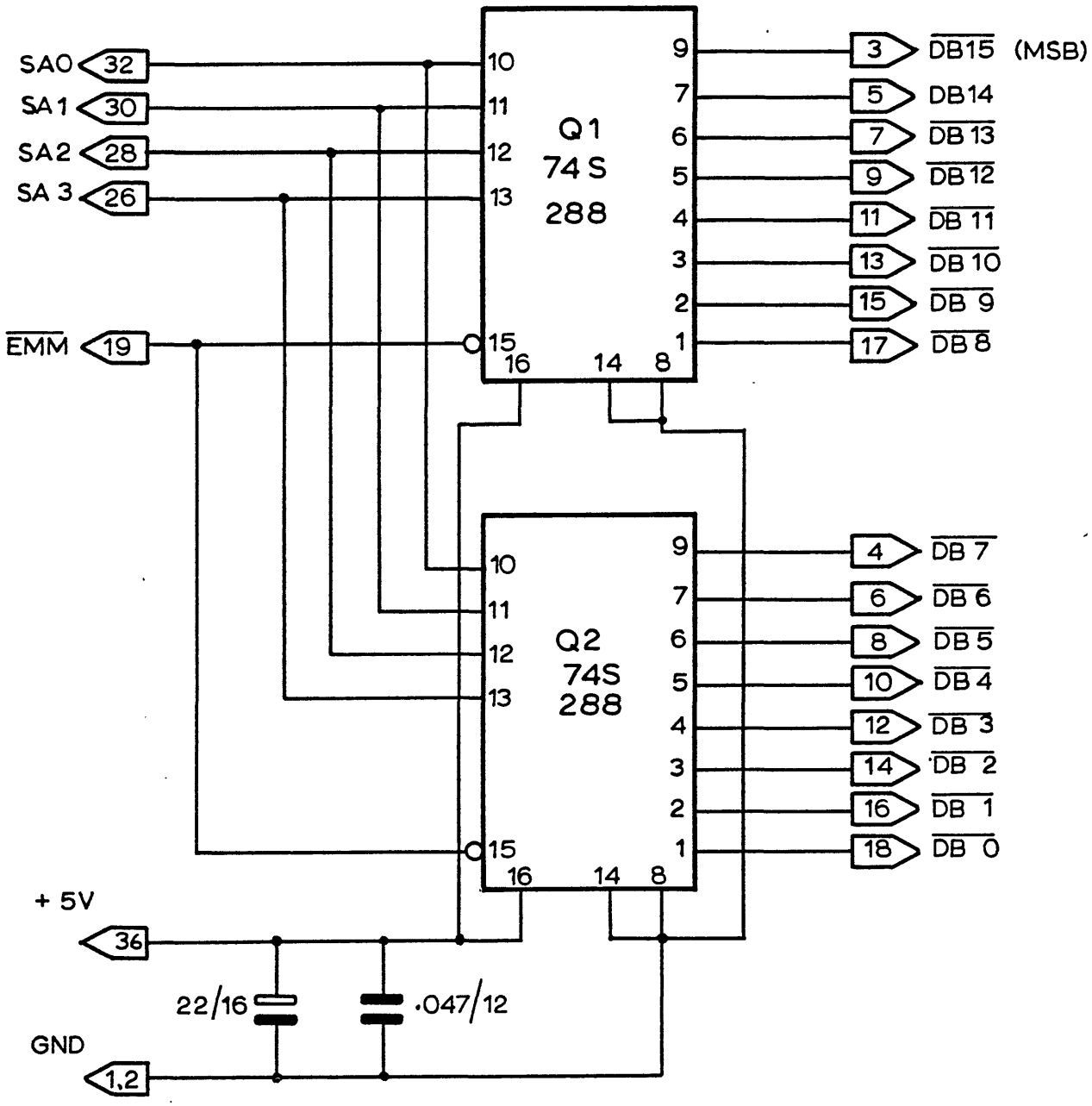


Fig. A2·18 BIT MASKING MEMORY CIRCUIT

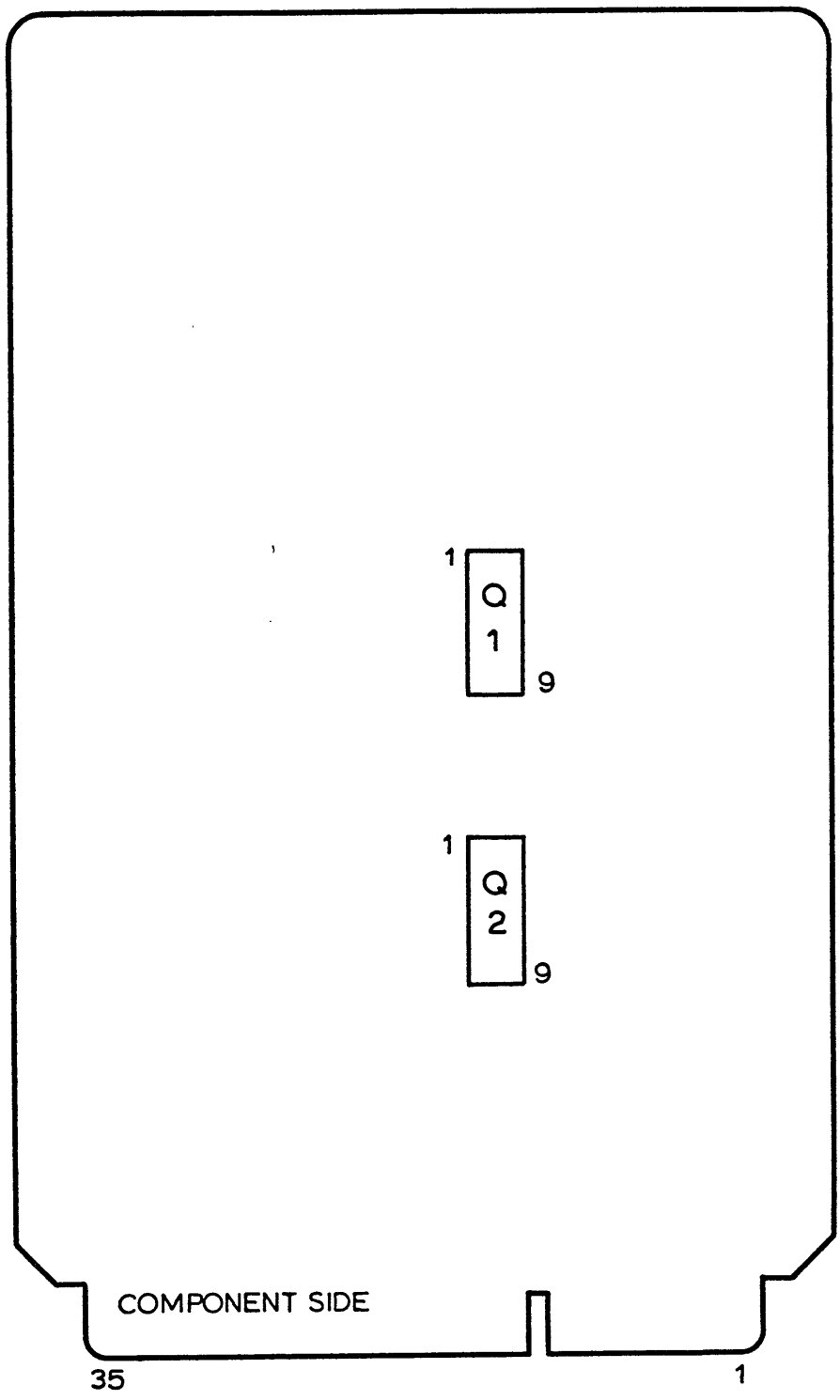


Fig. A2·19 BIT MASKING MEMORY DEVICE IDENTIFICATION

ADDRESS	DATA ROM 1	DATA ROM 2	MASK BIT
D----A	8---------1	8---------1	
0000	11111111	11111110	0
0001	11111111	11111101	1
0010	11111111	11111011	2
0011	11111111	11110111	3
0100	11111111	11101111	4
0101	11111111	11011111	5
0110	11111111	10111111	6
0111	11111111	01111111	7
1000	11111110	11111111	8
1001	11111101	11111111	9
1010	11111011	11111111	10
1011	11110111	11111111	11
1100	11101111	11111111	12
1101	11011111	11111111	13
1110	10111111	11111111	14
1111	01111111	11111111	15

Address E set to "0"

Table A2.2 Bit masking memory ROM contents map.

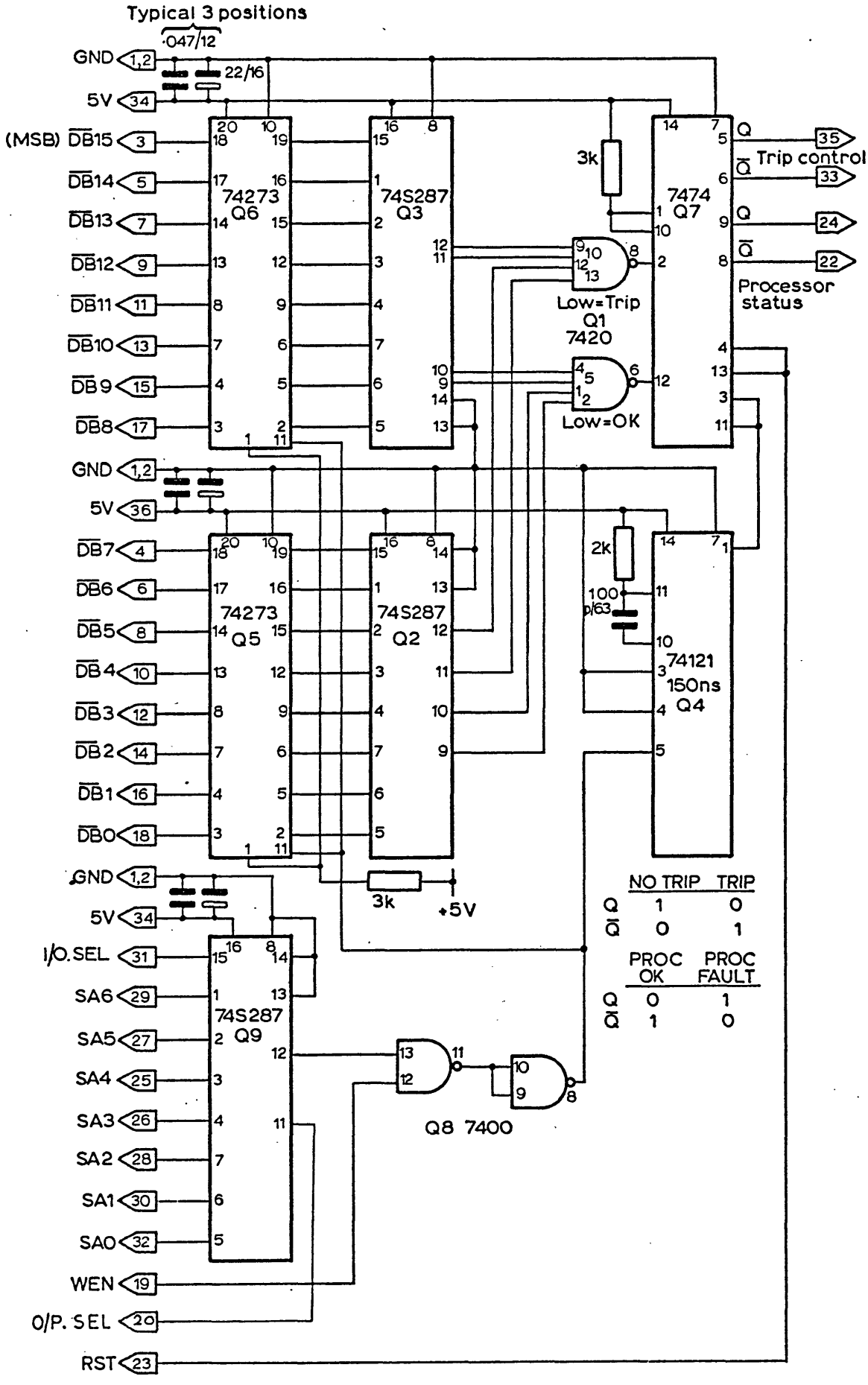


Fig. A2·20 CIRCUIT BREAKER CONTROL PORT CIRCUIT

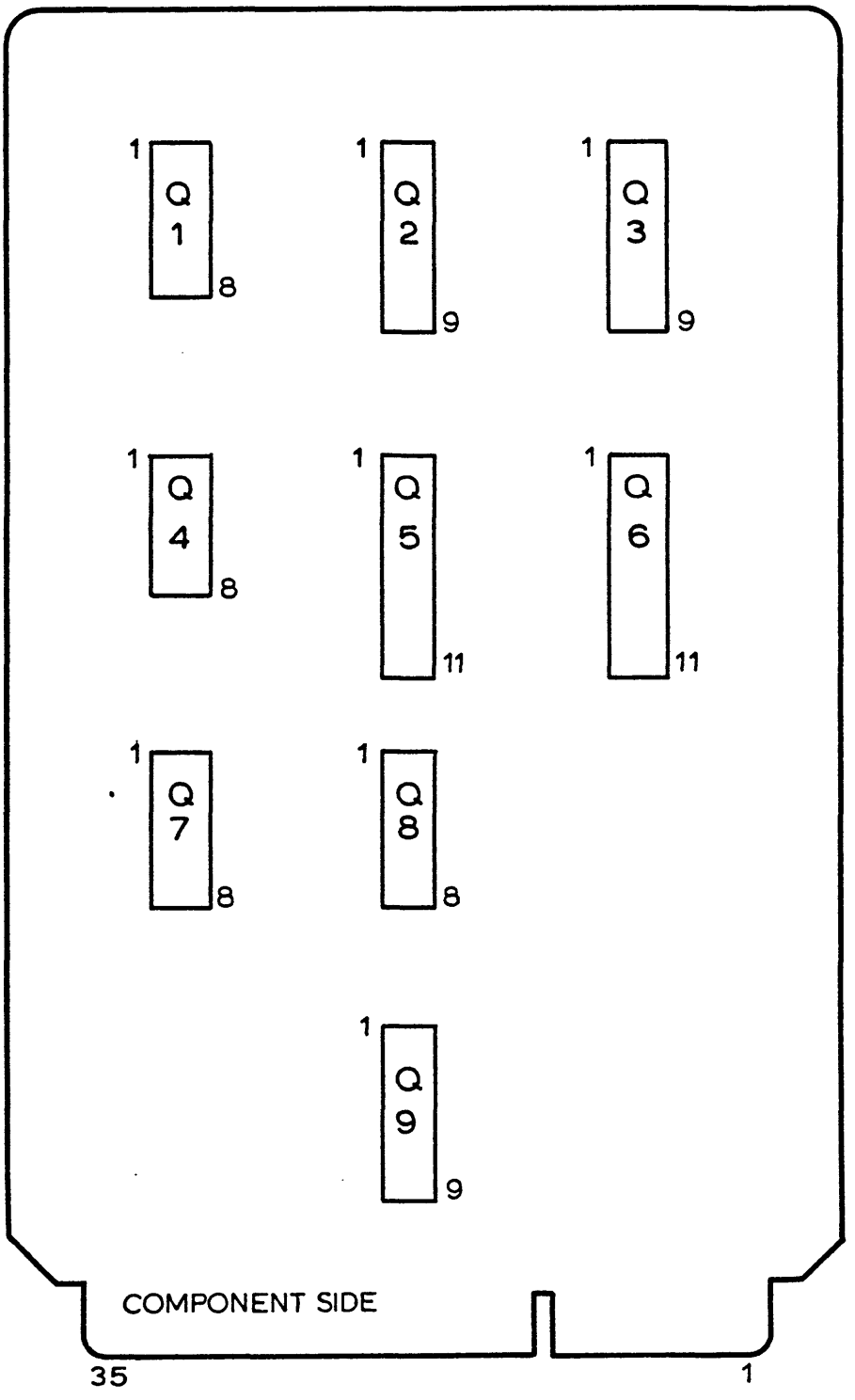


Fig. A2·21 CIRCUIT BREAKER CONTROL PORT DEVICE IDENTIFICATION.

ADDRESS	DATA ROM 1	DATA ROM 2
H------A	O_1 O_2 O_3 O_4	O_1 O_2 O_3 O_4
10101010	1 1 1 1	1 1 1 1
01010101	0 0 1 1	0 0 1 1
ALL OTHER ADDRESSES	0 0 0 0	0 0 0 0

Table A2.3 Breaker control port, control word decoding ROM (Q2,3) map.

ADDRESS	DATA	FUNCTION
H--------A	O_2 O_1	
111 111 11	0 1	Breaker control Port enable .
101 111 11	1 0	Communication port enable.
ALL OTHER ADDRESSES	0 0	

Table A2.4 Breaker control port, address decoding ROM (Q9) map.

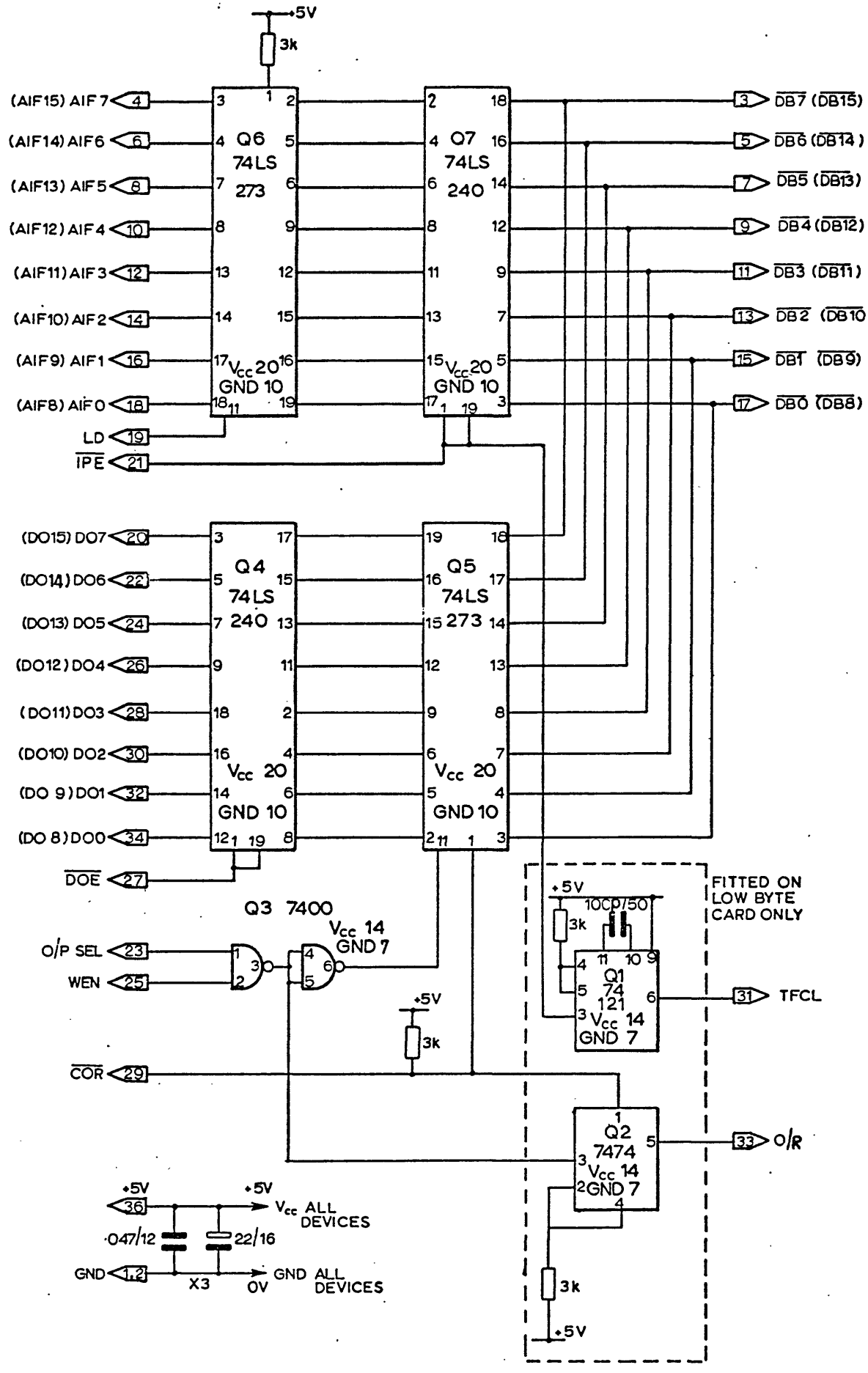


Fig. A2.22 INPUT/COMMUNICATION OUTPUT PORT LOW BYTE (HIGH BYTE) CIRCUIT

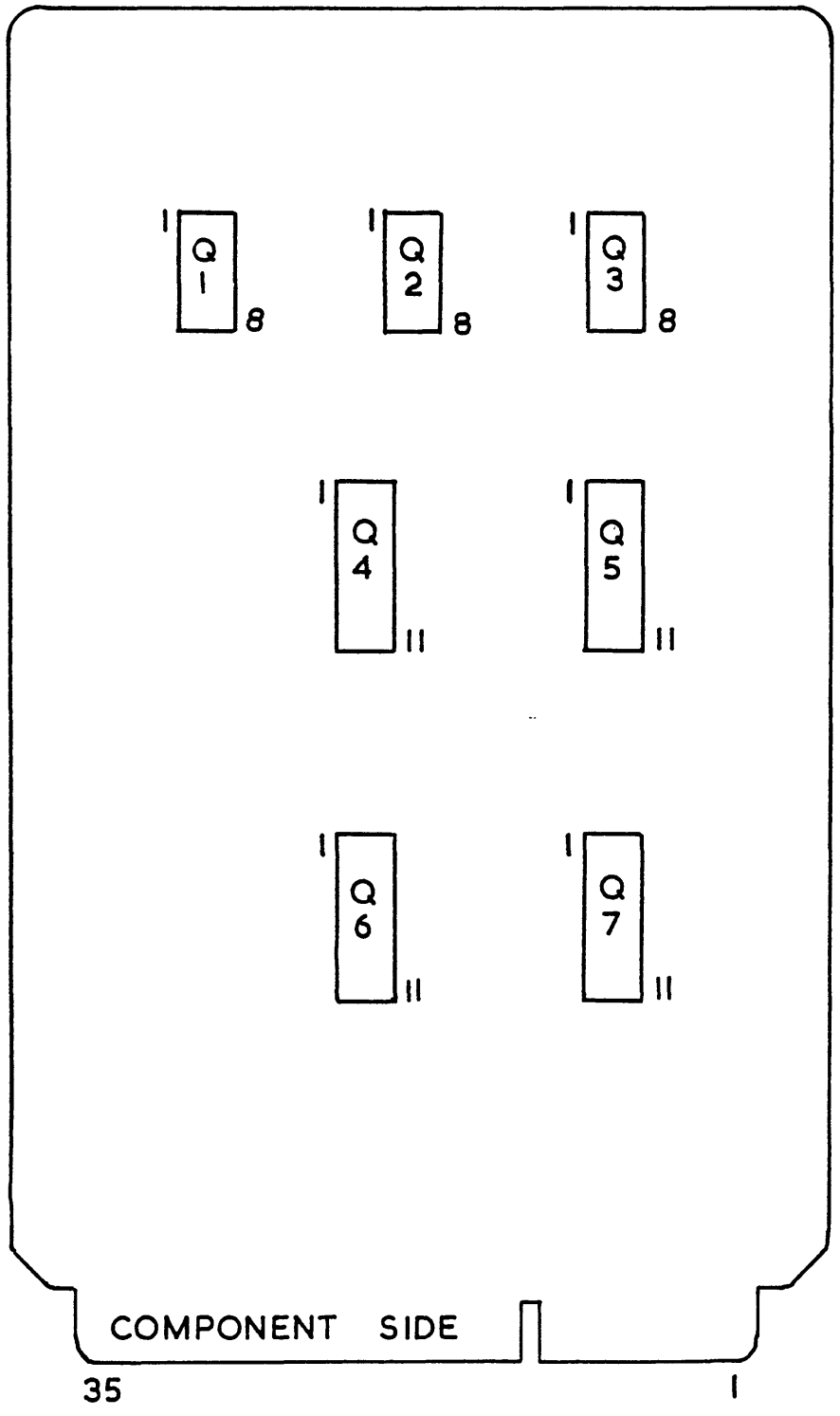


FIG. A2.23 INPUT/COMMUNICATION PORT DEVICE IDENTIFICATION

OUTPUT SOCKET (Mc MURDO RED-RANGE 24 WAY).

PIN NO	FUNCTION	PIN NO	FUNCTION
1	DATA OUT 0 (LSB)	13	DATA OUT 12
2	" 1	14	" 13
3	" 2	15	" 14
4	" 3	16	" 15 (MSB)
5	" 4	17	NC
6	" 5	18	OUTPUT READY FLAG
7	" 6	19	NC
8	" 7	20	CLEAR FLAG
9	" 8	21	NC
10	" 9	22	OUTPUT ENABLE
11	" 10	23	NC
12	" 11	24	NC

NC = NOT CONNECTED.

POWER PLUG (Mc MURDO RED-RANGE 8 WAY)

PIN NO	FUNCTION
1 & 5	- 15V
2 & 6	+ 15V
3 & 7	+ 5V
4 & 8	OV(GND)

Table A2.5 Protection processor data output socket and power plug details.

INPUT PLUG (McMURDO RED-RANGE 24 WAY)

PIN NO	FUNCTION	PIN N	FUNCTION
1	DATA IN 0 (LSB)	13	DATA IN 12
2	" 1	14	" 13
3	" 2	15	" 14
4	" 3	16	" 15 (MSB)
5	" 4	17	NC
6	" 5	18	DATA READY FLAG IN
7	" 6	19	NC
8	" 7	20	TRANSFER CLOCK OUT
9	" 8	21	NC
10	" 9	22	GROUND (OV)
11	" 10	23	NC
12	" 11	24	POWER ON CLEAR IN

NC = NOT CONNECTED

BREAKER CONTROL SOCKET (Mc MURDO RED-RANGE 8 WAY)

PIN NO	FUNCTION
1	$\overline{Q}$ TRIP CONTROL
2	Q " "
3	$\overline{Q}$ RELAY STATUS
4	Q " "
5	RESET RELAY STATUS
6,7,8	NC

Table A2.6 Protection processor data input plug and breaker control socket details.

APPENDIX 3

Protection processor.

Instruction set.

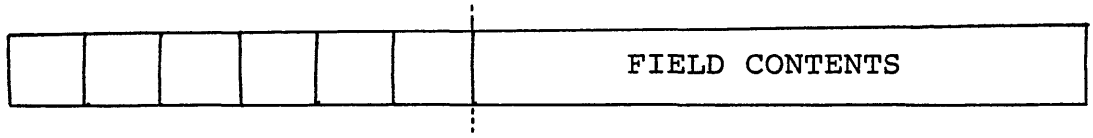
Introduction to Appendix 3.

This appendix describes the instruction set developed for the protection processor. The format adopted is as follows:

INSTRUCTION Brief description of the instruction function.

MNEMONIC The three letter mnemonic code used.

FORMAT



MICRO CODE The micro-program which implements the instruction, e.g.:

16	LDMT	JMPE	JCR6	00110100	00000000	00100000	00110110
(a)	(b)	(c)	(d)		(e)		

a) Micro-memory location address

b) CPU array control micro-function. Mnemonics used correspond to those given in reference 47.
The last figure or letter defines the processing element register employed, (if any). E.g. LDM(T)← register T, ILR(9)← scratchpad register 9, TZA(A)← accumulator. When no register is specified e.g. LMI the default register is RO, i.e. the program counter. Certain

functions, e.g. NOP have no effect on any of the registers and in these cases the register specifier is again omitted.

c) This group defines any special functions which are included in the micro-instruction. They are:
i) Data bus/memory control, e.g. JMPE, REN, WEN, etc.
ii) Carry/shift flag input control, e.g. FF1 forces the flag to logical "1" (by default it is logical "o").
iii) Special K bus conditions, e.g. K1100.
Any combination of these functions may be defined here e.g. ----REN FF1----

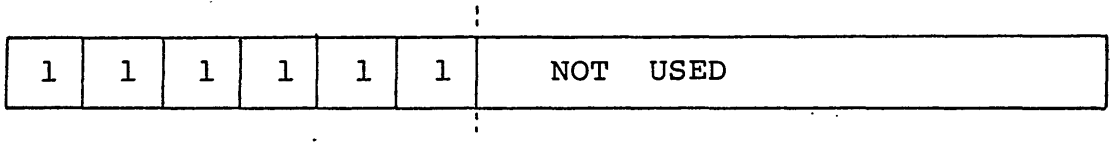
d) Micro-program jump address, i.e. the data from which the address of the next micro-instruction in the sequence is calculated. The mnemonics used here are those given in reference 51 or defined in Chapter 4.

e) The contents of the 4 micro-program memory PROMS for the address defined in (a).

INSTRUCTION Restart the processor at main memory location Ø.

MNEMONIC RST

FORMAT



MICRO-CODE

```
Ø  CLR  JZR6          10000000  11110000  00000000  00100110
6  LMI  FF1  JZR7     00100000  11110000  00000011  00100111
7  CLAA JZR15         10010110  11110000  00000000  00101111
```

Loc 15 is the start of a short fetch, whilst 14 is an extended fetch for use with short micro-code routines which are executed before macro memory access is possible.

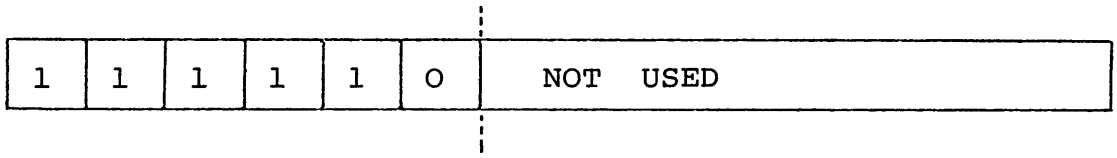
```
14  SDR9  FF1  CL  LD  01010010  00001000  00000011  10000000
15  SDR9  FF1  JZR14   01010010  00000000  00000011  00101110
```

The SDR9 instruction is used for accumulator restoration purposes.

INSTRUCTION Halt the processor.

MNEMONIC HLT.

FORMAT



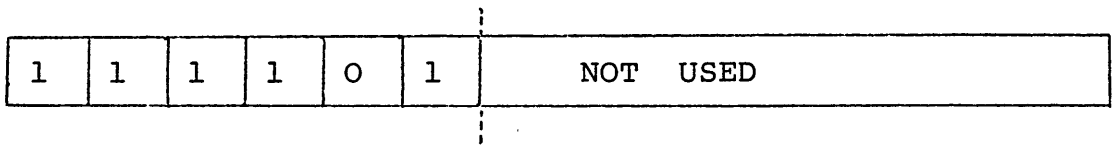
MICRO-CODE

```
1   NOP  JZR1   11000000   11110000   00000000   00100001
```

INSTRUCTION Wait until the input data ready flag is set (i.e. logical 1).

MNEMONIC WFS

FORMAT



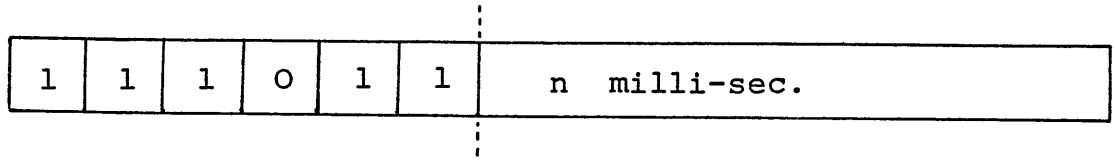
MICRO-CODE

```
8   LMI  FF1  JZR5    00100000   11110000   00000011   00100101
5   ORIT K1101 JZR4   11011100   11010000   00000000   00100100
4   NOP  JFL  0       11000000   11110000   00000000   01000000
3   NOP  JZR5         11000000   11110000   00000000   00100101
2   NOP  JZR13        11000000   11110000   00000000   00101101
```

INSTRUCTION — Wait for n milli-secs., where n is specified by the contents of the data field. This instruction must be preceded by loading the accumulator with the constant 1664. i.e. LDA loc where (loc) = 1664.

MNEMONIC — PAU.

FORMAT



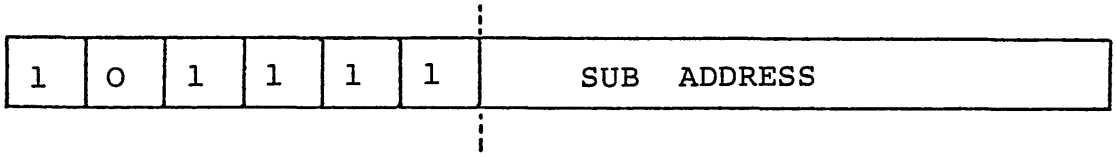
MICRO CODE

```
16  LDMT JMPE JCR6    00110100 00000000 00100000 00110110
22  LMI  FF1  JCR3    00100000 11110000 00000011 00110011
19  DCAA      JCR4    00111110 00000000 00000000 00110100
20  TZAA      JCR5    10111110 00000000 00000000 00110101
21  NOP       JFL1    11000000 11110000 00000000 01000001
18  TZRT K1100 JCR15  10111100 11000000 00000000 00111111
31  ILR9      JFL1    00010010 11110000 00000000 01000001
26  SDR9 FF1 CL LD    01010010 00001000 00000011 10000000
27  DCAT      JCR3    00111100 00000000 00000000 00110011
```

INSTRUCTION Load the accumulator from the data/constant store as specified by the sub address.

MNEMONIC LDA

FORMAT



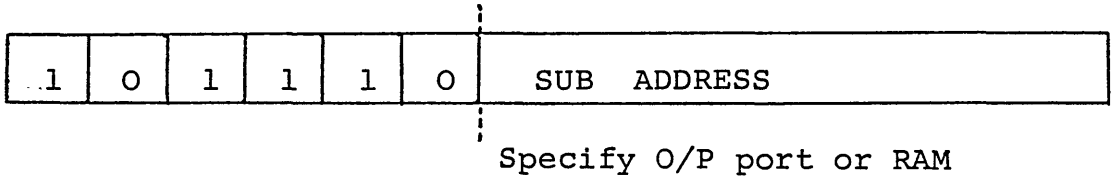
MICRO-CODE

64	LMI FF1 REN JCR6	00100000	11110000	00110011	00110110
70	NOP REN JCR 7	11000000	11110000	00110000	00110111
71	LMFA REN JZR 14	11010110	11110000	00110000	00101110

INSTRUCTION Store the accumulator in the data store location or output port as specified by the sub-address.

MNEMONIC STA

FORMAT



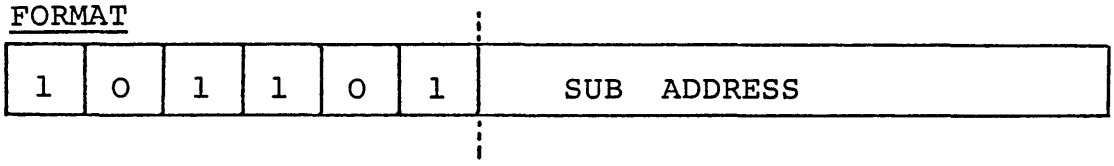
MICRO-CODE

65	LMI FF1 EDB JCR4	00100000	11110000	01000011	00110100
68	NOP EDB WEN JCR5	11000000	11110000	11000000	00110101
69	NOP EDB WEN JZR14	11000000	11110000	11000000	00101110

INSTRUCTION Add (2's complement) the data in the constant/data store location as specified by the sub-address to the contents of the accumulator.

MNEMONIC ADD

FORMAT



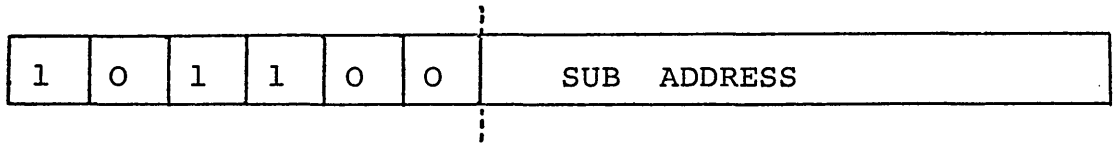
MICRO-CODE

72	LMI	FF1 REN	JCR14	00100000	11110000	00110011	00111110
78	NOP	REN	JCR15	11000000	11110000	00110000	00111111
79	AMAA	REN	JZR14	00010110	00000000	00110000	00101110

INSTRUCTION Logically 'and' the contents of the constant/data store location as specified by the sub-address to the contents of the accumulator.

MNEMONIC AND.

FORMAT



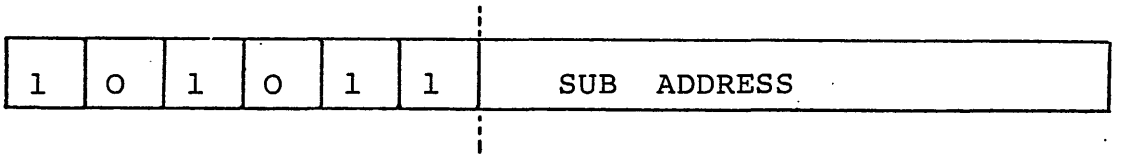
MICRO-CODE

73	LMI	FF1 REN	JCR12	00100000	11110000	00110011	00111100
76	NOP	REN	JCR13	11000000	11110000	00110000	00111101
71	ANMA		JZR14	10010110	00000000	00110000	00101110

INSTRUCTION Exclusive or the contents of the data/constant store location as specified with the contents of the accumulator.

MNEMONIC XOR

FORMAT



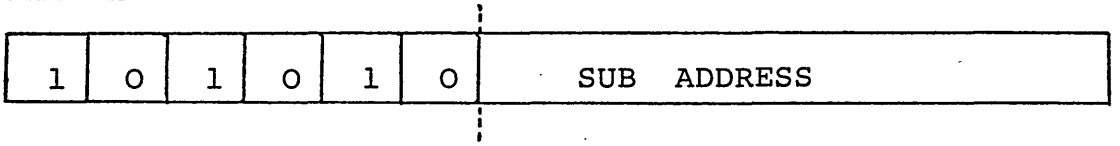
MICRO-CODE

80	LMI	FF1 REN	JCR5	00100000	11110000	00110011	00110101
85	NOP	REN	JCR6	11000000	11110000	00110000	00110110
86	XNMA	REN	JCR7	11110110	00000000	00110000	00110111
87	CMAA	JZR 14		11111110	11110000	00000000	00101110

INSTRUCTION Subtract the contents of the data/constant store location as specified from the contents of the accumulator.

MNEMONIC SUB

FORMAT



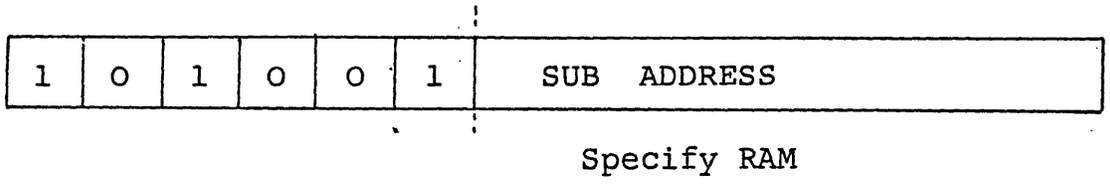
MICRO-CODE

81	LMI	FF1 REN	JCR2	00100000	11110000	00110011	00110010
82	NOP	REN	JCR3	11000000	11110000	00110000	00110011
83	LCMT	REN	JCR4	11110100	11110000	00110000	00110100
84	ALRT	FF1	JZR14	00011000	00000000	00000011	00101110

INSTRUCTION Decrement the contents of the data store location as specified and skip the following instruction if the result is zero.

MNEMONIC DSZ

FORMAT



Specify RAM

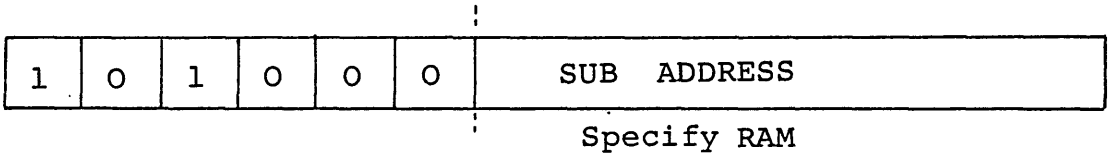
MICRO-CODE

88	NOP REN JCR15	11000000	11110000	00110000	00111111
95	NOP REN JCC6	11000000	11110000	00110000	00000110
111	LDMA REN JCR13	00110110	00000000	00110000	00111101
109	LMI FF1 JCC5	00100000	11110000	00000011	00000101
93	TZAA EDB WEN JCR12	10111110	00000000	11000000	00111100
92	NOP EDB WEN JFL5	11000000	11110000	11000000	01000101
91	ILR9 CL LD	00010010	11111000	00000000	10000000
90	LMI FF1 JCC6	00100000	11110000	00000011	00000110
106	ILR9 JZR15	00010010	11110000	00000000	00101111

INSTRUCTION Increment the contents of the data store location as specified and skip the following instruction if the result is zero.

MNEMONIC ISZ.

FORMAT



MICRO-CODE

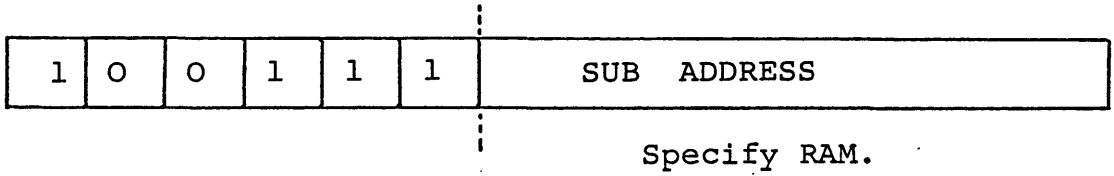
89	NOP REN JCR14	11000000	11110000	00110000	00111110
94	NOP REN JCC6	11000000	11110000	00110000	00000110
110	ACMA FF1 REN JCR13	00010110	11110000	00110011	00111101
109	LMI FF1 JCC5	00100000	11110000	00000011	00000101
93	TZAA EDB WEN JCR12	10111110	00000000	11000000	00111100
92	NOP EDB WEN JFL5	11000000	11110000	11000000	01000101
91	ILR9 CL LD	00010010	11111000	00000000	10000000
90	LMI FF1 JCC6	00100000	11110000	00000011	00000110
106	ILR9 JZR15	00010010	11110000	00000000	00101111

Shares μ code with instruction DSZ

INSTRUCTION Deposit zero in the data store location as specified by the sub-address.

MNEMONIC DZM

FORMAT



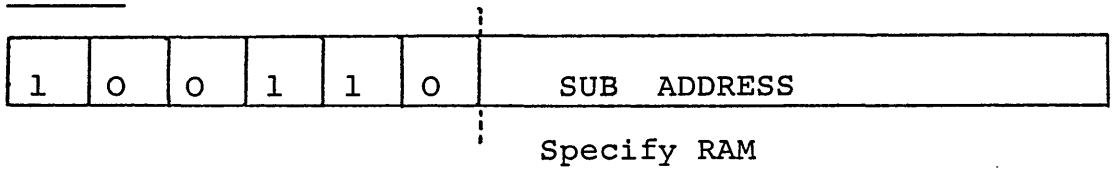
MICRO-CODE

96	LM1 FF1 JCR2	00100000	11110000	00000011	00110010
98	CLAA EDB JCC2	10010110	11110000	01000000	00000010
34	NOP EDB WEN JCR3	11000000	11110000	11000000	00110011
35	NOP EDB WEN JZR13	11000000	11110000	11000000	00101101

INSTRUCTION Sum the accumulator data in the memory location specified.

MNEMONIC SUM

FORMAT



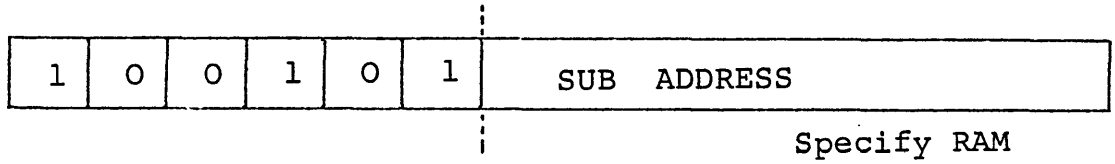
MICRO-CODE

97	LMI FF1 REN JCR3	00100000	11110000	00110011	00110011
99	NOP REN JCR4	11000000	11110000	00110000	00110100
100	AMAA REN JCR5	00010110	00000000	00110000	00110101
101	NOP JCR6	11000000	11110000	00000000	00110110
102	NOP EDB WEN JCC7	11000000	11110000	11000000	00000111
118	NOP EDB WEN JZR13	11000000	11110000	11000000	00101101

INSTRUCTION Transfer data from input port to data store location as specified.

MNEMONIC IDM.

FORMAT



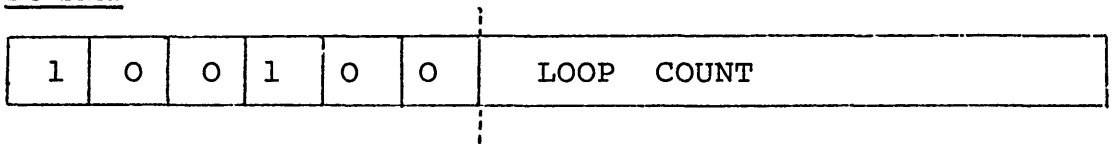
MICRO-CODE

```
104  LMI   FF1   IPE   JCR11  00100000  11110000  00010011  00111011
107  NOP   IPE   WEN   JCR12  11000000  11110000  10010000  00111100
108  NOP   IPE   WEN   JZR14  11000000  11110000  10010000  00101110
```

INSTRUCTION Load the loop counter with the data specified in the 10 least significant bits of the instruction.

MNEMONIC LDL

FORMAT



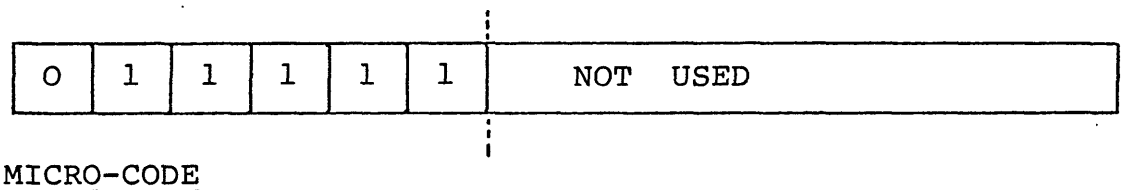
MICRO-CODE

```
105  LM1   FF1   JMPE   JCR7   00100000  11110000  00100011  00110111
103  ACMA  JMPE  JCC7           00010110  11110000  00100000  00000111
119  SDR   FF1   JZR13          01010000  00000000  00000011  00101101
```

INSTRUCTION Form the 1's complement of the accumulator contents.

MNEMONIC COM.

FORMAT



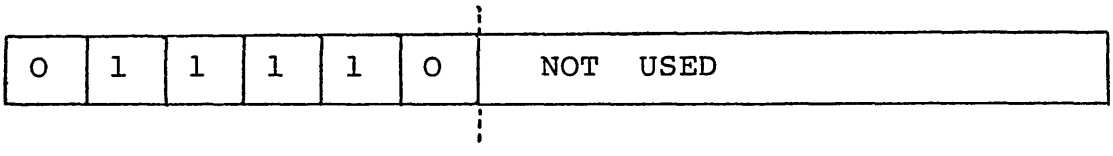
MICRO-CODE

```
128  LMI  FF1  JCR7  00100000  11110000  00000011  00110111
135  CMAA JZR15     11111110  11110000  00000000  00101111
```

INSTRUCTION Clear the accumulator.

MNEMONIC CLA

FORMAT



MICRO-CODE

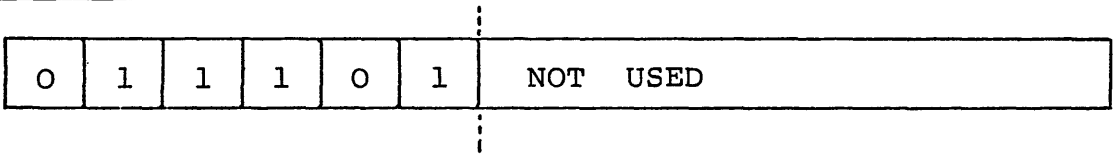
```
129  LMI FF1  JZR7   00100000  11110000  00000011  00100111
```

NOTE: Shares μ-code with instruction RST.

INSTRUCTION Form the 2's complement of the accumulator contents.

MNEMONIC NEG

FORMAT



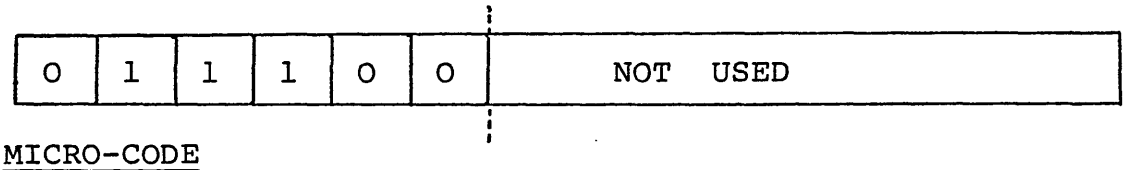
MICRO-CODE

```
136  LMI   FF1  JCR15    00100000   11110000   00000011   00111111
143  CIA   FF1  JZR15    00111110   11110000   00000011   00101111
```

INSTRUCTION Extend the sign of the A-D converter input data through the M.S.Bs of the accumulator.

MNEMONIC EXS

FORMAT



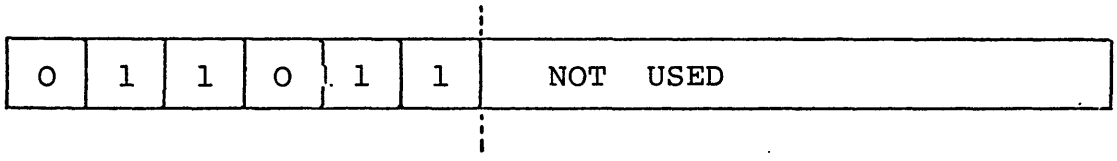
MICRO-CODE

```
137  TZAA  K1101  JCR14    10111110   11010000   00000000   00111110
142  LMI   FF1   JFL8      00100000   11110000   00000011   01001000
138  TZR9  K1110  JCR13    10110010   11100000   00000000   00111101
139  CSAA  JCR12           01010110   11110000   00000000   00111100
140  ORR9  K0011  JZR13    11010010   00110000   00000000   00101101
141  ILR9  JZR 14          00010010   11110000   00000000   00101110
```

INSTRUCTION Decrement the accumulator.

MNEMONIC DEC

FORMAT



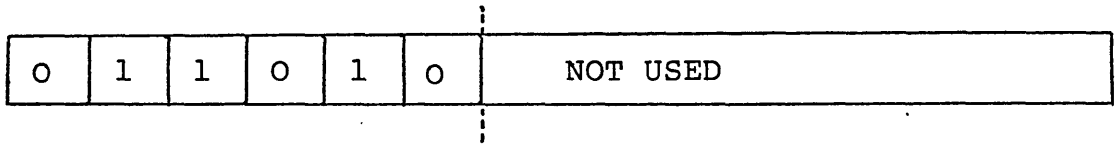
MICRO-CODE

```
144  LMI  FF1  JCR6    00100000   11110000   00000011   00110110
150  SDAA JZR15        01010110   00000000   00000000   00101111
```

INSTRUCTION Increment the accumulator.

MNEMONIC INC

FORMAT



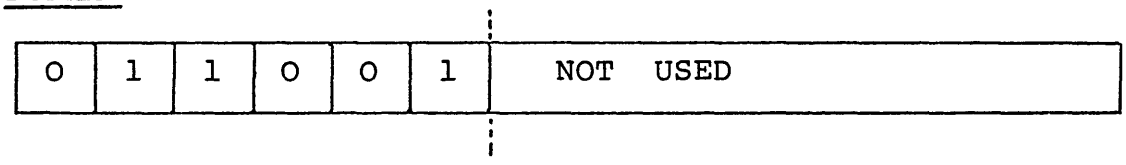
MICRO-CODE

```
145  LMI  FF1  JCR7    00100000   11110000   00000011   00110111
151  INAA FF1  JZR15   01111110   11110000   00000011   00101111
```

INSTRUCTION Shift the contents of the accumulator one place left and fill the L.S.B. with zero.

MNEMONIC SHL

FORMAT



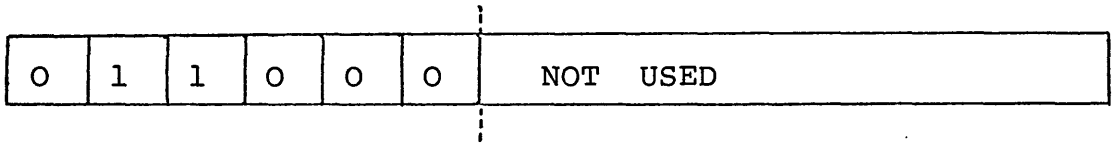
MICRO-CODE

152	LMI	FF1	JCR15	00100000	11110000	00000011	00111111
159	ALRA		JZR15	00011010	00000000	00000000	00101111

INSTRUCTION Shift the contents of the accumulator one place right and fill the M.S.B. with zero.

MNEMONIC SHR

FORMAT



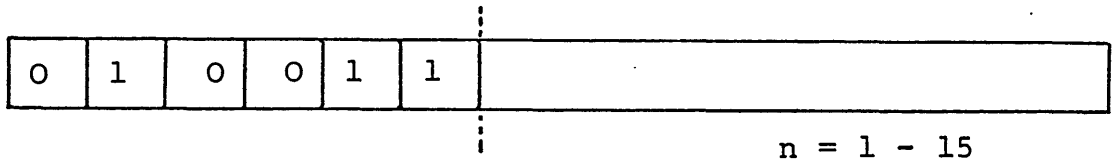
MICRO-CODE

153	LMI	FF1	JCR14	00100000	11110000	00000011	00111110
158	SRAA		JZR15	00011110	11110000	00000000	00101111

INSTRUCTION Shift the contents of the accumulator left n times, where n is specified by the 4 least significant bits of the address field.

MNEMONIC MSL

FORMAT



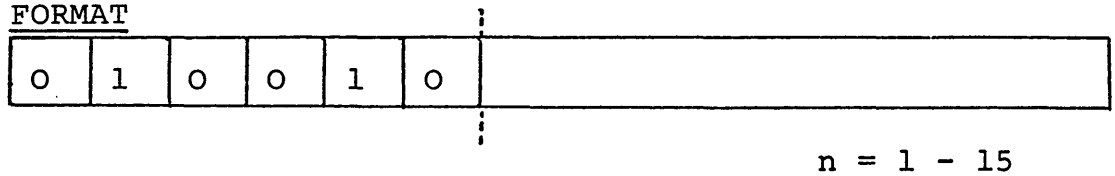
MICRO-CODE

176	LDMT JMPE JCR7	00110100	00000000	00100000	00110111
183	LMI FF1 JCR6	00100000	11110000	00000011	00110110
182	TZRT JCR4	10111000	00000000	11100000	00110100
180	ALRA JFL11	00011010	00000000	00000000	01001011
178	SDR9 FF1 CL LD	01010010	00001000	00000011	10000000
179	DCAT JCR6	00111100	00000000	00000000	00110110

INSTRUCTION Shift the contents of the accumulator right n times, where n is specified by the 4 least significant bits of the address field.

MNEMONIC MSR

FORMAT



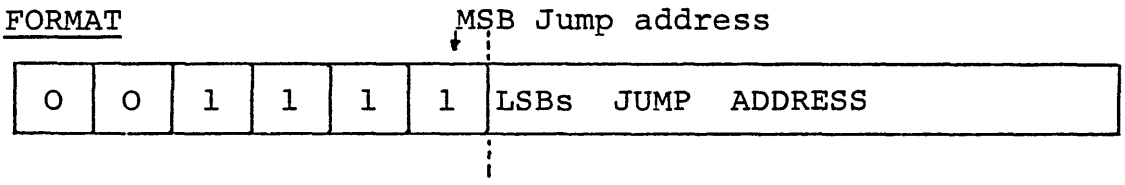
MICRO-CODE

177	LDMT JMPE JCR5	00110100	00000000	00100000	00110101
181	LMI FF1 JCC12	00100000	11110000	00000011	00001100
197	TZRT JCR4	10111000	00000000	00000000	00110100
196	SRAA JFL12	00011110	11110000	00000000	01001100
194	SDR9 FF1 CL LD	01010010	00001000	00000011	10000000
195	DCAT JCR5	00111100	00000000	00000000	00110101

INSTRUCTION Jump in high page i.e. fetch the next instruction from the location specified within the range 1024→2047.

MNEMONIC JHP

FORMAT



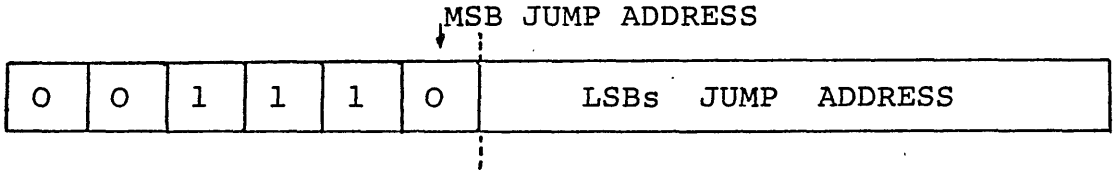
MICRO-CODE

192	LMMA	JMPE	FF1	JCR6	00110110	11110000	00100011	00110110
198	SDR		FF1	JCR7	01000000	00000000	00000011	00110111
199	ILR9			JZR14	00010010	11110000	00000000	00101110

INSTRUCTION Jump in low page. i.e. fetch next instruction from the location specified within the range 0 → 1023.

MNEMONIC JLP

FORMAT



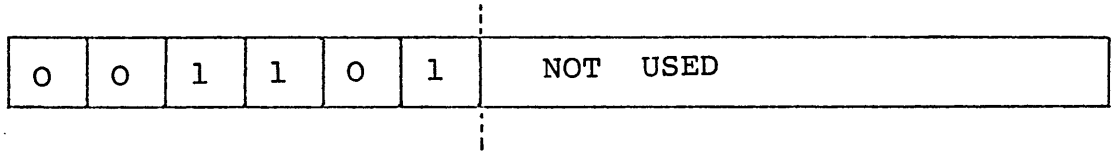
MICRO-CODE

193	LMMA	JMPE	FF1	JCR6	00110110	11110000	00100011	00110110
198	SDR		FF1	JCR7	01000000	00000000	00000011	00110111
199	ILR9			JZR14	00010010	11110000	00000000	00101110

INSTRUCTION Return from subroutine to instruction following subroutine call.

MNEMONIC RET

FORMAT



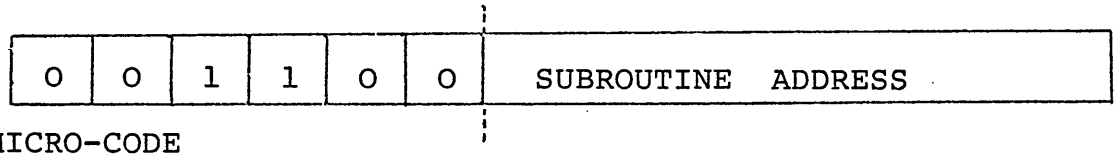
MICRO-CODE

200	LMI7	FF1	JCR15	00101110	11110000	00000011	00111111
207	ILR7		JCR14	00001110	11110000	00000000	00111110
206	SDR	FF1	JZR13	01000000	00000000	00000011	00101101

INSTRUCTION Jumpt to subroutine starting at the address specified.

MNEMONIC JSR

FORMAT



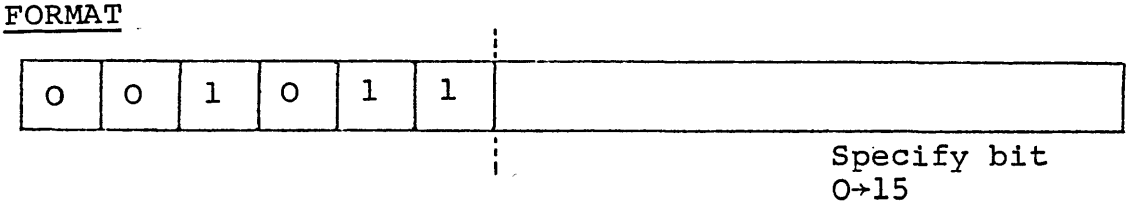
MICRO-CODE

201	ILR	JCR10		00000000	11110000	00000000	00111010
202	SDR7	FF1	JCR11	01001110	00000000	00000011	00111011
203	LMMA, JMPE	FF1	JCR12	00110110	11110000	00100011	00111100
204	SDR	FF1	JCR13	01000000	00000000	00000011	00111101
205	ILR		JZR14	00010010	11110000	00000000	00101110

INSTRUCTION Skip the following instruction if the accumulator bit, specified by the 4 LSBs of this instruction, is clear.

MNEMONIC SBC

FORMAT



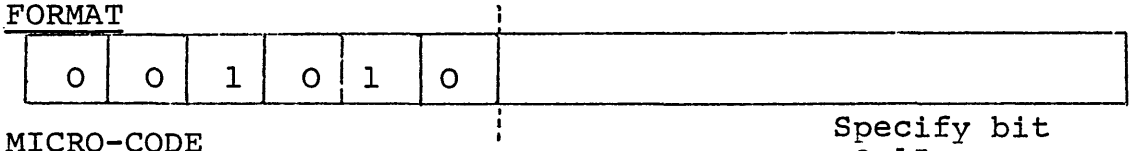
MICRO-CODE

208	LMI	FF1	EMM	JCR7	00100000	11110000	01010011	00110111
215	ANMA		EMM	JCR13	10010110	00000000	01010000	00111101
221	NOP		JFL13		11000000	11110000	00000000	01001101
218	LMI	FF1		JCR12	00100000	11110000	00000011	00111100
219	ILR9	CL	LD		00010010	11111000	00000000	10000000
220	ILR9			JZR15	00010010	11110000	00000000	00101111

INSTRUCTION Skip the following instruction if the accumulator bit specified by the 4 LSBs of this instruction is set.

MNEMONIC SBS

FORMAT



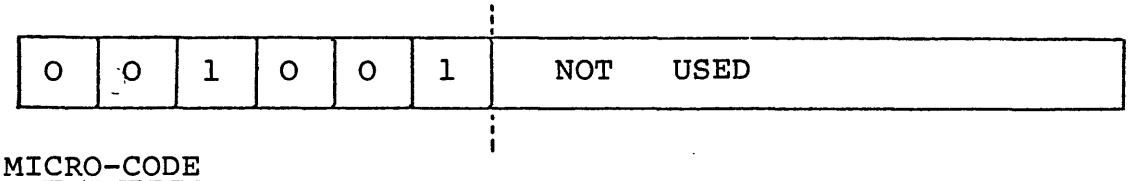
MICRO-CODE

209	LMI	FF1	EMM	JCR6	00100000	11110000	01010011	00110110
214	ANMA		EMM	JCR5	10010110	00000000	01010000	00110101
213	NOP		JFL13		11000000	11110000	00000000	01001101
210	ILR9		CL	LD	00010010	11111000	00000000	10000000
211	LMI	FF1	JCR4		00100000	11110000	00000011	00110100
212	ILR9		JZR15		00010010	11110000	00000000	00101111

INSTRUCTION Skip the following instruction if the contents of the accumulator are zero

MNEMONIC SZA

FORMAT



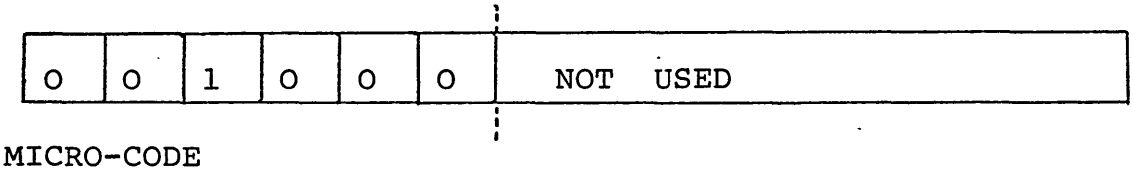
MICRO-CODE

```
216  LMI   FF1   JCR15   00100000  11110000  00000011  00111111
223  TZAA        JCR13   10111110  00000000  00000000  00111101
221  NOP         JFL13   11000000  11110000  00000000  01001101
218  LMI   FF1   JCR12   00100000  11110000  00000011  00111100
219  ILR9  CL    LD      00010010  11111000  00000000  10000000
220  ILR9        JZR15   00010010  11110000  00000000  00101111
```

INSTRUCTION Skip the following instruction if the contents of the accumulator are positive

MNEMONIC SPA

FORMAT



MICRO-CODE

```
217  LMI   FF1   JCR14       00100000  11110000  00000011  00111110
222  TZAA  KO111 JCR13       10111110  01110000  00000000  00111101
221  NOP   JFL13             11000000  11110000  00000000  01001101
218  LMI   FF1   JCR12       00100000  11110000  00000011  00111100
219  ILR9  CL    LD          00010010  11111000  00000000  10000000
220  ILR   JZR15             00010010  11110000  00000000  00101111
```

INSTRUCTION Skip the following instruction if the contents of the accumulator differ from the contents of the data/constant store location as specified.

MNEMONIC SAD

FORMAT

0	0	0	1	1	0	SUB ADDRESS

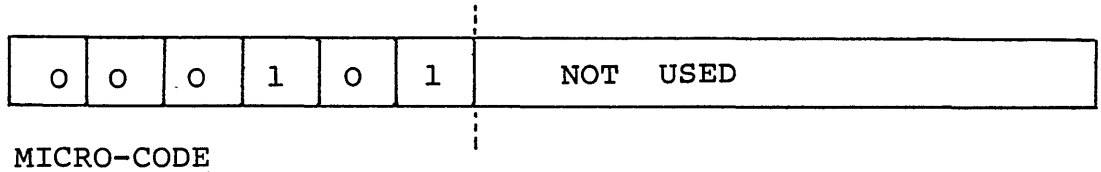
MICRO-CODE

225	LMI FF1 REN JCR2	00100000	11110000	00110011	00110010
226	NOP REN JCR3	11000000	11110000	00110000	00110011
227	XNMA REN JCR4	11110110	00000000	00110000	00110100
228	CMAA JCR5	11111110	11110000	00000000	00110101
229	TZAA JCR15	10111110	00000000	00000000	00111111
239	NOP JFL14	11000000	11110000	00000000	01001110
234	ILR9 CL LD	00010010	11111000	00000000	10000000
235	LMI FF1 JCR12	00100000	11110000	00000011	00111100
236	ILR9 JZR15	00010010	11110000	00000000	00101111

INSTRUCTION Skip the following instruction if the converter data is positive i.e. sign bit = 0.

MNEMONIC SPD

FORMAT



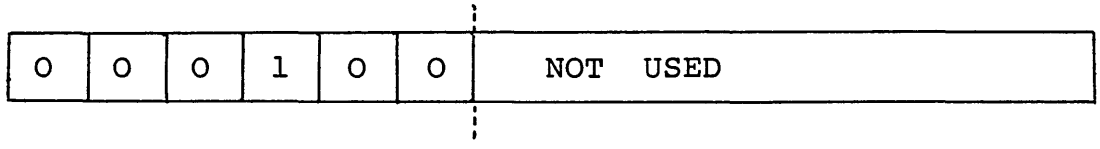
MICRO-CODE

232	LMI FF1 JCR13	00100000	11110000	00000011	00111101
237	TZAA K1101 JCC13	10111110	11010000	00000000	00001101
221	NOP JFL13	11000000	11110000	00000000	01001101
218	LMI FF1 JCR12	00100000	11110000	00000011	00111100
219	ILR9 CL LD	00010010	11111000	00000000	10000000
220	ILR9 JZR15	00010010	11110000	00000000	00101111

INSTRUCTION Skip the following instruction if the converter data is negative i.e. sign bit = 1.

MNEMONIC SND

FORMAT



MICRO-CODE

233	LMI FF1 JCR14	00100000	11110000	00000011	00111110
238	TZAA K1101 JCR15	10111110	11010000	00000000	00111111
239	NOP JFL14	11000000	11110000	00000000	01001110
234	ILR9 CL LD	00010010	11110000	00000000	10000000
235	LMI FF1 JCR12	00010000	11110000	00000011	00111100
236	ILR9 JZR15	00010010	11110000	00000000	00101111

INSTRUCTION Skip the following instruction if the contents of the accumulator are negative.

MNEMONIC S N A

FORMAT

0	0	0	1	1	1	NOT USED

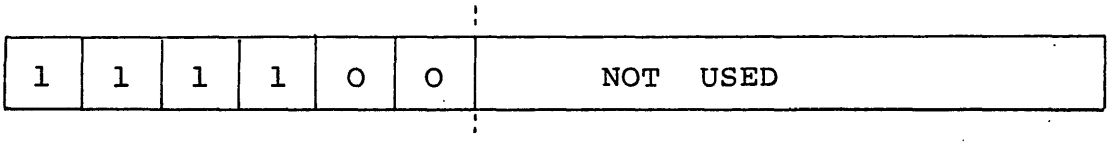
MICRO-CODE

224	LMI FF1	JCR7	00100000	11110000	00000011	00110111
231	TZAA KO111	JCR15	10111110	01110000	00000000	00111111
239	NOP	JFL14	11000000	11110000	00000000	01001110
234	ILR9 CL LD		00010010	11111000	00000000	10000000
235	LMI FF1	JCR12	00100000	11110000	00000011	00111100
236	ILR9	JZR15	00010010	11110000	00000000	00101111

INSTRUCTION No operation.

MNEMONIC NOP

FORMAT



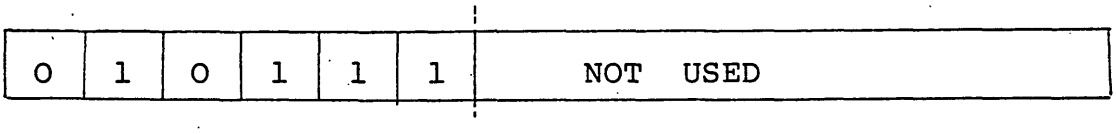
MICRO-CODE

9	LMI FF1	JZR10	00100000	11110000	00000011	00101010
10	NOP	JZR15	11000000	11110000	00000000	00101111

INSTRUCTION Decrement the loop counter and skip the following instruction if the result is zero. (count skip loop).

MNEMONIC C S L

FORMAT



MICRO-CODE

160	DSM8	JCR4	00110000	00000000	00000000	00110100
164	TZR8	JCR5	10110000	00000000	00000000	00110101
165	LMI FF1	JFL10	00100000	11110000	00000011	01001010
162	LMI FF1	JCR3	00100000	11110000	00000011	00110011
163	NOP	JZR15	11000000	11110000	00000000	00101111

INSTRUCTION — Multiply the contents of the accumulator by the contents of the data/constant store location specified. The signed 32 bit product is in Accumulator (MSB's) and Register T (LSB's).

MNEMONIC — MUL

FORMAT

1	0	0	0	1	1	SUB ADDRESS

MICRO-CODE

```
112    TZAA   KO111   JCC16
256    ILR9   REN     JFLO
259    CIA A FF1  REN JCR2
258    SDRT   FF1  REN JCR1
257    LTMA KO111  REN  JCC19
305    ACMA   REN  JFL3
307    SDR2   FF1  JCR7
311    CIAA   FF1  EDB  JCR4
308    NOP  EDB  WEN  JCR5
309    NOP  EDB  WEN  JCRO
306    SDR2   FF1  JCRO
304    CSRA   JCR6
310    ALRA   JCC16
262    ALRA   JCR5
261    ALRA   JCR4
260    ALRA   JCR7
263    SDR1   FF1  EDB  JCR8
264    CLAA   STZ  JCR9
265    SRAT   REN  HCZ  JCR10
```

MUL (Contd.)

```
266    INR1  FF1  REN  STC  JFL 3
315    AMAA  REN  HCZ  JCR10
314    SRAA   REN  STZ JCR9
313    SRAT  FFZ  REN  HCZ  JCFO
267    SRAA  STZ  JCR12
268    SRAT  FFZ  HCZ  JCR13
269    SDR6  FF1  EDB  JCR14
270    ILR2  EDB  JCR15
271    NOP  EDB  WEN  JCC23
383    LMI  FF1  EDB  WEN  JCR14
382    ALR9  KO111 EDB  JCR13
381    TZAA  KO111  EDB  JCR12
380    ILR6  EDB  JFL5
347    CIAT  FF1  STZ  JCR12
350    CIAA  FFZ  EDB  JCR10
346    SDR9  FF1  CL  LD
```

ROM CONTENTS.

```
112    10111110    01110000    00000000    00010000
256    00010010    11110000    00110000    01000000
259    00111110    11110000    00110011    00110010
258    01011000    00000000    00110011    00110001
257    10110110    01110000    00110000    00010011
305    00010110    11110000    00110000    01000011
307    01000100    00000000    00000011    00110111
311    00111110    11110000    01000011    00110100
308    11000000    11110000    11000000    00110101
309    11000000    11110000    11000000    00110000
306    01000100    00000000    00000011    00110000
```

ROM CONTENTS (Contd.)

304	01011010	11110000	00000000	00110110
310	00011010	00000000	00000000	00010000
262	00011010	00000000	00000000	00110101
261	00011010	00000000	00000000	00110100
260	00011010	00000000	00000011	00110111
263	01000010	00000000	01000011	00111000
264	10010110	11110000	00001000	00111001
265	00011100	11110000	00111100	00111010
266	01100010	11110000	00110111	01000011
315	00010110	00000000	00111100	00111010
314	00011110	11110000	00111000	00111001
313	00011100	11110000	00111101	01010000
267	00011110	11110000	00001000	00111100
268	00011100	11110000	00001101	00111101
269	01001100	00000000	01000011	00111110
270	00000100	11110000	01000000	00111111
271	11000000	11110000	11000000	00010111
383	00100000	11110000	11000011	00111110
382	00010010	01110000	01000000	00111101
381	10111110	01110000	01000000	00111100
380	00001100	11110000	01000000	01000101
347	00111100	11110000	00001011	00111100
350	00111110	11110000	01000001	00111010
346	01010010	00001000	00000011	10000000

DIGITAL PROCESSORS FOR SUBSTATION SWITCHING AND CONTROL

E. Horne and B.J. Cory

1. Introduction.

As digital processors increase in reliability and decrease in cost, their use within the substation environment for control and protection purposes will become commonplace. To ensure optimum performance of high voltage systems under fault and heavy loading conditions a more sophisticated controller will be necessary, especially to limit fault damage and to organise reswitching for security and maintenance purposes. Already processors for on-line data acquisitions, load flow calculations, network configuration and security checks etc. are becoming standard and the extension of similar techniques to waveform monitoring, fault measurement and circuit switching are now feasible.

This paper describes one possible philosophy of applying mini-computers to the collection of data and to the control of a substation. This scheme is being investigated on a laboratory model. Some algorithms for fault measurement using the digital system described here are presented in a companion paper.

2. General outline of the model system.

For research and development purposes, the requirements for data acquisition and control of a typical 400 kV mesh substation on the CEGB's network has been chosen. As indicated in fig. 1, one corner of such a station has been modelled by setting up 220 v, 3 phase circuits in the laboratory in which the circuit breakers and isolators have been simulated by 3 phase contactors. These are slugged electronically to provide realistic operating times and the control circuitry enables them to be operated from a digital processor. In this mesh corner, full instrumentation is applied by means of voltage and current transducers, but in the remainder of the substation only the status of breakers and isolators is modelled by single phase relays so that the station configuration can be determined.

The digital system connected to the model is shown in fig. 2 and consists of two processors having a common data base (DB). The data acquisition processor (DAP) is used to scan both analogue (transducer) signals and digital (c.b. position) signals from the model, to check that both forms of data are compatible and to insert the data in conditioned form into the DB.

The control and protection processor (CPP) is responsible for running all the algorithms which determine the state of the substation and surrounding

E. Horne and B.J. Cory are in the Department of Electrical Engineering, Imperial College, London SW7 2BT.

system, to switch it to eliminate faults or after a manual command from the local control room and generally to provide communication with control centres. In the arrangement described here four DAP's and four CPP's would be required for a complete mesh substation with interlinking of both data and control channels so that a suitable measure of redundancy is achieved for adequate reliability.

3. System Hardware.

In our system, the DAP is a NOVA 1210 mini-computer having a 4 k word core store and a 1.35 µs cycle time. An interface has been specially constructed to provide 4-16 bit digital word inputs for breaker status data and 48 multiplexed analogue inputs for current and voltage measurement. Not all these are needed for the studies to be described.

Each analogue output from the model is connected to a sample/hold circuit controlled by a single phase locked clock synchronised to the 50 Hz supply. A clock pulse causes all sample/hold circuits to sample simultaneously with a window width of 1 µs. The sampling rate can be varied manually or by computer control at 32, 24, 16, 12, 8, 4 etc. samples per cycle and an integrator is included in the sample/hold construction to give a measure of filtering.

The analogue/digital conversion is performed by 3-10 bit, 25 µs converters within the interface, thus enabling all 48 inputs to be converted and input in less than 500 µs. In practice, a maximum of 42 inputs are used for system data, the others being provided for check/calibration purposes.

The interface has been specifically designed to minimise data acquisition overheads within the computer to allow maximum time for data validation procedures, so that analogue and switch position data can be checked for consistency. At 32 samples per cycle, the DAP must collect, validate and transfer data to the DB in less than 625 µs.

The DAP is linked to the CPP (a PDP15 with 16 k store in our case) via a first-in, first out (FIFO) buffer. Data output to this link is under NOVA program control and on completion of a data block the link generates a data channel request to the CPP and the block is transferred to the DB which is a portion of the PDP15 core store. All protection and control algorithms are run within the CPP using the DB which normally stores two complete 50 Hz cycles of data before being overwritten. The model is controlled from the CPP by means of an 18 bit output word enabling switching commands to be issued at a maximum rate of one per microsecond, if necessary.

4. System Software.

Apart from the protection algorithms described in a companion paper, efforts have been concentrated on devising data validation and auxiliary control routines. Some of these are briefly described below.

4.1 Data validation in DAP. Since the link between the CCP and the system is through the DAP, it is important that data collected via the interface is correct and reliable. For system protection the speed of acquisition and validation must be compatible with desired fault

protection times. In the concept discussed here, a time of 625 μs is available for reading in data from switch positions and sampling circuits, validating it and forming it into 16 bit words for the DB transfer. Consequently, two principles were established:

(i) the method must be simple; there is no time to recalculate missing or erroneous data.

(ii) the algorithm must be flexible enough to cope with system configuration or transducer position changes with the minimum of modification.

After considering various alternatives, the method adopted reads switch position data as two bits which are logic complements of each other. A simple complement instruction in the NOVA is used to check this and if an error is detected the switch data is given a logic 1 (tag) to indicate an error. This data is packed into 16 bit words each containing 6 switch positions and 4 error bits.

Analogue data is read in from an A/D convertor as a 10 bit word and checked by hardware to see if it is zero within a tolerance of 2 bits. A zero value cannot be checked since this may indicate that the sample has been taken when the waveform passes through zero, that the circuit is de-energised or that an interface fault exists. Consequently, zero data must be accepted unvalidated. If the data is non-zero, the status of the neighbouring switch(es) is examined. If closed, the data is assumed to be consistent, otherwise the data is tagged as doubtful. It is the job of the CPP to choose either to disregard tagged data or to replace it by correct data calculated from other measurements.

For a given substation configuration, a condition list of data and switches is stored in successive locations of the NOVA memory but, to avoid delays, no interrupts or subroutines are used. Each A/D convertor output is scanned at predetermined intervals in strict sequence and the success of the method depends upon following a programmed cycle of events each time a clock pulse initiates a sample/hold operation. At slower rates than 32 samples per cycle, more time is available for validation so that an estimation scheme could be appropriate. Possible estimation methods for substation data collection are now being evaluated.

4.2 Control and switching routines in CPP. Besides running algorithms for the determination of busbar, line and transformer faults the CPP is used for data recording, alarm analysis, interlocking and control functions, all obtained by appropriate manipulation of data in the DB. It is advisable to hold all algorithms in direct memory to avoid any problems likely to occur with disc stores.

Previous papers[2,3,4] have shown how most control functions can be implemented in on-line processors and only the operations of synchronisation and transformer tap-changing, which have not previously appeared in the literature, will briefly be considered here.

For check-synchronisation it is necessary to determine the phase-angle or voltage difference between the peaks of the two voltage waveforms. Consequently, one of the published methods for peak determination can be employed based on a number of sampled values. In practice the method

given by Mann & Morrison[5] was found most suitable using three consecutive samples to find

$$\frac{v}{v'} = \frac{\tan \omega t}{\omega}$$

where v is the voltage at any point on the waveform and v' its calculated slope. For two voltage waveforms v_1 and v_2 with phase displacement ϕ we have

$$\omega t = \arctan \frac{\omega v_1}{v_1'},$$

$$\text{and } (\omega t + \phi) = \arctan \frac{\omega v_2}{v_2'},$$

$$\text{hence } \phi = \arctan \left(\frac{\omega v_2}{v_2'}\right) - \arctan \left(\frac{\omega v_1}{v_1'}\right). \quad \ldots\ldots\ldots \quad (1)$$

Tests showed that this method employing a 'look-up' table of arctangents at 2^0 intervals was sufficiently accurate for most purposes. It was superior to both the arcsine method and the counting of samples between peaks method. However, a measurement of ω is required from a sampled frequency meter. The program as developed checked that the difference between the peak value of the voltages was within a given tolerance and, if so, it proceeded to calculate the phase angle. The phase difference must be less than a given value and stay there for at least 0.1s (obtained by at least 3 checks at predetermined intervals) before a satisfactory check-synchronising signal can be given. If desired, a combination of voltage difference and phase angle limits can be employed for special situations.

For tap-change control, more time is available for voltage peak determination and an algorithm based on a Fourier technique[1] is advised. In the method tested on a PDP8 computer, a clock is employed to initiate the program every second in which samples from the previous half cycle of voltage are used to calculate the peak value, V_p. The deviation from the setting value $\Delta V = (V_{setting} - V_p)$ is then calculated. If $|\Delta V| > \pm 0.2|V_p|$ an alarm is given, otherwise the program checks for $\Delta V \geq \pm 0.03|V_p|$ If this is so, the partial sum $\lesseqgtr \Delta V_i.t$ is stored, where t is the time interval between checks (1s in this case). If this partial sum is greater than a predefined tolerance (depending upon response time required) a LOWER or RAISE tap-change signal is outputted, depending upon the sign of $\lesseqgtr \Delta V_i.t$. No action is taken until the partial sum reaches the tolerance value. If ΔV is less than the $\pm 0.03|V_p|$ tolerance, nothing is added to the partial sum. This program produced similar tap-change operations to that of a conventional AVC relay.

4.3 Overall CPP software. At the time of writing, the separate control and protection algorithms have not been integrated within the PDP15 processor. It is anticipated that continuous monitoring of substation voltage will be used to initiate the protection algorithms which will be given priority when a disturbance is detected. Since all data is collected independently of the CPP, algorithms for buszone check, transformer

protection and line fault impedance measurement can be run sequentially within 5 ms since the major delay is generally in awaiting up-to-date samples to arrive. Lower priority will be given to reswitching sequences, alarm analysis and data logging. Most primary protection algorithms employ a 2 out of 3 check before a trip signal is initiated and back-up protection can be obtained by requesting data from adjacent corners of the mesh.

With the present known reliability of mini-processors having programmable core stores, adequate security can only be achieved by duplication of all processors and data bases even though some measure of redundancy is obtained by interlinking mesh corner CPP's. Although such duplication looks expensive processors costing around £2,000 are now available suited to this task but the CPP is likely to require at least 12k of store, thus increasing the cost. With the advent of inexpensive and reliable micro-processors, however, many of the tasks performed by the CPP could be paralleled by dedicated hardware, leading to higher speed and security of operation. Development along these lines is continuing.

5. Conclusions.

Whilst the sub-station model and data acquisition system described previously provides a flexible and valuable research tool on which validation and protection algorithms can be developed, several shortcomings are apparent which could detract from the practicality of an on-line sub-station installation.

The mini-computers used are typical of the present generation, having cycle times of the order of 1.2μs, a factor closely related to the performance of the machine core store. From experience to date these times impose some restriction on data validation capabilities, and hence on the system reliability, in the case of the DAP, and on fault detection speeds in the case of the CPP. The latter, of course, becomes more significant as circuit breaker operation times decrease. Additionally both machines possess many functions not required for the task in hand and these unwanted facilities are reflected in the cost of the overall system.

A solution to these problems could be found in the use of very fast purpose built microprocesses (this term should not be confused with the currently available microprocessor families which at the present time do not better the speed of the mini-computer).

Semiconductor technology has made available integrated circuit high speed arithmetic logic units and RAM/ROM memories of increasing size but falling cost. It is envisaged that a dedicated digital relay will consist of a unified central processor with program control supplied by plug-in semi-conductor memories, thus enabling the relay to perform any line, bus-bar, or transformer protection task in a similar manner to its electromechanical counterpart. A major advantage of this technique is the ability of each relay in a substation to communicate its operational data to a central co-ordinating and event recording mini-computer.

The most useful and desirable co-ordinating functions are yet to be determined.

6. Acknowledgements.

The authors are grateful to the Science Research Council for their financial support for this work and to Dr. S. Miniesy and Mrs. Y. Stevens for help in the design of hardware and program development.

7. References.

1. Ranjbar A.M. & Cory B.J. : Algorithms for Distance Protection. Companion paper IEE Conference "Developments in Protection" 1975.

2. Dy Liacco T. & Kraynak T.J. : Processing logic programming of circuit breaker and protection relaying information. IEEE PAS 88 Feb 1969. pp. 171-5.

3. Cory B.J. & Moont J.F. Application of digital computers to busbar protection. IEE Conference Bournemouth, 1970.

4. Row C.H.T. & Cory B.J. Substation interlocking and switching with a small digital computer. IEE Conference : Computers in Power System Operation and Control, Bournemouth 1972.

5. Mann B.J. & Morrison I.F. : Digital calculation of impedance for transmission line protection. IEEE PAS-90 Jan/Feb 1971.

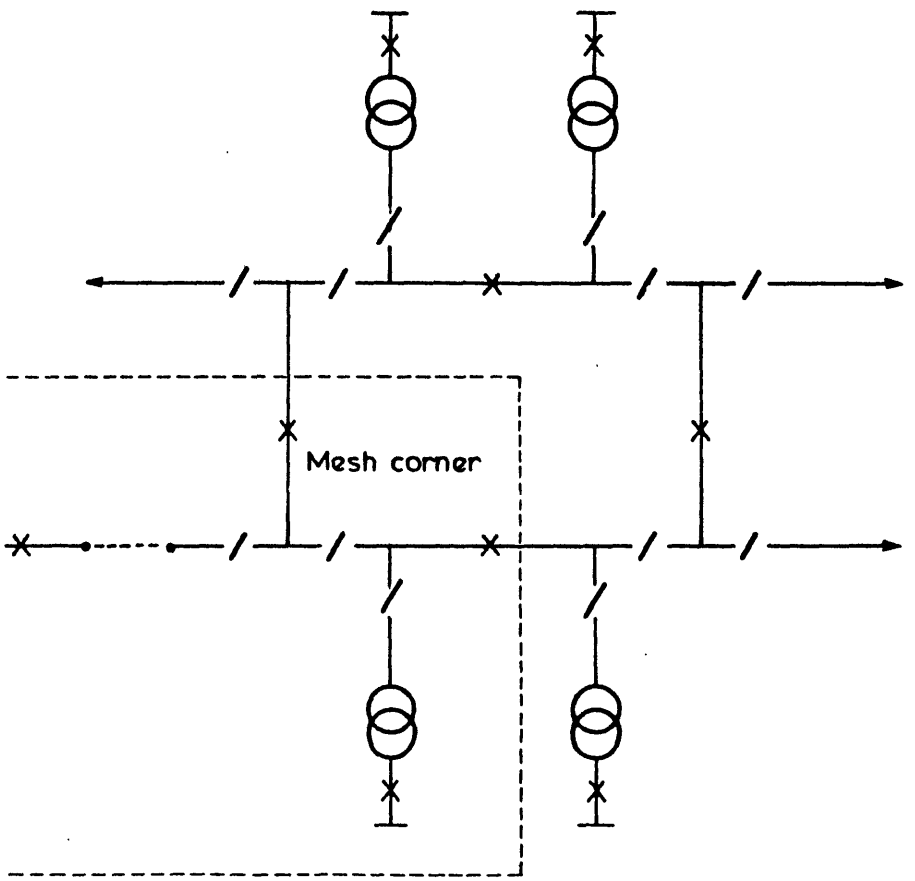


Fig. 1. Mesh type substation layout

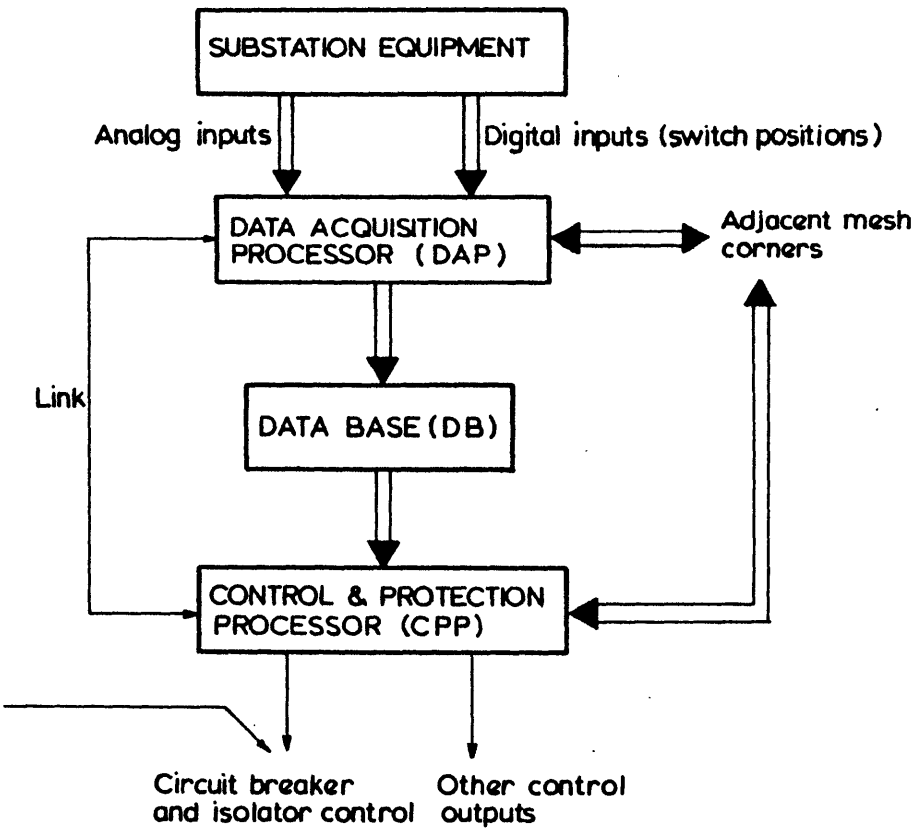


Fig. 2. Proposed hardware configuration for mesh corner protection and control.

Micro-computer applications in protection relaying.

L. Petrou, E. Horne, B.J. Cory.
Imperial College, London SW7.

1.1 Introduction

The application of digital computing techniques to the task of power system protection has been a topic of research by power utilities, industry and universities for several years. The advent of the relatively inexpensive "mini" computer provided a great impetus to studies in this field, and the majority of schemes which have been proposed or implemented are based on processors of this type.

Work undertaken at Imperial College using a multi-mini computer system (Fig 1) for protection and control of one corner in a mesh type substation has demonstrated that schemes of this nature are subject to inter-related economic and technical constraints which may prove unacceptable for widescale application.

1.2 Disadvantages of mini-computer protection

Inevitably in any type of main protection scheme a duplication or triplication of the relaying equipments involved is necessary if the stringent reliability criteria of the task are to be satisfied. From an economic viewpoint, therefore, a multi-minicomputer solution will represent a substantial investment, despite the reductions in cost which have been achieved for these machines. As a result, the tendency, evident in literature published by research groups, has been to develop the concept of integrated protection methods in which multiple relaying functions are embodied in a single computer. Whilst this integrated approach ensures maximum utilisation of the installed computing power, it also gives rise to several intractable technical difficulties. The major problems are summarized as:-

a) Complex Software

The software required for the integrated protection system may be classified into two distinct groups, (i) the algorithms capable of performing the required protective functions, and (ii) the "operating system" for these routines. As the extent of integration increases, the complexity of the operating system software also increases and its operational reliability deteriorates.

b) Initalisation

The initalisation of protective routines is basically a function of the operating system mentioned above. In this context the correct identification of the appropriate plant protection routine at the onset of a fault condition entails a difficult programming task.

c) Response time

Due to the sequential nature of the digital computer the time consumed by the organisational software during a fault may well result in a degredation of clearance speeds, particularly in cases of incorrect initalisation selection. This factor obviously assumes increasing importance as the transmission system load approaches its maximum.

d) Reliability

A high level of hardware and software reliability is

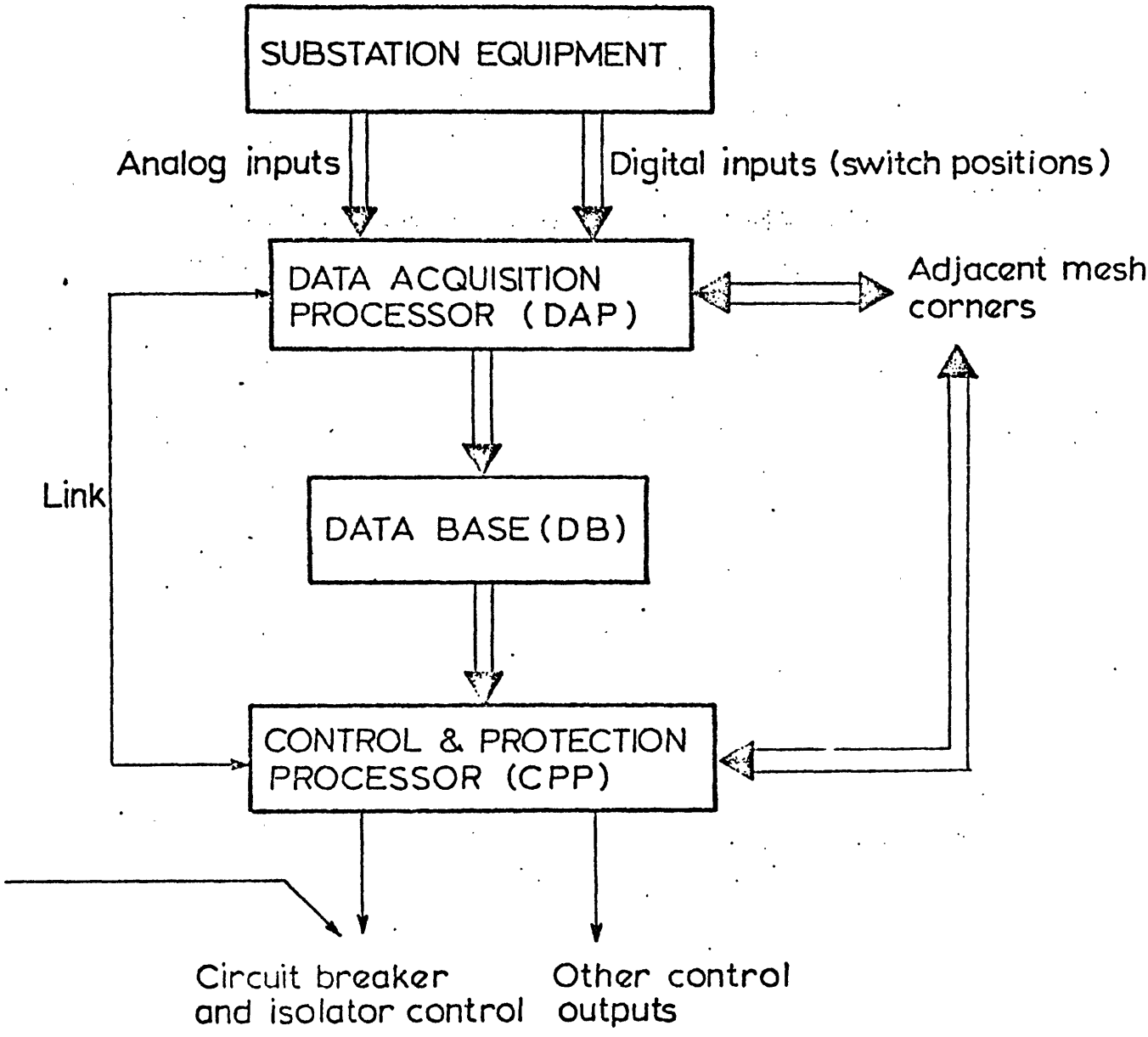


Fig. 1 Proposed hardware configuration for mesh corner protection and control.

required in the integrated system since any major failure in either area will affect a substantial portion of the protection facilities. The provision of high reliability systems would impose further economic penalties.

From the foregoing it is seen that the major limitations of the mini-computer oriented scheme largely originate as a consequence of the integration of the protection processes.

2.1 The Microcomputer dedicated relay

To overcome the difficulties outlined above, an alternative approach has been adopted which, in common with current electromechanical and static analogue relaying practice, proposes a dedicated parallel unit organisation.

The rapid advance of semiconductor technology has made economically possible the employment of a dedicated digital relay for each task, based upon low cost computing devices which fall within the general term "microprocessor". A single relay will consist of two component sub-systems broadly similar to the mini computer configuration, these being the data acquisition or measurement sub-system, and the protection computation processor. Irrespective of the relaying task to be performed, each unit can comprise common hardware but allows for considerable flexibility in use.

2.2 The Measurement sub-system

The function of this sub-unit is primarily one of data collection. This requires that at intervals the a.c. quantities required for the protection algorithm are simultaneously sampled and held at the appropriate transducer outputs and the sampled values are then multiplexed from the sample-holds to an analogue-digital converter, as shown in Fig 2. The resulting digital values are loaded into a first-in-first-out buffer store with breaker/isolator status information which may be required to await the demand of the protection processor.

The timing and configuration control of these elements is provided by an 8 bit micropocessor of the MOS type. Although of low speed, the 8 bit devices which are available are more than adequate for the requirements of the unit, and enable a simple stored program controller to be implemented with a minimum number of integrated circuits whilst providing a flexibility of data acquisition which would not be available in a hard wired arrangement. It should, however, be noted that in this scheme the primary multi-bit data paths are independent of the control processor and thus no provision is made for data pre-processing.

2.3 The Protection Processor

The task of the second portion of the relay differs widely from that of the processor application in the measurement unit. For each block of sampled data provided by the measurement unit, the protection processor must solve the particular fault detection algorithms to which it is dedicated. In this respect the protection algorithm is regarded as including any data filtering which may be required, the fault detection routines, and output switching commands.

An emphasis must thus be placed in this unit upon the ability to perform logical and arithmetic operations, and in order to satisfy these requirements several factors must be considered in

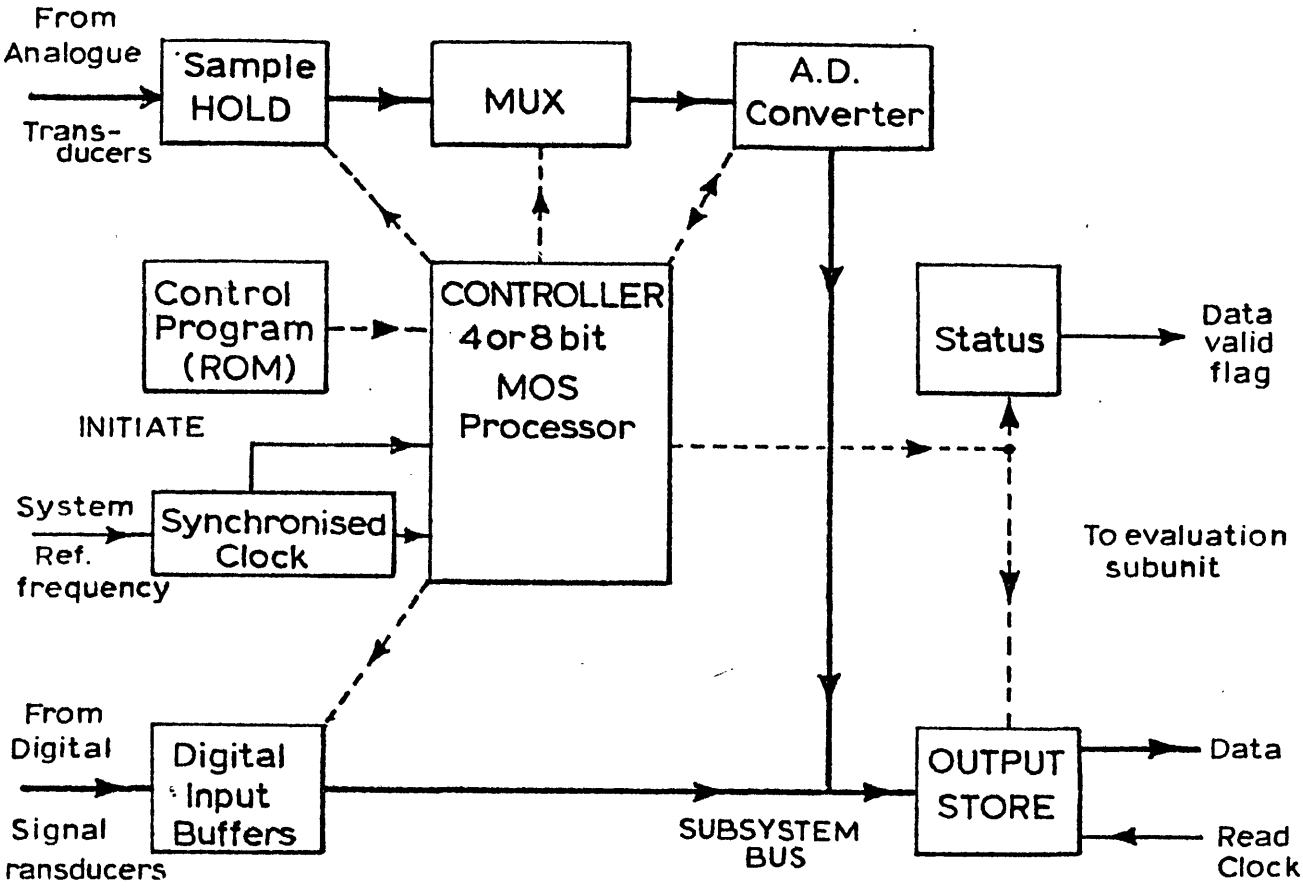


Fig. 2 The Measurement Subsystem

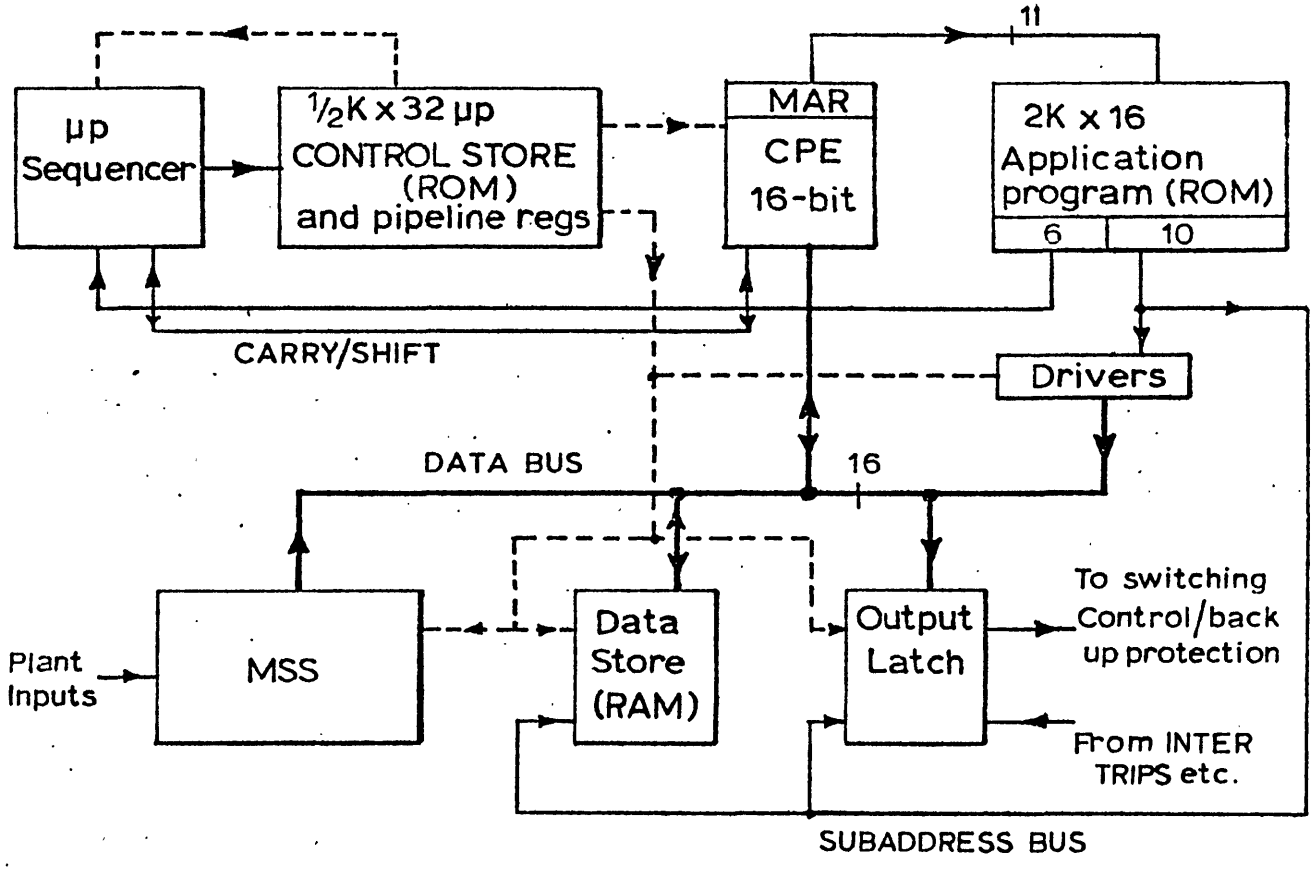


Fig. 3 The Protection Processor

the choice of processor. The most important of these are outlined below, and they indicate that the bipolar bit-slice family of micro-processors can provide a satisfactory solution.

a) Word length

For arithmetic operations employing an 8 bit word length consistent with that commonly found in the single chip microcomputer great care must be taken to avoid a serious degredation of accuracy due to rounding and truncation errors. The bit slice devices allow a great freedom of word length in 2 or 4 bit multiples and the processor shown in Fig 3 has a 16 bit structure.

b) Operating speed

Previous work has shown that in order to achieve a complete evaluation of the a.c. system data between consecutive samples (i.e. 2.5 ms), an instruction cycle time of mini computer standard (i.e. 1-2μS) is desirable. Once again this requirement is satisfied by a bipolar design, whereas with existing M.O.S. devices these speeds are not realizable, particularly if multiple byte operations are involved to maintain accuracy, or to provide a usable instruction format.

c) Powerful Software and Architecture

Although the bit-slice devices are more primitive basic components than their single chip counterparts, they do thereby allow the overall processor architecture to be closely matched to its proposed function. The system of Fig 3 bears only a superficial resemblance to a conventional computer architecture providing as it does only limited read-write storage which is separated for address purposes from the main program memory. This structure combined with extensive pipelining in the control paths and a modified form of Direct Memory Access for input data from the measurement sub-system serves to optimise the unit performance.

Finally from a software viewpoint the micro-programmable organisation of the processor allows a highly efficient dedicated macro-language to be developed for the description of fault detection algorithms.

3. Conclusions

It is well known that in order to achieve optimum results from micro-computers they should be used to perform limited and closely defined tasks. Extension to multiple task or general purpose use negates many of the valuable attributes which the devices possess, and a more parallel structure is desirable (i.e. multi-processor) beyond a certain task complexity. This implies that for power system protection use they are ideally employed as dedicated single function first and second main relays, and the major advantages which may be obtained from such an implementation are:-

a) Standardisation

Since the component sub-units of a relay are constructed with standard hardware features, individual relays for specific functions can be assembled by inclusion of appropriate control algorithms. A considerable easing of the difficulties encountered when an equipment manufacturer is confronted with special requirements, which currently require design changes in conventional electromechanical or state relays, is thus achieved. For the

relay user, standardisation of many items provides a considerable simplification of testing and maintenance operations, and reduces the burden of personnel training.

b) Flexibility

The micro-computer based relay offers the great adaptability which characterises all stored program digital machines. Alterations to the power system plant configuration or parameters may thus be easily reflected in corresponding relaying modifications avoiding protracted and costly re-wiring or re-setting.

c) Reliability

Although the basic components relibilities within a micro-computer relay are only comparable with those used, for example, in a static analogue relay, the dynamic nature of the device enables many of the well proven error detection and self test routines to be incorporated in the operating programs. Thus the possibility of a coincident system/relay fault is minimised and the overall protective equipment reliability enhanced.

d) Compatibility

As a final consideration it is possible to examine the role of the dedicated digital relay within a hierarchical sub-station control scheme. It is envisaged that the microcomputer front line equipment would be provided with a back-up system incorporated in a mini-computer based integrated protection system. The software limitations of the integrated solution mentioned in Section 1.2 would not be critical in a back up mode, and the sophisticated mini machine would provide a constant monitoring service for the front line relays, in addition to communication with controlling stations. For such an implementation the value of a uniform digital device structure is obvious and the micro-computer relay reduces the problems of equipment compatibility.

Currently, effort at Imperial College is directed towards the construction of a protection relay based on the principles outlined in this paper, and it is hoped that the performance of the equipment will be evaluated for several typical relaying tasks in the near future.

References

Micro-processors:

1) M. Healey: Minicomputers and Microprocessors. Hodder and Stoughton. 1976.

2) B. Francis: Microprocessors; the minicomputer/random logic alternative ? Electronic Engineering. March, 1975.

3) Justin Rattner: Bipolar LS1 computing elements usher in new era of digital design. (Electronics Sep 5, 74)

4) Uzunoglu Vasil: Analysis and design of digital systems. Gordon and Breach 1975.

Digital Protection:

5) B.J. Cory, G. Dromey, B.E. Murray: Digital system for protection. CIGRÈ 1976. Paper 34 - 08.

6) Murray, B.E., Dromey, G. : "Practical design considerations affecting the use of digital computers in substations." (4th IFAC/IFIP Conference on digital Computer Applications to Process Control, Zurich, March 1974, page 292)

A MICROPROCESSOR POWER SYSTEM PROTECTION RELAY

L. Petrou, E. Horne, B.J. Cory
Imperial College
London SW7 U.K.

ABSTRACT

First a brief discussion is given outlining the reasons for developing the digital protection scheme described. Next the design of a dedicated digital protection relay is considered, from the hardware and software points of view.

In the last chapter some applications of the relay for different protection schemes are examined.

1. INTRODUCTION

Digital computers are already well established in several areas of power system control. They are widely employed in network control and dispatch centres and in power stations.

An extension of digital techniques in power system protection, is an obvious area of development. Digital protection offers improved accuracy, enhanced reliability, better protection characteristic matching, greater flexibility and modular construction. However it introduces some disadvantages, for example, the need to convert the analogue measurements of the system to digital quantities, interference problems and difficulty in completely testing the system under all possible operating conditions.

The first applications of digital protection, used a minicomputer as the processing device. However the use of minicomputers imposes inter-related economic and technical constraints upon the schemes proposed, most of which are encountered when a scheme of this type was constructed at Imperial College. For reliability a duplication or triplication of the relaying equipment used in a multi-minicomputer scheme is required, which represents a considerable cost. As a compromise of this, "integrated protection" schemes have been proposed, in which multiple protection functions are embodied in a single computer to maximize usage of the installed computer power. It has been found that the integrated scheme suffers from several difficulties, such as complex software, critical initialisation, degredation of clearance speeds and reliability[3]. An alternative approach[2] is to replace the existing electromechanical or static analogue relays by dedicated digital relays constructed with microprocessors. Back-up protection and monitoring of the dedicated relays are provided by a mini or microcomputer. Our paper concentrates on the design of a microprocessor based relay.

2. THE MICROCOMPUTER DEDICATED RELAY

In each relay the data acquisition and processing functions are separated. Thus two units comprise a relay, they are: i) the data acquisition interface and ii) the protection processor. This division of the relay components, is dictated by the differing nature of the two functions.

2.1 The data acquisition interface

This unit is concerned with data collection, and has been designed to fulfill the requirements of many different protection schemes. It is a self-contained unit, controlled by a programmable device. The interface can handle up to 16 analogue channels and 4 digital channels. A block diagram of the unit is shown in "Fig. 1". The analogue signals are the a.c. quantities of the system and the digital inputs determine circuit breaker (CB) and isolator status. The analogue signals are derived from the power system by the configuration shown in "Fig. 2". These signals are filtered by second-order Butterworth anti-aliasing filters with a cutoff frequency dependent on the sampling rate. The analogue signals are sampled simultaneously and held on the corresponding sample-hold devices. They are then multiplexed to an analogue-digital converter (ADC). The A/D conversion takes 6μs and has a resolution of 10 bits. The resulting digital values are stored in an output buffer, together with the CB/isolator status (digital inputs). A data ready flag is generated, which indicates to the protection processor that data have been prepared. The timing control of the unit is implemented by counters & PROMs. Another pair of PROMs defines the configuration control i.e. the number of analogue and digital channels, to be used. The sampling rate is variable within the range of 2-24 samples per fundamental system cycle, corresponding to a period of 20ms at 50 Hz and 16.7 ms at 60 Hz.

The unit thus has good flexibility and its internal interface controller relieves the main protection processor of the simple, but time consuming, data acquisition control overheads.

2.2 The protection processor

Studies of existing algorithms for protection, indicate that a processor with a speed close to that of a minicomputer is required. In addition accuracy requirements demand the use of a 16-bit machine. It would be also desirable to construct a flexible system which could be tailored to specific applications.

All the above requirements are met by using a bit-slice bipolar microprocessor. The processor is constructed using the INTEL 3000 device, block diagram of the microcomputer scheme being given in "Fig. 3". In the following sections some critical design factors are described.

2.2.1 Processor control

The INTEL 3000 is a microprogrammable processor meaning that the control portion of the processor is not hard-wired but is software defined by instructions stored in the micromemory. Micro-programmable control permits a flexible instruction set to be defined for specific applications. Additionally it simplifies hardware design.

The control module consists of the microsequencer which controls the execution of the microinstructions, stored in the micromemory. Pipeline registers are used to permit a parallel fetch/ execution cycle, which improves the microcycle time by 30%.

Each microinstruction controls the CPU,the memories, the I/O ports and defines the address of the next microinstruction to be fetched. A microinstruction word of 32 bits satisfies those requirements. Each main program instruction (macroinstruction), is thus executed by a sequence of microinstructions.

A max. of 512 locations can be addressed by the microsequencer but on average each macroinstruction is executed by just five microinstructions.

2.2.2 The CPU

The 16 bits CPU of the processor, comprises an array of 8 2-bit slice elements. It provides one accumulator, eleven internal registers and a macro-memory address register (MAR). The accumulator is connected directly to the data bus of the processor. The macromemory is addressed by a separate bus which is defined by the MAR. Some of the internal registers have been designated as program counter, loop counter, subroutine return address register, as well as for other special operations such as multiplication and multiple shifts.

2.2.3 Data buses

The macromemory address bus has been discussed above. The data bus is bidirectional, through which the CPU communicates with the memories and I/O ports. Some other buses, for example the mask-ing (bit testing) bus and flag input bus which improve the system timing are used only at the microinstruction level.

2.2.4 Main memory (macromemory)

Although the CPU can address a maximum of 64 K of memory, this is greatly in excess of the require-ments for dedicated protection purposes. From existing algorithms, is found that a 2 K memory is adequate. By employing a 6-bit operation code, we can address the whole memory directly which saves time and leads to a simpler design. The relay application program must be stored in a nonvolatile memory, for which EPROMs have been used. These devices offer high storage density, low cost, low power consumption and they are erasable. A pipe-line architecture is employed in the macromemory to provide speed improvements.

2.2.5 Data and constant store

Typically less than 200 locations are required for the storage of data and constants. As the data storage is limited, a complete separation of the program and data memories produces reduction of program counter manipulations. Addresses for the data memory are supplied via the subaddress bus of "Fig. 3", from the 10 lsbits of the macroinstruc-tions. 1024 locations of storage or I/O ports can thus be directly addressed. The data memory con-sists of banks of RAMs, each 128-word deep and of PROMs each 32-word deep.

2.2.6 Input - output (I/O) ports

The input port provides an interface between the processor data store and the output buffer of the data acquisition unit.

The data output module is dedicated to communication with a supervisory, back-up/monitoring computer. I/O communications are controlled by flags.

For CB control an additional output port is provided with three states, namely: i) trip/relay healthy, ii) no trip/relay healthy and iii) no trip/relay failed. The trip signal is used to control CBs, whilst the relay status signal is decoded by back-up equipment.

2.2.7 Processor timing

In the design presented a complete microcycle takes 200 ns. The clock circuit has been specifically designed to have a stable and predictable start-up performance following initial switch-on or a transi-ent disturbance of the relay power supply.

3. INSTRUCTION SET

Since the processor is microprogrammable the instruc-tion set is not fixed, but can be defined by the user to satisfy a specific application. A basic set of typical minicomputer assembler level instructions has been used. Additionally, several instructions required by the protection algorithms have been included. The instruction set can be divided into 4 groups i.e. control, memory reference, operate and jump/skip instructions. Each instruction consists of 2 fields, a 6-bit operation-code and a 10 bit address/constant field. Therefore a maximum of 64 instructions can be defined. The average instruc-tion exccution time is approximately 0.8-1 μs, which

indicates that the main objective i.e. to construct a microprocessor with speed similar to a minicomputer has been satisfied. This performance was only achievable by employing a bipolar microprogrammable processor with an extensive pipeline configuration.

As multiplication is one of the main features that must be included in the processor, a microprogrammed implementation rather than a hardware one has been chosen. This approach reduced both cost and power requirements. Typically the multiplication routine handles two signed 16-bit numbers and produces a 32-bit product in 17 μs.

4. DEVELOPMENT AIDS

To facilitate development of protection processor software, an assembler for the processor language and an EPROM programmer control package have been written. These aids are resident in a host minicomputer (NOVA 3). Another PROM programmer has been built, for programming the bipolar PROMs used in the micromemory and in the constant store.

For the initial testing and development of the microinstructions a bipolar PROM simulator was employed. Development time can be reduced if application routines are first tested using a MOS PROM simulator. For this purpose another PROM simulator has been built which can accommodate up to 512 16-bit words.

Finally extensive programs have been written in the monitoring computer, to enable the relay performance to be assessed. Additionally a monitoring program, which types fault reports has been written.

5. APPLICATIONS

The great advantage of a digital relay is that common hardware can be used for different protection purposes, simply by changing EPROMs, which contain the protection algorithms.In the following sections, a brief discussion of some protection schemes is given.

5.1 Generator negative sequence protection

The protection of expensive generating plant is an important area of relay application. Specifically, prolonged asymmetrical loading of a generator can produce considerable heating within the machine, through negative sequence current generation in the rotor. A digital negative sequence protection relay has been developed using the microprocessor equipment described. The relay algorithm employs symmetrical component techniques to determine the magnitude of the negative sequence components present in the generator load currents. Vector rotation required by symmetrical component analysis is readily available if the a.c. quantities are sampled at 12 times per power system cycle.

A further portion of the algorithm then evaluates a tripping characteristic from the expression:

$$I_2^2 t = K$$

where I_2 = negative sequence current magnitude
t = time to trip
K = constant dependent upon the heating characteristics of a particular machine type.

Laboratory tests have shown that the digital implementation fully satisfies the requirements of the protection function. In several areas, particularly those of alarm facilities and reduced frequency sensitivity in the digital negative sequence filter, the microprocessor relay represents an improvement on existing static analogue equipments.

5.2 Distance relay

A distance protection algorithm, is now being implemented. It uses the McInnes and Morrison concept of a transmission line composed of R and L only[4]. A simple algorithm can be written which solves the equation $Ri + L\,di/dt = V$, using first differences[5].

The computed R and L are compared with the nominal values and a decision is taken according to the conditions: i) no fault, ii) external fault and iii) internal fault. For this algorithm the primary system quantities are sampled eight times per cycle. Between each sampling point i.e. 2.5ms, the relay executes 6 impedance calculations, i.e. three to detect phase-phase faults, and three more for phase-earth faults.

Tests have been carried out for a phase-phase fault which indicate a detection time of about 3/4 cycle "Fig.4". For close-up faults a directional element is needed to discriminate between line faults and faults behind the relay. A voltage memory principle is easy to apply, as the prefault voltages are available. The same relay structure has been used for overcurrent, earth faults, transformer and generator unit protection.

6. CONCLUSIONS

An hierarchical digital protection scheme can be implemented, with an improved reliability and reasonable cost, by using front line relays of the type described. The system offers good flexibility and compatability in a complete digital protection scheme. From the manufacturers point of view standardisation of relay hardware is an attractive feature.

Further, differing protective functions, can be easily implemented with characteristics to accommodate changes in the power system. The fault report produced by the monitoring computer, is also a very useful addition in the system operation.

REFERENCES

1. Protective relays application guide by G.E.C.

2. B.J. Cory, G. Dromey, B.E. Murray: Digital system for protection. CIGRÉ 1976, Paper 34-08.

3. L. Petrou, E. Horne, B.J.Cory: Microcomputer applications in protection relaying. UPEC meeting 1977.

4. A.D. McInnes, I.F. Morrison: Real time calculation of resistance and reactance for transmission line protection by digital computer. IEA (Australia) March 1971.

5. W.D. Breingan, M.M. Chen, T.F. Gallen: The laboratory investigation of a digital system for the protection of transmission lines. IEEE PES meeting 1977.

6. Justin Rattner: Bipolar LSI computing elements usher in new era of digital design. Electronics Sept. 5, 1974.

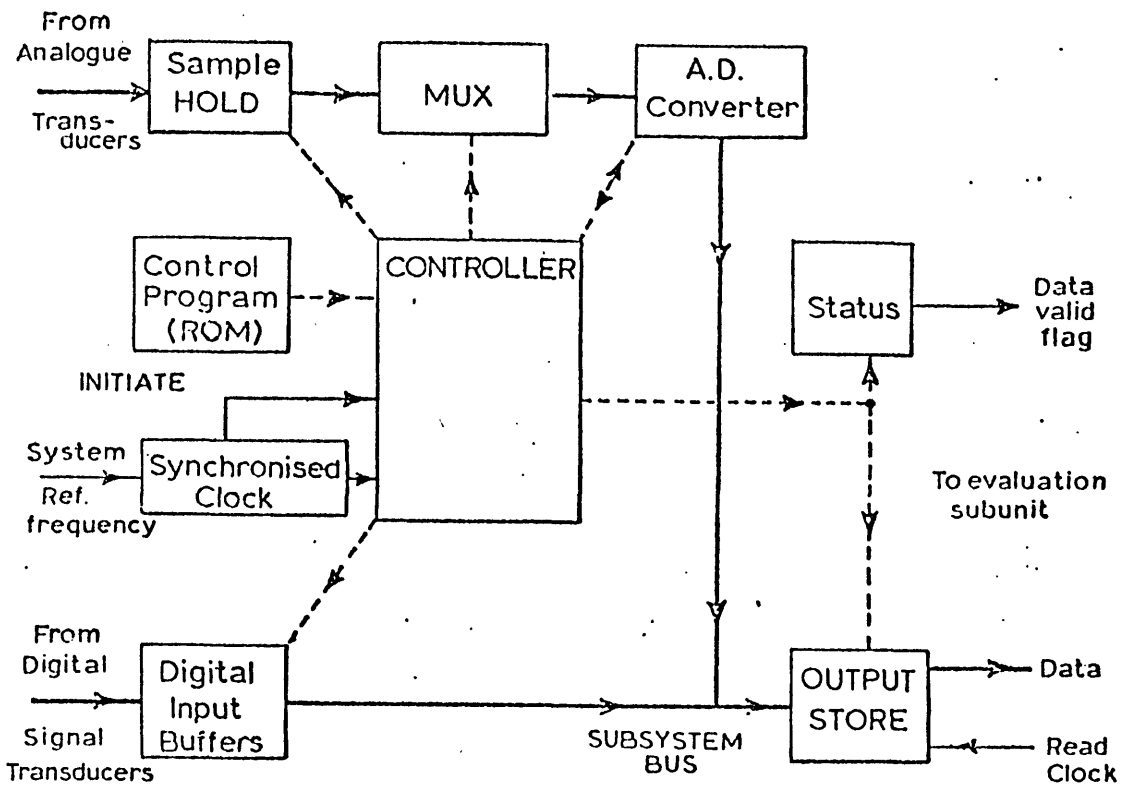
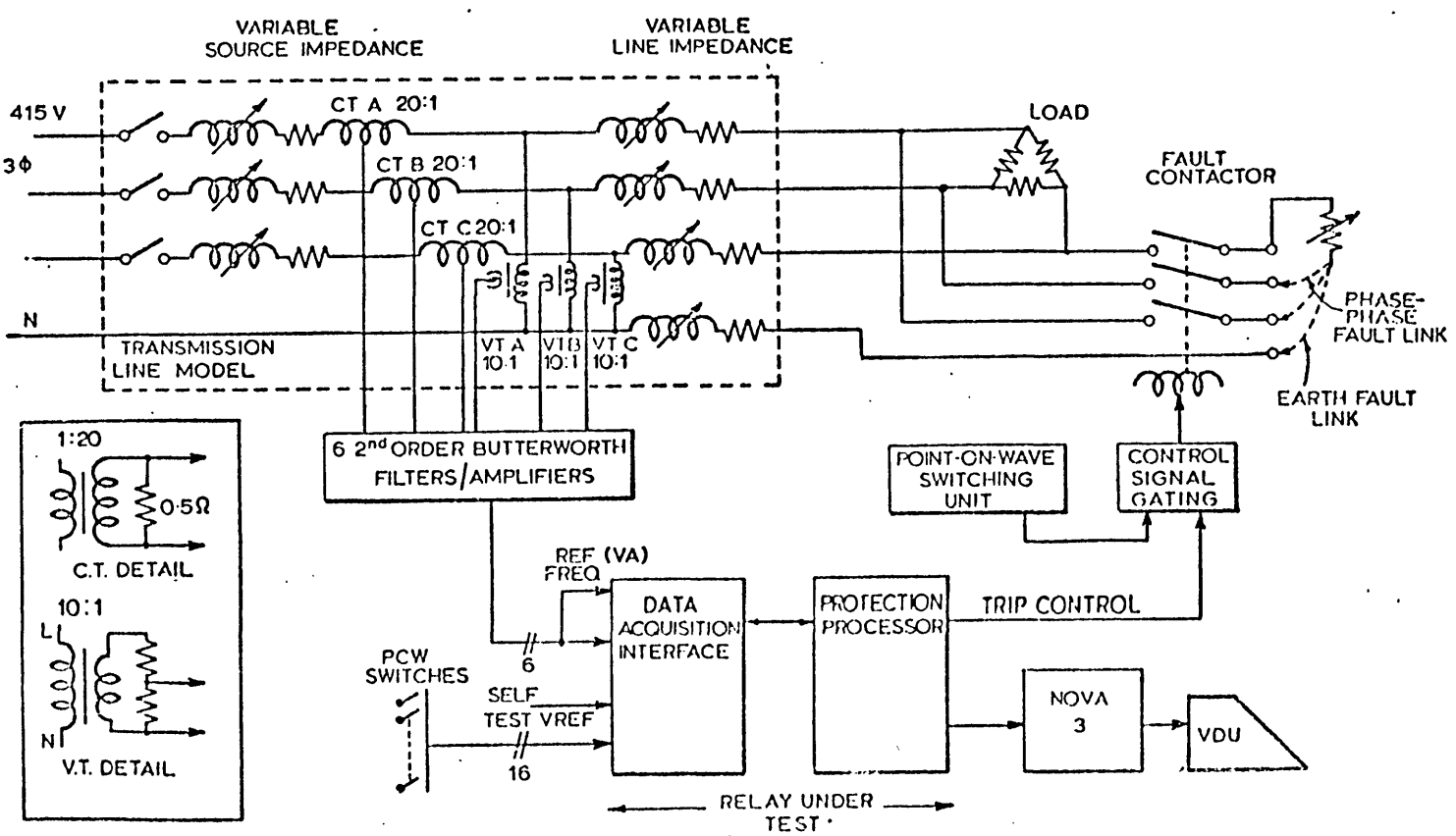


Fig.1 The data aquisition interface



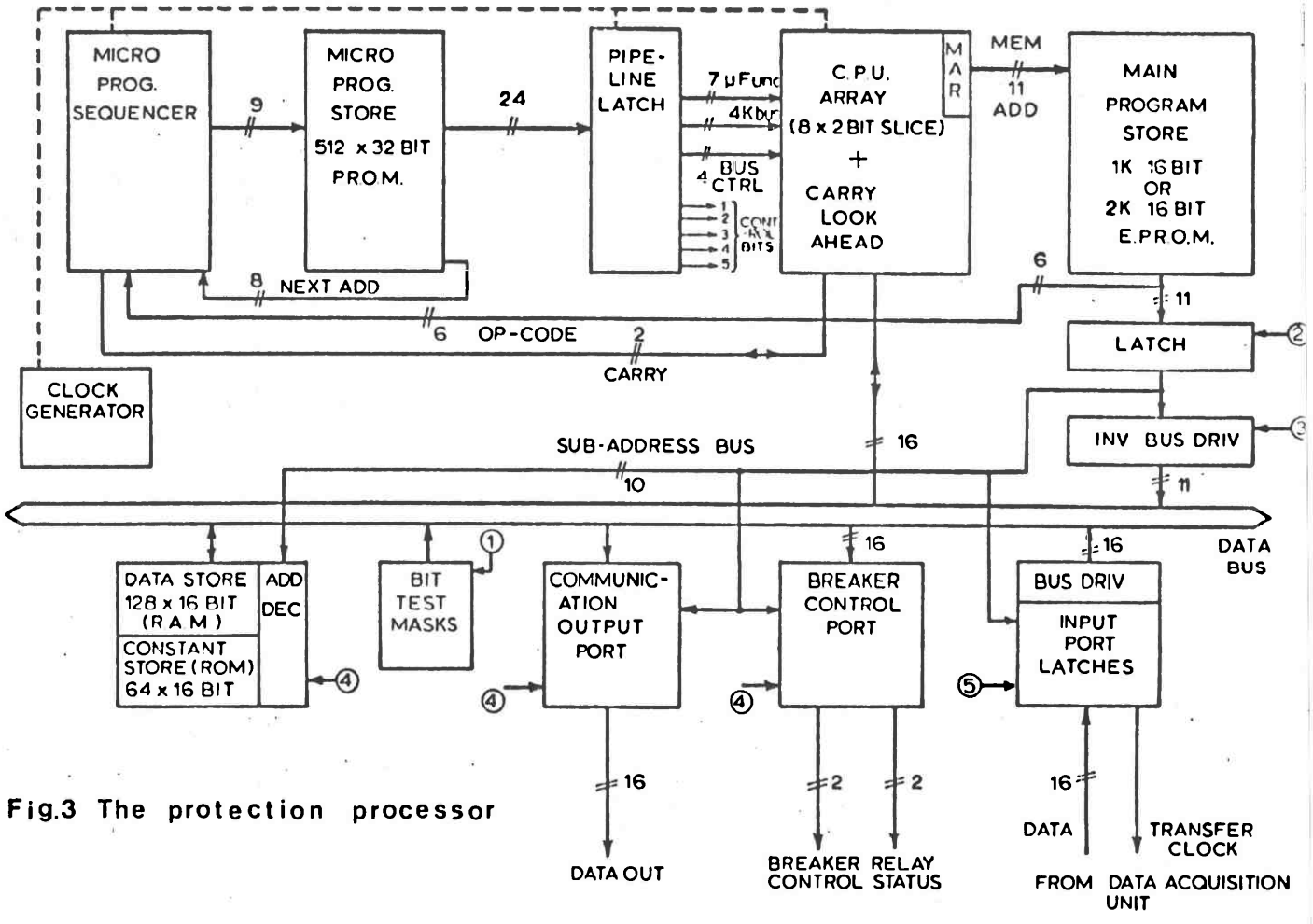


Fig.3 The protection processor

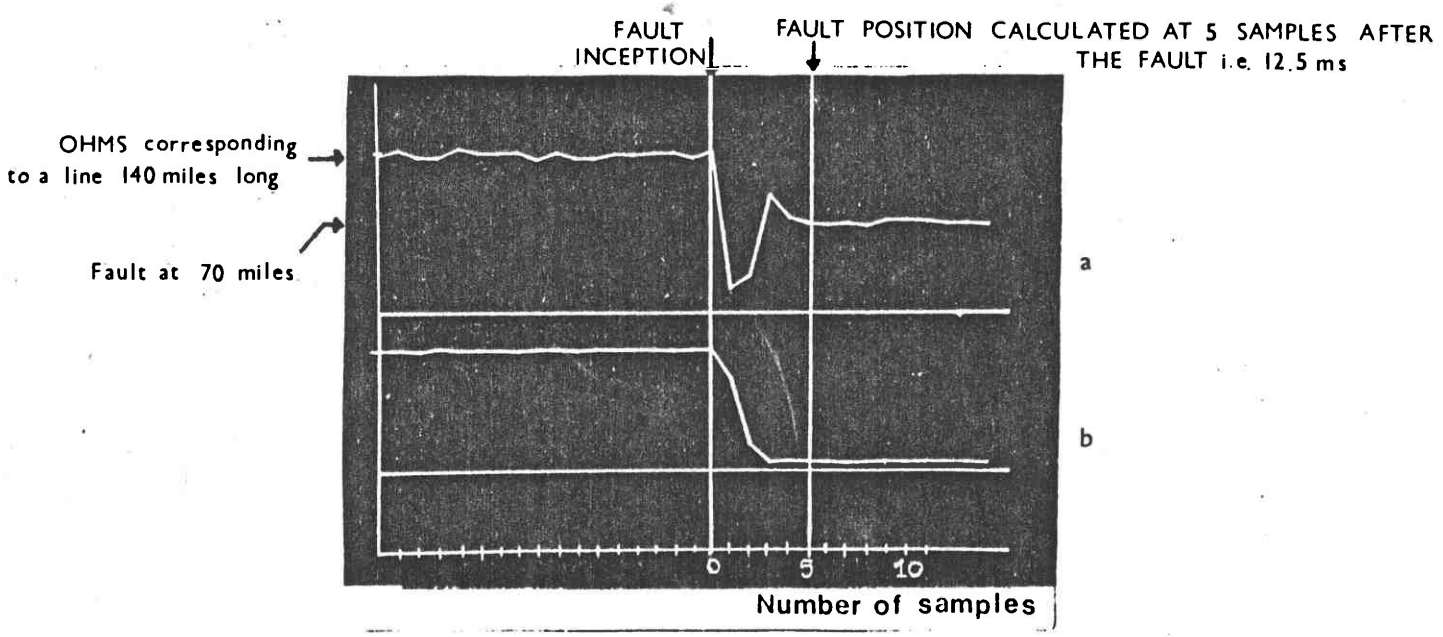


Fig.4 (a)reactance,(b)resistance for double phase fault calculated by the processor

Supporting papers

(1) "Digital processors for substation switching and control." IEE conference on modern developments in protection. March 1975

(2) "Micro-computer applications in protection relaying." 12th Universities power engineering conference. April 1977

(3) "A microprocessor power system protection relay." Panhellenic Society of Mechanical and Electrical Engineers, conference MECO'78. June 1978