4th International Conference on

TRENDS IN ON-LINE COMPUTER CONTROL SYSTEMS

5-8 April 1982

IEE

Conference Publication Number 208

Fourth International Conference on

Trends in On-Line Computer Control Systems

5-8 April 1982

Organised by the
Computing and Control Division of the
Institution of Electrical Engineers

in association with the
British Computer Society (BCS)
Institute of Electrical and Electronics Engineers Inc
(United Kingdom and Republic of Ireland Section) (IEEE(UKRIS))
Institute of Mathematics and Its Applications (IMA)
Institute of Measurement and Control (IMC)
Institute of Physics (IoP)
Institution of Chemical Engineers (IChemE)
Institution of Mechanical Engineers (IMechE)

Venue
University of Warwick, UK

Organising Committee

C G Eades *(Chairman)*

A J Bill

P M Burke

M J H Chandler

P P Docherty *(BCS)*

M S Jennions

T T McCarthy

C McKenzie

D G Parkinson

R J Preston-Whyte

Dr B V Raja Rao *(IMC)*

R J Scott-Kerr

Dr G K Steel

Professor D R Wilson *(IEEE(UKRIS))*

Published by the Institution of Electrical Engineers, London and New York, ISBN: 0 85296256 8 ISSN: 0537-9989

The cover was designed by GWA Design Consultants and is an abstract design based upon system symbols and suggesting 'trends' by the arrowhead formation.

Produced by Bamber Press Ltd, UK

Contents

The Institution of Electrical Engineers is not, as a body, responsible for the opinions expressed by individual authors or speakers.

N.B. For convenience the Review Papers are published in a separate section at the back of the publication

Contents

Contents

**Not available for publication at the time of going to press.*

List of Authors

TRULY DISTRIBUTED CONTROL, USING ONE MICROPROCESSOR PER REAL-TIME TASK

J. Altaber, F. Beck, M.C. Crowley-Milling, R. Rausch

CERN, European Organization for Nuclear Research, Switzerland

INTRODUCTION

At the CERN SPS accelerator, we have developed a strategy of control, using a network of geographically distributed computers, each computer having a computing task of the responsibility for a certain amount of hardware. Control problems are solved by executing interpretive programs in various locations of the network, some programs requiring the collaboration of several computers for their completion. Control programs can be launched at various places in the system: they may be started by an external stimulus or a scheduled event, and this can happen on any computer with hardware connected; they may originate at the operator interface by human intervention; or they may be requested from another computer as a sub-process of one of the previous two types. The specially developed SYNTRON multi-computer operating system used in this control system deals with all of these kinds (which we call 'class') of task, allocating a software interrupt level for each class and allowing space in each computer for one task of each class to be resident concurrently. The classes of task provided for are as follows:

- short, urgent jobs to be performed for other computers,
- normal jobs to be performed for other computers,
- local operator interaction,
- scheduled programs.

The operating system in every computer manages these tasks. Tasks which cannot currently be performed because a task in their class is already in execution, are correctly queued until their particular class is free. The result is a truly distributed control system in which all resources are available from any computer of the network.

This paper will describe a project, currently under way, to replace a computer of this network by an assembly of microprocessor units, each of which performs a single class of tasks. For the realisation of this project the proposed E3S standard (1) for a crate and interconnecting bus is very suitable and the reason why this is so will be outlined. An incomplete model of the proposed assembly has been built in CAMAC. This model, and what has been learnt from it, will be described.

MULTIPROGRAMMING ASSEMBLY

A full multiprocessing assembly will consist of a number of microcomputer plug-ins, some of which can be used for performing the different classes of tasks referred to above, and others, with additional hardware, acting as I/O units. All these units must be connected to the bus in such a way that information can flow between I/O units and task microcomputers. In addition, tasks must be capable of being supervised by a scheduler microcomputer. It should be noted that communications between these units will usually take the form of a 'message', rather than of simple one-word commands as in control interfacing.

We are thus faced with the requirement that the units must be able to exchange messages, that it must be possible to establish temporary master-slave relations between individual units, and that such a relationship, once established, should exclude access to the slave resource by other would-be masters. It is clearly advantageous if neither the building of a master-slave relationship nor any subsequent transactions require any action by equipment uninvolved in the relationship: i.e. the reservation mechanism should be distributed.

The desired results can be achieved by a number of general purpose processing units all having access to a suitable bus. This bus must also allow communication with peripherals, such as hardware under control and data-links to other multiprocessing assemblies. Figure 1 shows the general layout of such an assembly. The following are the properties required of the processing unit and of the bus.

The processing unit

The processing unit will normally take the form of a plug-in unit in a crate or card-cage. To constitute a useful module in a modular system, it should contain a general-purpose microprocessor, in compact form, and with sufficient memory to avoid the need for using the bus during the normal operation of the computer. A multicomputer arrangement which avoids the need for common memory allows each unit to do its own work, independently of the others, and to communicate with the other units only when this is functionally necessary. The unit should be physically compact, occupying one printed circuit card if possible. If a one-card design is impossible, use of a second card is tolerable if communication between the two avoids use of the common bus.

A unit should be able to reserve another unit on the bus for its own temporary use. The mechanism for doing this should be totally symmetrical, so that A can reserve B in exactly the same way as B would reserve A. Thus each unit should contain a reservation register, and each should have implemented a 'test and set' instruction which can operate on any other unit's reservation register. The input-output arrangements must also be suitably constructed so that each unit should be capable of outputting single words or blocks of data directly into the memory of another, provided that the other has previously been reserved.

An arrangement like this, ideal for multiprocessing, can only be achieved if the test-and-set, as well as all accesses to the memories of other units, is presented to the bus accompanied by the identity of the originator. This enables the receiving unit to accept the reservation and set itself as being the temporary property of another, identified unit. It also enables the hardware of the receiving unit to accept subsequent accesses only if they originate from the reservee, and to reject them if they are from any other unit. This is a vital feature in a multiprocessing environment, corresponding to memory protection in a multiprogramming machine. To complement the 'test-and-set' instruction there must of course be a 'release' instruction, and all reservations

must be subject to a time-out, so that a failure to release a reserved unit does not block that unit for ever. Interrupts, too, are included in the reservation scheme, so that a master may pre-set the interrupt vector available to the slave.

To sum up, a general-purpose unit must have the following features:

- A powerful local microcomputer,
- Input-output, via the bus, of words data blocks and interrupt vectors destined for an addressed area in any other unit,
- Facilities for registering temporary ownership of other units and of being reserved by another unit in a similar way (test-and-set),
- Capability of retaining bus mastership for the completion of compound bus operations,
- A filtering mechanism, ensuring that data-transfers are only accepted from the unit which has done a successful reservation, and an input-output mechanism which sends information with an identification of the source unit.

The bus

A bus for connecting general purpose units of this kind must allow the use of an arbitrator to give word-cycles in interleaved form to all active units. Fairness in bus allocation to processors is required in a symmetric scheme of this kind. Implementation of the reservation mechanism does require that a unit retains access to the bus for a small number of cycles, in order to complete an uninterruptible chain of instructions (example: test-and-set, transfer of a multi-precision word). The bus must be able to carry data accompanied by source addresses from any unit to any identified destination address, and it must have the address and data lines necessary for this, as well as provision for the protocol of the word transfer in each case.

Additional requirements for the bus are the ability to allow, in an uncomplicated way, for the use of extension crates, and to allow easy connections to external hardware.

The E3S (ESONE Small System Standard) bus, which has been submitted to the European Standards Organization on Nuclear Electronics for acceptance, is designed with just such a multi-master assembly in mind. The crates and cards comply with IEC standard 297. (The E3S system using cards 220 by 233 terminated in 64 pin connectors mating with connectors attached to the printed backplane.) Units have integral address recognition and are therefore position independent, and can be exchanged without switching off power, an important requirement for the realisation of a maintainable system. The construction of a minicomputer replacement in a multicomputer environment requires the design of the following units:

- an arbiter with suitable servicing algorithm,
- a General Purpose Processing Unit (GPPU),
- a Data Link Driver Unit (DLDU) for network connection,
- a Multidrop Bus Controller (MDBC) for equipment connection,
- a Mass Storage Driver.

DESIGN PROJECT

A project is currently under way for the design of these units, and for the re-writing of the minicomputer system software to suit them. It should be noted that the multiprocessing assembly needs a scheduler, housed in a dedicated GPPU plug-in, but no conventional operating system, since each GPPU of the assembly works in a singlestream mode. Using the reservation mechanisms described, a processing unit running a real-time program can temporarily reserve another which is dedicated to a data-link driver or other output unit, and release it after the data-transfer transaction is completed. Running processes can schedule others, and so on. Cross-assembly is used from larger computers, with down-line loading of the assembled system software.

It is intended that this two year project, involving about 15 people, all of whom have other primary responsibilities, will result in the construction of multiprocessing assemblies which will replace any individual computer currently performing control, calculation or message-switching functions in our 50 computer control network. The completed assemblies could be used for the network extensions needed for CERN's projected machine LEP, which we hope will be approved at the end of 1981. They will in any case be invaluable as and when the minicomputers in the current control network, some already nine years old, become too old to be maintained.

It is expected that units like those described, built to an international standard with portable system software, can survive the vicissitudes of technological obsolescence much better than commercial minicomputers, which become expensive to maintain, rather than wearing out, in a decade. The life of a large accelerator is usually reckoned to be at least twenty years, or about twice that expected from its control computers. Preliminary estimates put the cost of the multiprocessing arrangement at about half that of the minicomputer it would replace, and similar speeds are easily achieved with modern microcomputers, since processes can often be simultaneous and interrupts, being dealt with by a dedicated processor, usually suffer no latency.

THE MODEL

The ideas involved in task reservation and the operation of a multi-master, multiprocessing assembly have already been tested in a CAMAC model. CAMAC was chosen as a construction norm for the experiment since a suitable computer plug-in and all the necessary software infrastructure already existed in the form of an autonomous crate controller (SPS 2420) (2). A NORD-100 computer of our network was used in the first place as a simulator for all those functions (Data Link Driver, Equipment Interfacing) which needed to be performed for a demonstration (see fig. 2). The latest version of the model is an independent computer replacement (see fig. 3). The photograph (fig. 4) shows a laboratory test set-up having six GPPU's, each consisting of a CAMAC autonomous crate controller containing a TEXAS 9900 16-bit microprocessor, and a communications card with the reservation mechanism (fig. 5).

The laboratory model works, and can already run scheduled and interactive tasks, perform tasks for other computers in the network, and cause tasks to be performed in other computers.

Weaknesses of the CAMAC norm are becoming apparent. Because it is impossible to execute uninterruptable multi-instruction sequences, reserved units cannot be protected from access by units other than the reserver except by the use of elaborate and slow software protocols. The absence of a continuation crate addressing scheme might make it difficult to make a practical assembly. In CAMAC, the arbitration algorithm is of the priority type, while for this project a 'fairer' algorithm is desirable. Multi-master operation in

CAMAC requires a second bus, which is mechanically difficult to implement. While CAMAC is a valuable international standard, it was never designed with multi-master operation of this kind in mind.

CONCLUSIONS

Given that the system architecture of a real-time control network is so configured that the number of simultaneous tasks per computer is never large, a computer can be replaced by a multiprocessing assembly. A laboratory set-up, using the CAMAC standard, has been built, and used to assess the interprocessor protocol problems and the speed and cost potentialities of the technique.

A prototype multiprocessing assembly is being designed in the E3S norm being considered by the ESONE committee as a European standard. This norm has all the mechanical, electrical and informatic properties needed for the construction of an assembly of this kind.

The prototype multiprocessing assembly will be assessed in the context of a control system for a projected very large accelerator. With the exception of applications requiring high-powered computing, this control system will probably only use assemblies of this type, but its network will be an extension of an existing control system network using conventional computers. Software compatibility between the two networks is a design criterion.

Work done to date shows that both the speed and the cost of this technique compare favourably with those of existing minicomputers, but the advantages of a simple modular construction in the fields of reliability, maintainability, possibilities for expansion and reconfiguration and above all freedom from manufacturer-dependent obsolescence, augur well for its success in the computer control of large complexes like research accelerators, which are subject to continuous development and therefore require flexible and expandable control systems.

REFERENCES

1. E3S - Proposal for a Small System Standard. ESONE Committee, Small System Study Group, CEC - Brussels, Belgium.

2. Beck, F., Guillaume, C., Kugler, H., Rabany, M., Rausch, R., 1978, "Auxiliary CAMAC crate controllers using a 16 bit microprocessor. Applications in accelerator control", ESONE AGA, DESY, Hamburg.

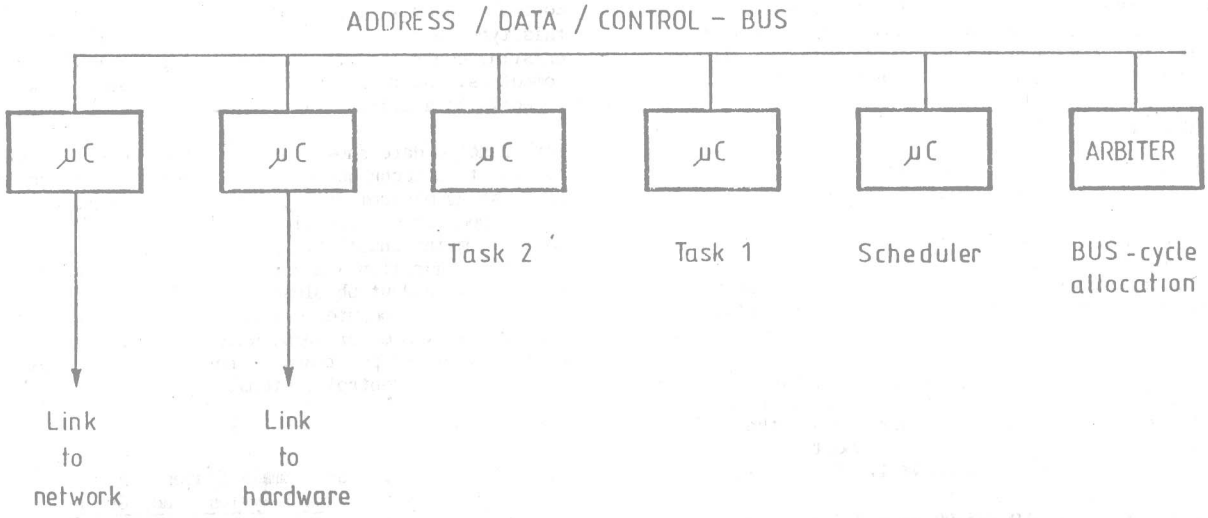

Figure 1 Typical multi-microprocessor configuration

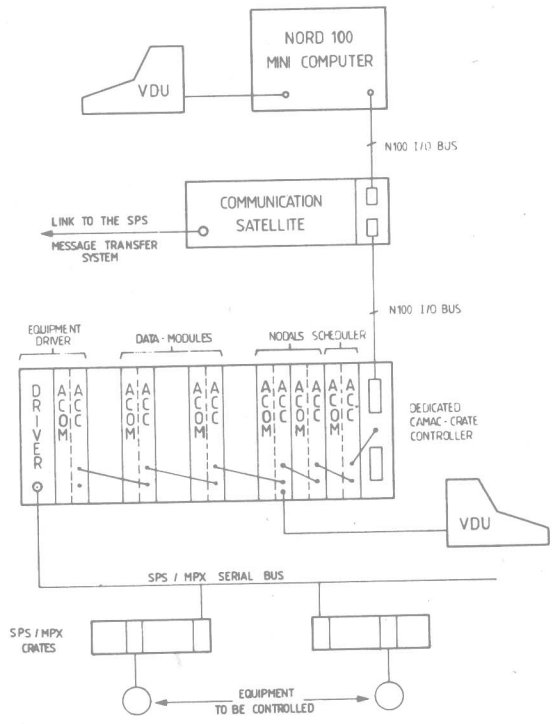

Figure 2 Multiprocessing Assembly (MPA)
Phase 1 (September 1981)

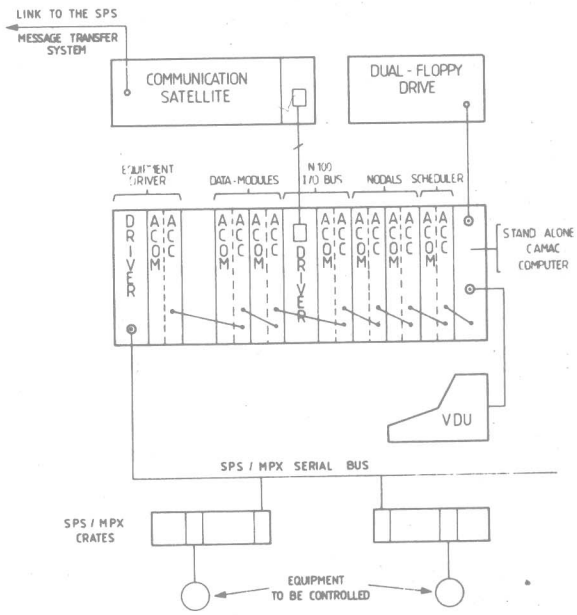

Figure 3 Multiprocessing Assembly (MPA)
Phase 2 (end 1981)

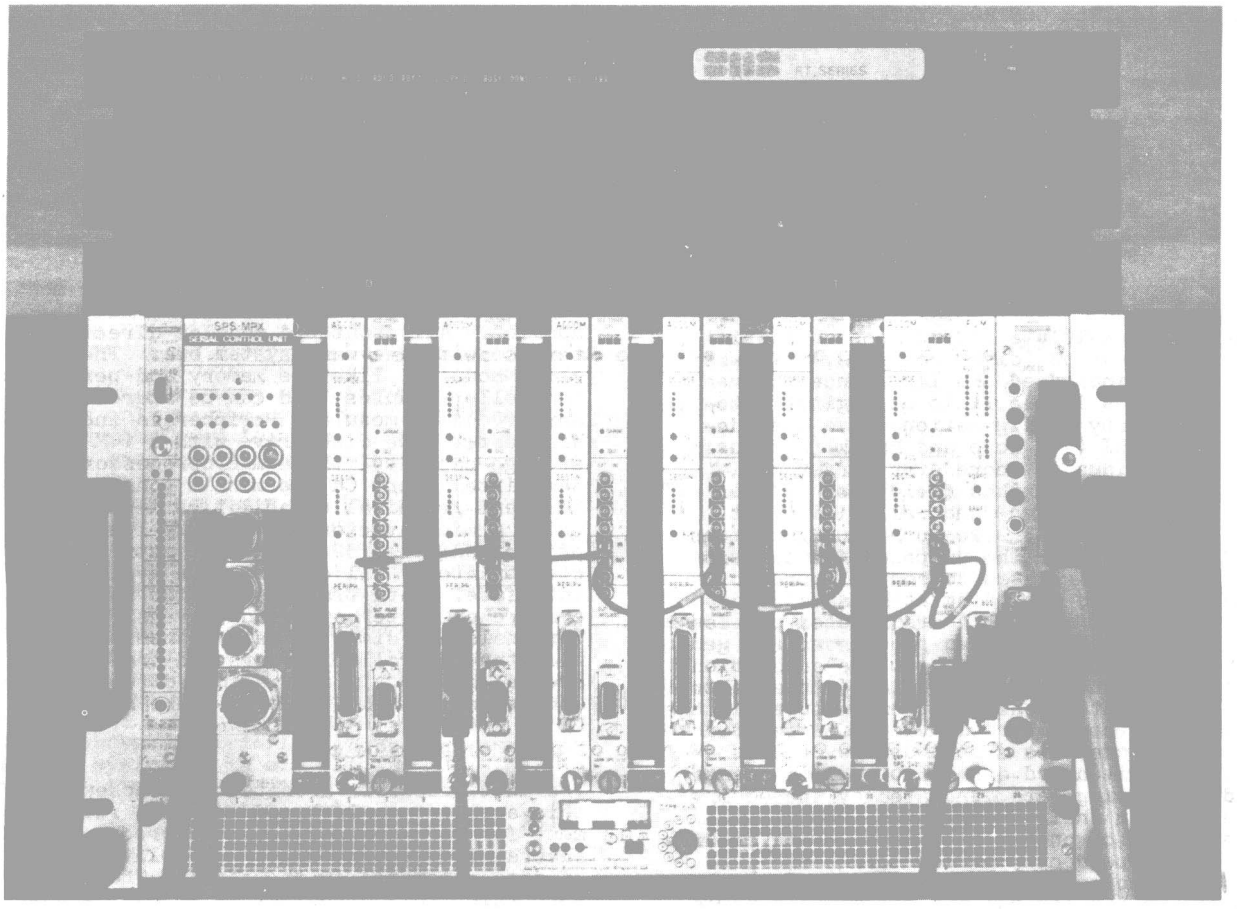

Figure 4

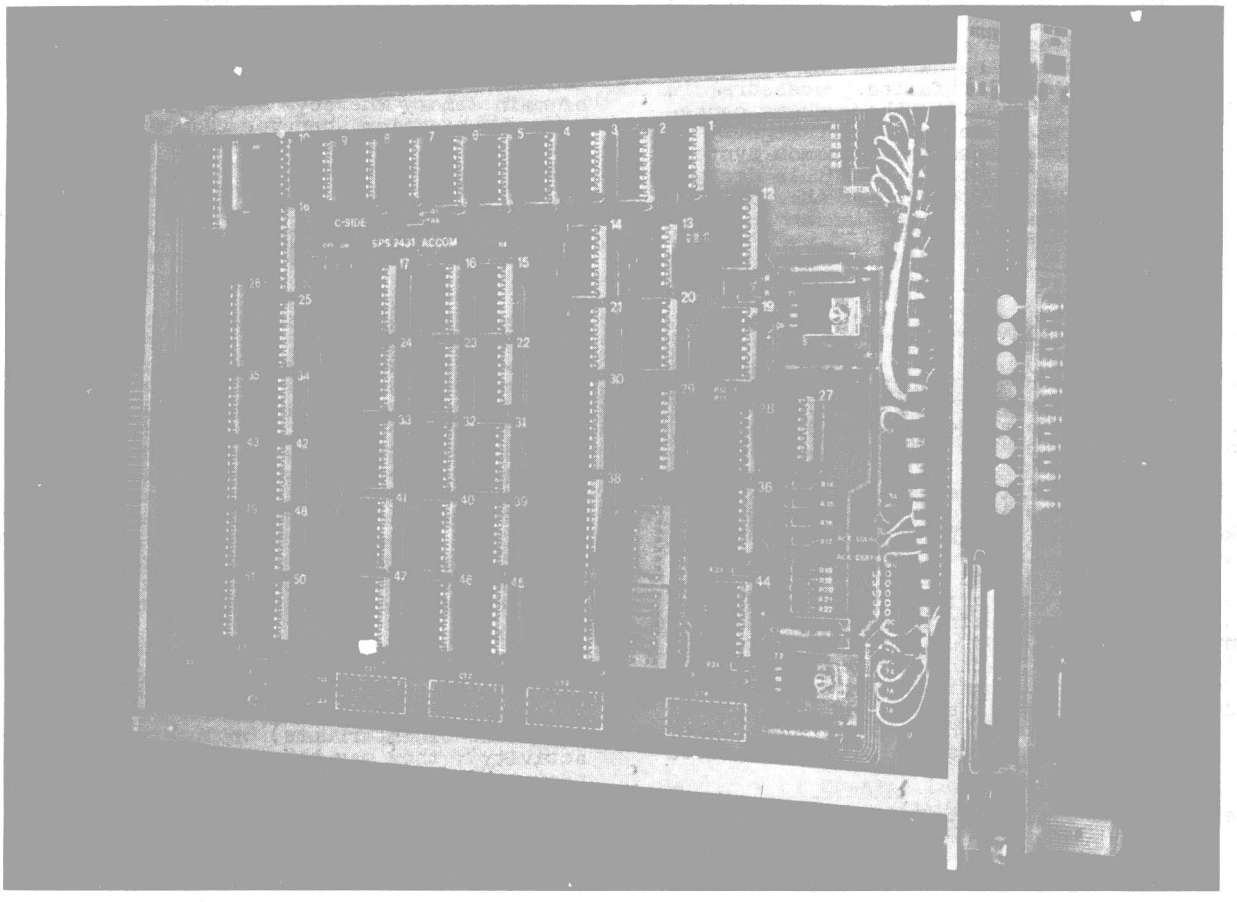

Figure 5

ON-LINE DIGITAL CONTROL SYSTEMS FOR HIGH VOLTAGE SUBSTATIONS

B.S.Sharma S.A. Bajpai L.K. Gupta

U.P. State Electricity Board, India Uptron Digital Systems Limited, India

INTRODUCTION

The use of multiple microprocessor sharing costly resources (disk, consoles, printers etc.) has increased the importance of even 8 bit micros in the field of digital automation. Many configurations have been designed (EPRI (1), Hope and Malik (2)) using this technique to score over larger minicomputers used in this field, not only in terms of performance and reliability but also economically. Two such configurations were detailed for the purpose of automating all functions related to control, protection, monitoring and recording at HV substations. One design utilises the distributed processor concept. In this design processors are geographically distributed over the substation switchyard and loosely coupled with a central processor (star configuration), housed in the substation control room, via communication links. Processors distributed over the substation switchyard are equipped with front ends acquiring data generated by the equipment within their geographical boundaries consisting of one or more bays. Data acquired is independently processed for the purpose of protection and control and is transmitted to the central processor which performs the recording and monitoring functions apart from handling the man-machine interaction and co-ordinating resulting changes in the processors distributed in the switchyard. The central processor also detects the failure of other processors and handles the reorganisation of functions, of failed processors, in the active processors. Another scheme concerns the use of multiple processors tightly coupled together and sharing a common system database. The application using this design is illustrated with a portion of the substation depicted in the line diagram in figure 1.

SYSTEM OVERVIEW

Hardware Philosophy

The multiprocessor system hardware consists of several processor modules sharing common resources like global memory and I/O devices via the main system bus (figure 2).

Each processor module has its own local bus allowing access to its own private resources like 32K bytes of dual port RAM, programmable interval timers, interrupt controller etc. (depicted within dotted lines in figure 2). These modules are interfaced via bus arbitration logic to the main system bus. Bus contention is resolved using the parallel priority resolving technique (Nadir and McCormick (3)); priority of each processor module is programmable and is supervised by the operating system. Hardware support for interprocessor communication is also included on each module.

The front end comprises of Analog input multiplexer and Analog to Digital converter modules, digital input modules and digital output modules. All front end modules directly interface with the main system bus. These are slave modules, like the memory and peripheral controller modules, and can be addressed by any processor module. Peripherals include a Winchester type disk drive with a 14M byte storage capacity, three microprocessor based semi-intelligent CRT terminals (one of which has semi-graphical capabilities implemented using a micro vector set), a 300 lpm line printer and one 132 column dot matrix character printer. All devices have a parallel interface. Data transmission for both printers are character oriented whereas for the terminals and disk the data transmitted between the controller buffers and the devices are in the block mode via four direct memory access channels in the hand-shake mode.

Software Philosophy

The multiprocessor architecture chosen for this application is based upon the requirement for high speed concurrent processing of several independent and inter-related items of data generated in the field. The software supporting this architecture enables a high degree of parallelism by segregating related processing activities in independent environments, processor modules (Chu et al (4)), thereby minimising the clashes for system resources as much as possible. Environments running related activities are strongly coupled with each other via the shared 64K bytes of main memory thereby allowing inter-processor communication whenever required. The real power of the system lies in the flexible scheme of each environment being able to dynamically reconfigure itself in accordance with a suitable criterion, allowing the system to automatically, logically isolate malfunctioning modules and still perform at full speed.

Real Time Executive

The operating system is based upon a real time multi-tasking executive. The executive provides the facilities for scheduling of activities within a processor (or job) and despatching, communication between two activities, sharing of resources by request queue and flag management (semaphore), inter processor (job) communication and interrupt management. Also built into the executive are system fail-safe features using which the failure of a module is detected and the system is reorganised dynamically to logically isolate the failed module.

The basic unit i.e. the activity, is an application program which services one or more events. Events may be generated externally (by means of interrupts) or internally (by an activity); they may be synchronous (with respect to the real time) or asynchronous (unpredictable). For example, the start of a scan interval is an event generated in

conjunction with the real time clock. This event would drive the scan activity which will command the front end to scan analog and digital inputs, read the data presented by the front end, process this data and generate the event signalling the alarm processing activity to start. Activities also form a basis of code sharing in a manner analogous to reentrant procedures. Activities are grouped together to form jobs. The job is created by a collection of related activities and one or more jobs are associated with a processor; however, long jobs may be split over more than one processor. Each activity within a job is assigned a priority in accordance to which an activity might be scheduled to be ready and despatched to be run. Each job too is associated with a priority which is used by the scheduler only in the event of activities of different jobs communicating with each other or in the case of system reorganisation.

Activities may exist in the system in any one of the following states: running, ready, preempted or sleeping. Transition of activities from one state to another is dependent upon their priorities and occurrences of events.

Activities are also associated with a real time constraint. This defines the time in which the activity is to be completed once it has been started. The real time constraint provides the scheduler with information, to enforce a dynamic priority scheme ensuring that the system is at all times synchronised with real time and provides another dimension to ease the application system design (while assigning priorities to various activities), and also helps to debug the application system and resolve dead-lock situations not envisaged by the designer.

A watch-dog scheme ensures that failed modules are detected and system reorganisation is speedily invoked to isolate the bad module. If a processor module fails, the jobs associated with this module have to be redistributed. The detecting processor module suspends all its activities and forces the other good modules to go into the reorganisation mode. In each module, then, the alternative system configuration map, catering for the failed processor is read from a system file on disk and reconfiguration takes place after which the system starts afresh.

Operating System

Configured around the executive is the rest of the operating system in the form of several jobs. The association of these built-in jobs with processor modules are at the designer's discretion. These jobs are:-

- File management job which contains activities that cater for disk I/O events generated by other jobs. The activities in this job manage the complete mass storage as a named space for access. They consist of file directory management, file space management, multiple access and file security management, file input-output.

- Report generation job which contains activities that spool, format and print reports generated by other jobs. These activities are report formatters, spoolers to organise simultaneous multiple report generation and printer driving activities.

- Man-machine interface job consisting of activities which format the display of the 3 operator consoles and activities which interpret and act upon the commands given to the system via the keyboards. Other activities are screen management upon alarm acknowledgement, line diagram generation and update of values on diagram, synchronised with scan.

- Program development job consisting of activities like text editor, linkage editor, system generation interpreter (both static and dynamic), high level language compilers, user program debugger monitor and program library manager. All these activities are overlaid in a transient operating system program area, and are loaded from the disk, where they exist as system files, upon operator request.

Application Design

The control system application for high voltage substations for the purpose of recording, control, monitoring and protection has been subdivided into several activities which have been regrouped to form jobs and jobs grouped to be allocated to the processor modules. A judicious fragmentation and grouping has to be accomplished to:-

- maximise throughput by parallel processing of concurrent events,
- minimise inter activity communication specially those involving inter processor communications,
- achieve high degree of modularity for easy maintenance and expansion,
- cater for speedy system reorganisation still retaining maximum throughput and processing power in case of failure of one or more modules.

Following is a brief description of the activities illustrating a part of the application:-

Scanning. All inputs, analog and digital, which include current, voltage, frequency, phase difference, temperature, pressure, circuit breaker status and isolator switch positions are scanned at the speed of one millisecond for analog and two milliseconds for digital points. The scan event is generated by the real time clock every millisecond. This activity has a very acute real time constraint of one millisecond (activity A in figure 3).

Engineering Conversion. Analog inputs scanned are converted into their respective engineering units like amperes, KV, Hz, ^{o}C, angular degree etc. Values like power, reactive power and energy are derived from the converted values. Inference checks are performed on the converted values; bad inputs are annunciated on the output devices, the current database residing in global memory is updated with the good values, both scanned and derived (activity C in figure 3).

Alarm Annunciation. Limit checking for the purpose of alarm generation is performed on the values generated by engineering conversion. Messages are generated, in case of limit violations and digital point status changes, to be displayed on the alarm CRT and output on the printer (activity E in figure 3).

Sequence of Events. Logging of sequence of events on the printer corresponds to the relay operations associated with their

occurences in real time. This log is used for the purpose of fault diagnosis (activity D in figure 3).

Protection. Different kinds of operations are performed simultaneously on the scanned values for the purpose of protecting various equipments used in the high voltage substation.

Limit Checking- Limit checkings are performed on the currents and voltages scanned (activity F in figure 3).

Distance and time graded- Limit checking performed on impedance and direction derived from voltage and current inputs scanned. In the case of time graded protection a timer is initiated for the interval of the programmed delay (activity G in figure 3).

Differential- This involves the comparision of currents of two inputs which ideally should be the same within a tolerance limit (activity F in figure 3).

Other activities include overflux, over temperature, high pressure types of transformer protection. With each protection activity, anomalous situations cause control action to be taken in the form of digital outputs to operate circuit breakers for isolating protected zones. The effect of control action is also monitored and alternative strategies initiated in the case of circuit breaker failure.

Control Functions. Fluctuations in active and reactive power flow are checked for deviation beyond the specified limits. Deviation may require switching in or out of capacitor banks, changing of transformer tap positions, or a combination of both. The control action algorithm chosen in this activity is based upon the aim to choose the best possible alternative and maintain the optimum level of performance of the substation (activity E in figure 3).

Programmed Load-Shedding. The event invoking this activity may be external or internal to the computer system and is triggered by frequency deviation or an operator command via one of the operator consoles. The purpose of this activity is to perform load shedding, planned or emergency, during power shortage periods. The control action taken by this activity is to isolate one or more feeders taking into account time of the day, day in the week, feeder locations (rural or urban) and other past load shedding information (activity B in figure 3).

Auto-Reclosing. The events invoking this activity are generated by various protection activities. The purpose of this activity is to try and restore power to isolated areas by switching the tripped circuit breakers at programmed intervals for a specified number of times; failing which a warning message is output on the operator's console and on the printer (activity H in figure 3).

Logging. This activity can be triggered by events generated in conjunction with the real time clock upon operator request or by the alarm annunciation activity. The activity collects and transmits the relevant data to be printed out in a predefined format by the report generation job (activity J in figure 3).

Oscilloperturbography. This activity is invoked by various protection activities which have recognised fault conditions. This activity collects related pre-fault, fault-time and post-fault data from the database for a specified period, analyzes this data and generates a fault report (activity I in figure 3).

For effective concurrent processing consider the scanning of group of inputs and the conversion for the group of inputs to be sequential. But while the conversion of the group of inputs scanned is in progress scanning of the next group can be done in parallel. The sequence of events recording is also accomplished in parallel to the conversion of analog inputs. Alarm processing, protection and control come in the next level (after conversion) and can execute in parallel. Auto-reclosing and fault log preparation which are triggered by the events generated by the protection activities function in parallel. The logging activity which is triggered by events generated by alarm processing, protection, control, auto-reclose, fault analysis, operator interaction or the real time clock (at programmed log intervals) can function in parallel to any other activity but with a non-critical real time constraint; the other activities mentioned above have stringent time constraints associated with them. The distribution of these activities in jobs, and jobs in processors is illustrated in figure 3.

CONCLUSIONS

The utility of such systems for on-line control of substations is dependent totally on the requirement of the substation. In turn the substation requirement is generally dependent on the class of the substation. The base system configuration described above would prove uneconomical in LV substations where functions like sequence of events recording, oscilloperturbography, under frequency load shedding and auto-reclosing are usually not required. However, this scheme provides enough flexibility for upgradation, both in terms of hardware and software, to provide more functions like Reactor Protection, high speed bus-bar protection and telemetering which are required in EHV substations.

ACKNOWLEDGEMENT

The authors are grateful to Uptron Digital Systems Limited, for providing the necessary facilities to produce this paper.

REFERENCES

1. Workshop Proceedings, EPRI, 1980, "Digital Techniques for Control and Protection of Transmission class substations".

2. Hope, G.S., and Malik, O.P. 1980, IEE Conference Proceedings, 187, 40-45.

3. Nadir, J., and McCormick, B., 1980, Computer Design, 103.

4. Chu, W.W., Holloway, L.J., Lan, M., and Efe, K., 1980, IEEE Computer, 13, 57-69.

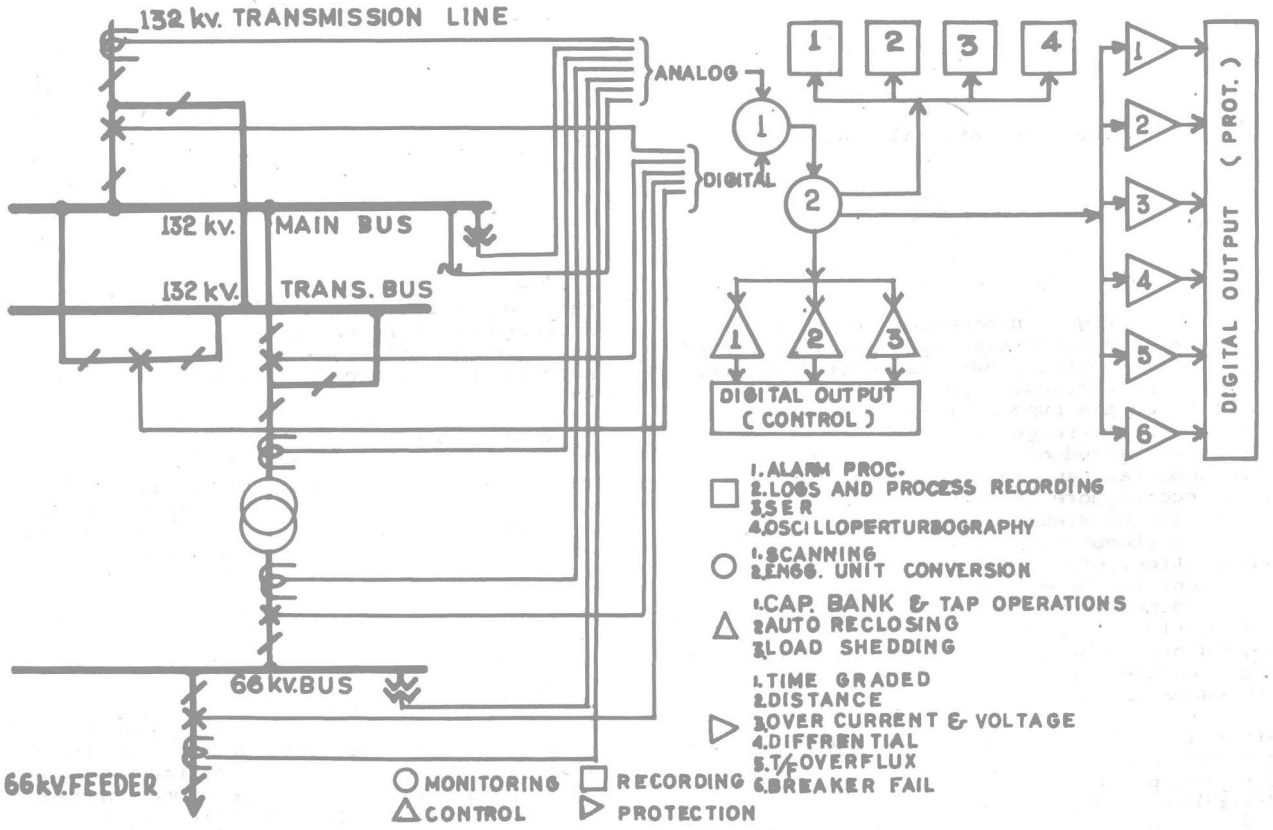

Figure 1 Association of system functions with a HV substation

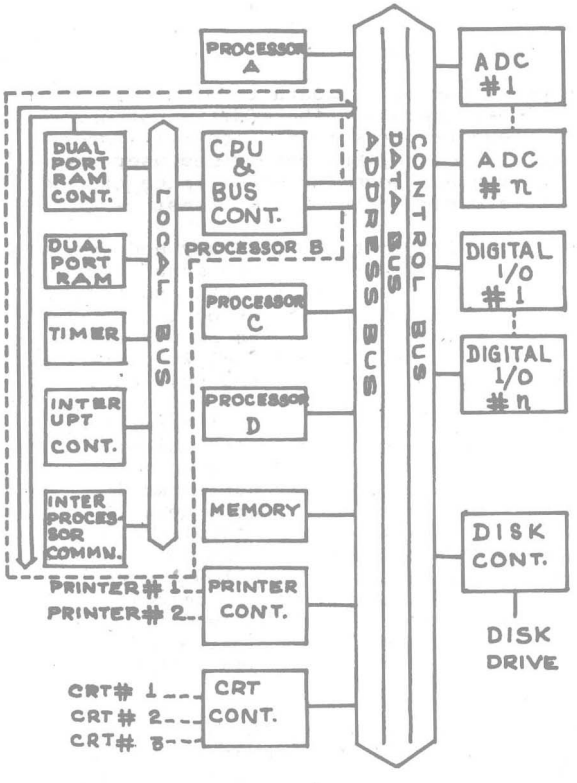

Figure 2 System architecture

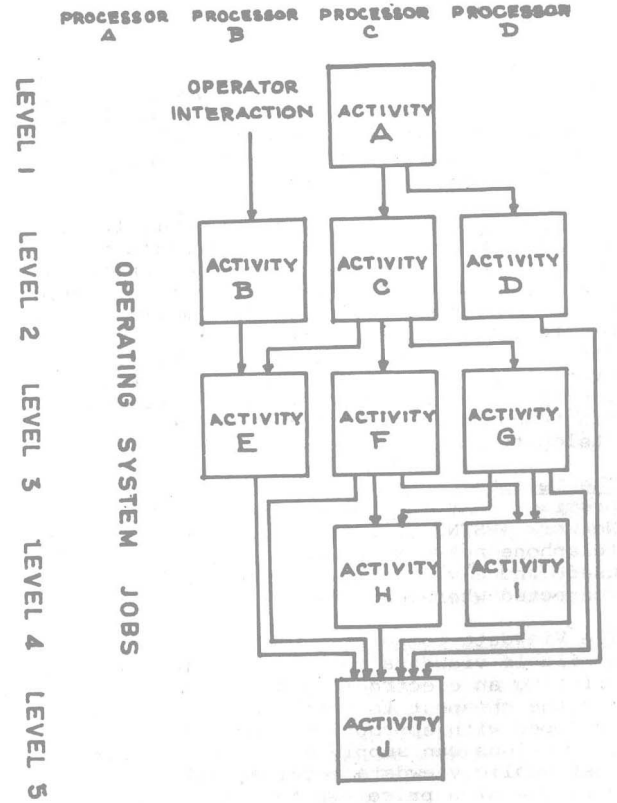

Figure 3 Application system structure

VIEWDATA AS A CONTROL TOOL

David R. Hirst

EDG Engineering International Ltd., UK.

INTRODUCTION

Getting the right information to the right people at the right time is a vital aspect of successful control. But, as control systems become more complex, sophisticated and distributed the number of right people increases, the range of right information increases, speed of communication becomes more important and the control system design task becomes more difficult. The recent technology of viewdata, now internationally known as videotex, and its prime commodity, information, offers a range of economic solutions to these problems without the risk of flooding the wrong people with the wrong information. But the possibilities extend beyond provision of information, offering the right people opportunities to control processes in new, more convenient ways.

After a brief description of viewdata and its conventional uses the paper examines methods for incorporating the technology into distributed control systems, outlining the new control options this provides, and discusses some of the problems to be overcome. Finally there is a brief look into the future, and the emerging standards for more advanced viewdata equipment.

VIEWDATA

Viewdata was pioneered by the British Post Office (Fedida (1)) and is now an established technology (Fedida & Malik (2)) available, with some variations, in most countries of the western world. Three main components are identifiable.

The Viewdata Terminal. The viewdata terminal is based on the domestic TV set, with the addition of a keyboard, logic and memory to display text and graphics, and the connection to the telephone network. With many manufacturers offering terminals an increasingly wide range of features are available, with the more expensive devices incorporating microcomputers in their own right and tentatively being called 'teleputers'.

The Telephone Network. From domestic premises only the Public Switched Telephone Network (PSTN) is available, but any private telephone network or PABX system may also be used, and a viewdata terminal can be connected wherever there is a telephone.

The Viewdata computers. The publicly available viewdata computers, primarily offering an electronic publishing service, are the cheapest to reach, but any computer equipped with appropriate ports and telephone connections can supply a viewdata service. Most public viewdata services will also supply - at a price - a through connection to a privately operated computer, and can thus act as a data transport service.

A wide and rapidly expanding range of services is becoming available through standard and very cheap devices, easily installed in offices and homes. Among the services can be access to process control systems.

VIEWDATA CONTROL SERVICES

The concept of a central control room, continuously manned by trained operators who are familiar with sophisticated presentation and control facilities is adequate for the normal control of most processes. Often the control room is of such importance that expensive technology such as high resolution VDU's and custom designed control desks and panels can be employed.

During exceptional moments in the life of a process however these facilities may be inadequate. During start up for example, or when part of the process or control system is behaving abnormally, a large number of people - managers, maintenance technicians, design engineers and others - who are less familiar with the control system operating procedures, may also require information and some decision implementation power. Even if not all the interested people are able to get to site a control room can become very crowded and providing adequate facilities for visitors has, up to now, been prohibitively expensive.

The provision of viewdata terminals for these occasional users and the addition of one or more viewdata ports to the control system permits access by any authorised user from any convenient location and a range of new services become economically feasible.

Management Information. Summary and overview pages provide supervisory and management staff with access to up to date and accurate information about the status of the process at any time, and without the need to disturb the normal operating personnel who, at critical moments, may be hard to reach.

Diagnostic Information. The information required for diagnosis of abnormal process or control system behaviour is often of a level of detail or a complexity not normally of interest to control room personnel. Such information may be comprehensible only to a limited number of off site specialists. Diagnostic viewdata pages allow these specialists, once alerted, to select and obtain the information they need and implement and monitor corrective action without visiting the site and with minimal delays.

Remote Control Services. A process left running without on site supervision and controlled by a system providing viewdata access can be checked and adjusted by control and supervisory staff from their homes.