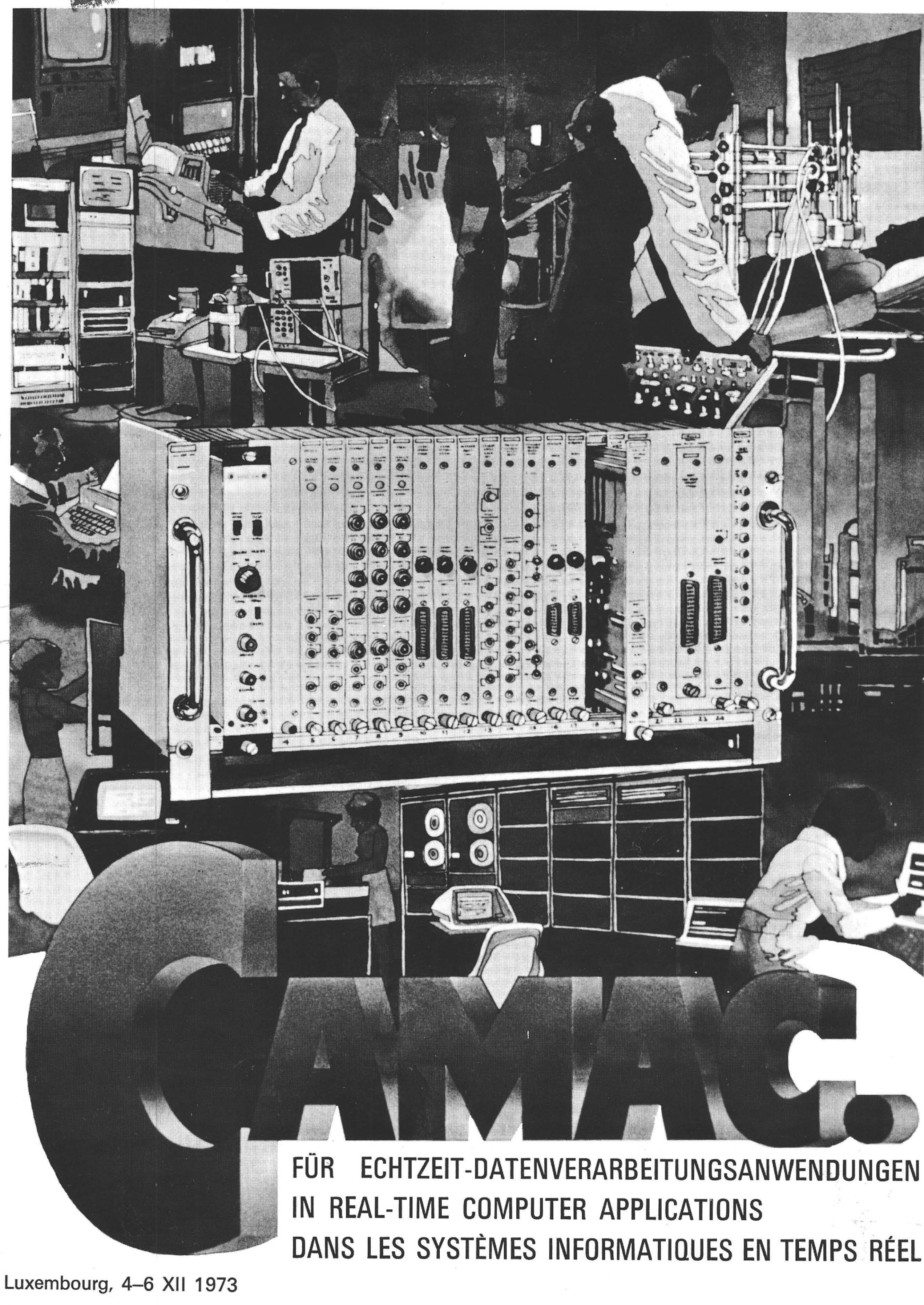
1st INTERNATIONAL SYMPOSIUM ON

Supplement to
CAMAC Bulletin

CAMAC

proceedings

Issue 9
April 1974



FÜR ECHTZEIT-DATENVERARBEITUNGSANWENDUNGEN
IN REAL-TIME COMPUTER APPLICATIONS
DANS LES SYSTÈMES INFORMATIQUES EN TEMPS RÉEL

Luxembourg, 4–6 XII 1973

ERSTES INTERNATIONALES SYMPOSIUM ÜBER
CAMAC
FÜR ECHTZEIT-DATENVERARBEITUNGS-ANWENDUNGEN

Luxemburg, 4.-6. Dezember 1973

PREMIER SYMPOSIUM INTERNATIONAL
CAMAC
DANS LES SYSTÈMES INFORMATIQUES EN TEMPS RÉEL

Luxembourg, 4-6 décembre 1973

FIRST INTERNATIONAL SYMPOSIUM ON CAMAC IN REAL-TIME COMPUTER APPLICATIONS

Luxembourg, 4-6 December 1973

Veranstaltet von / Organised by / Organisé par :

KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN (KEG)
COMMISSION OF THE EUROPEAN COMMUNITIES (CEC)
COMMISSION DES COMMUNAUTÉS EUROPÉENNES (CCE)

in Zusammenarbeit mit / in collaboration with / en collaboration avec :

ESONE Komitee
ESONE Committee
Comité ESONE

Symposium Programme Committee

*W. Attwenger, SGAE Wien, Österreich
W. Becker, CCR-EURATOM Ispra, Italia
*H. Bisby, AERE Harwell, U.K.
*P. Christensen, AEK Risø, Danmark
*P. Gallice, CEA Saclay, France
H. Halling, KFA Jülich, Deutschland
I. Hooton, AERE Harwell, U.K.
F. Iselin, CERN Genève, Suisse
K. Müller, KFA Jülich, Deutschland
*H. Meyer, BCMN-EURATOM Geel, Belgique
*H. Trebst, Universität Erlangen-Nbg, Deutschland
P.C. van den Berg, RCN Petten, Nederland (†)

*Members of Symposium Action Group

Proceedings Editorial Group

W. Attwenger, SGAE Wien, Österreich
R. Barnes, AERE Harwell, U.K.
P. Christensen, AEK Risø, Danmark
P. Gallice, CEA Saclay, France
A. Lewis, AERE Harwell, U.K.
A. Katz, CEA Saclay, France
H. Meyer, BCMN-EURATOM Geel, Belgique
K. Müller, KFA Jülich, Deutschland

Scientific Secretary

H. Meyer, Commission des Communautés Européennes
BCMN-EURATOM Geel, Belgique

Picture on Front Cover by: Fred Preston

PROCEEDINGS
OF THE FIRST INTERNATIONAL SYMPOSIUM ON
CAMAC
IN REAL-TIME COMPUTER APPLICATIONS

Luxembourg, December 4-6, 1973

- **INTRODUCTION TO:**
  - — HARDWARE AND SOFTWARE FEATURES OF CAMAC
- **APPLICATION OF CAMAC TO:**
  - — AUTOMATION OF LABORATORY INSTRUMENTATION
  - — AUTOMATION IN MEDICAL AND HEALTH SERVICES
  - — MEASUREMENT AND CONTROL IN INDUSTRY

Proceedings published by the:
Commission of the European Communities
Directorate General Scientific and Technical
Information and Information Management
29, rue Aldringen
Luxembourg

FOREWORD

More than 500 engineers, scientists and management representatives from laboratories, industry and the medical area in 24 countries attended the First International Symposium on CAMAC in Real-Time Computer Applications in Luxembourg, Dec. 4-6, 1973.

The main intention of the organizers was that the Symposium should contribute to the initiation of new CAMAC uses in different areas of application. For that reason introductory presentations were included for those participants who were not familiar with the CAMAC idea and with the potentialities of CAMAC in general.

Other presentations were devoted to specific areas of activity where automation is opening up possibilities for rationalization, as in laboratories, in industry and in medical and health services. Here the primary idea was to demonstrate that CAMAC is being applied or considered for uses in several areas other than the nuclear field, in order to encourage people to consider CAMAC for the solution of their problems. The first CAMAC Symposium was only to some extent comparable with a conference where experts in a field exchange experiences and are able to consider specific problems in depth. The next Symposium on CAMAC which could take place in autumn 1975 is expected to provide more opportunities for handling such topics.

These Symposium Proceedings will be distributed widely in order to introduce CAMAC to other potential users who were not able to attend the meeting.

Let me use this opportunity to thank again the services of the Commission of the European Communities, all authors of papers, all those who participated in the discussions, the exhibitors of CAMAC equipment and systems, and the representatives of the ESONE Committee. Their contributions to the preparation, organization and content of the Symposium provided the preliminary conditions for its obvious success.

Special acknowledgements are also due to those of my ESONE colleagues and to the staff of the Directorate General XIII of CEC who participated in the preparation of these proceedings.

H. Meyer

VORWORT

Mehr als 500 Ingenieure, Wissenschaftler und Vertreter des Managements aus Laboratorien, der Industrie und dem medizinischen Bereich in 24 Ländern nahmen teil am ersten Internationalen Symposium über CAMAC für Echtzeit-Datenverarbeitungsanwendungen in Luxemburg vom 4. bis 6. Dezember 1973.

Die Anregung zu neuen CAMAC Anwendungen in verschiedenen Bereichen war das Hauptanliegen der Organisatoren mit dem Symposium. Deshalb wurden einführende Vorträge für solche Teilnehmer vorgesehen, welche CAMAC und die durch die Verwendung gegebenen Vorteile noch nicht kennen.

Andere Vorträge waren bestimmten Aufgabenbereichen in Laboratorien, der Industrie sowie in der Medizin und für Gesundheitsdienste gewidmet, für welche Automation eine Rationalisierung von Arbeitsmethoden erlaubt. Das Hauptziel dabei war aufzuzeigen, daß CAMAC in verschiedenen nichtnuklearen Bereichen schon benutzt wird oder eine Anwendung erwogen wird, um Vertrauen in CAMAC für die Lösung von Problemen zu wecken. Das erste CAMAC Symposium war nur begrenzt vergleichbar mit solchen Konferenzen, auf welchen Fachleute eines Fachbereichs Erfahrungen austauschen und spezifische Probleme mehr im einzelnen erörtern. Das nächste Symposium, welches wahrscheinlich im Herbst 1975 stattfinden wird, sollte mehr Gelegenheit dazu geben.

Diese Symposium-Berichte sollen eine weite Verbreitung finden, um CAMAC auch solchen potentiellen Anwendern nahezubringen, welche nicht an dem Symposium teilnehmen konnten.

Die Veröffentlichung dieser Berichte erlaubt es mir, noch einmal den Diensten der Kommission der Europäischen Gemeinschaften, allen Autoren von Beiträgen, allen Diskussionsteilnehmern, den CAMAC Ausstellern sowie den Vertretern des ESONE Komitees zu danken. Ihre Beiträge zur Vorbereitung, Organisation und zum Inhalt des Symposiums waren die Grundvoraussetzung für den erzielten Erfolg.

Besonderer Dank gilt auch denjenigen meiner ESONE Kollegen und den Bediensteten der Generaldirektion XIII der KEG, welche sich an der Vorbereitung dieser Berichte beteiligten.

H. Meyer

PRÉAMBULE

Plus de 500 ingénieurs, scientifiques et représentants des laboratoires, de l'industrie ainsi que du secteur médical, en provenance de 24 pays, ont participé au premier Symposium International „CAMAC dans les Systèmes Informatiques en Temps Réel" de Luxembourg (4 au 6 décembre 1973).

Le but principal des organisateurs était de faire connaître par ce symposium des nouvelles utilisations de CAMAC dans différents secteurs d'application. C'est dans cette perspective que des exposés introductifs avaient été inclus à l'intention des participants non familiarisés avec l'idée CAMAC et les possibilités de CAMAC en général.

D'autres exposés étaient consacrés à des secteurs d'activité spécialisés où l'automatisation apporte des possibilités de rationalisation tels que les laboratoires, l'industrie, et les activités médicales. L'idée fondamentale était de démontrer que l'utilisation de CAMAC est actuellement effective, ou envisagée dans plusieurs domaines autres que le secteur nucléaire, et peut être prise en considération pour la solution des problèmes posés. Le premier Symposium CAMAC fut, dans une certaine mesure, comparable à une conférence où des experts dans un domaine d'activité, échangent leur expérience et ont la possibilité d'approfondir des problèmes particuliers. Le prochain Symposium CAMAC, qui pourrait se tenir en automne 1975, devrait procurer une meilleure occasion de traiter de tels sujets.

Ces annales seront largement diffusées pour permettre l'introduction de CAMAC auprès d'autres utilisateurs potentiels qui n'ont pas pu assister à ce Symposium.

Permettez-moi de saisir cette occasion pour remercier à nouveau, les Services de la Commission des Communautés Européennes, tous les auteurs, tous ceux qui participèrent aux tables rondes, les exposants de matériels et systèmes CAMAC ainsi que les représentants du Comité ESONE. Leur contribution à la préparation, l'organisation, et aux conférences du Symposium ont procuré les conditions élémentaires d'un succès évident.

Une reconnaissance particulière est due également à ceux de mes collègues du Comité ESONE et à l'équipe de la Direction Générale XIII de la CCE qui ont participé à la préparation de ces annales.

H. Meyer

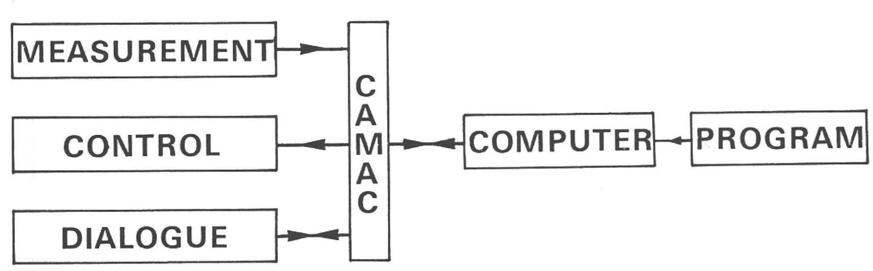
WHAT IS CAMAC?

CAMAC is the designation of rules for the design and use of modular electronic data-handling equipment. The rules offer a standard scheme for interfacing computers to data transducers and actuators in on-line systems. The aim is to encourage common practice and compatibility between products (both hardware and software) from different sources and for uses in different application areas.

CAMAC was originally defined by the ESONE Committee, a multinational inter-laboratory organisation of data-processing experts from nuclear institutes. However, CAMAC is concerned with data-handling problems that are not specific to nuclear research and is being applied already in many other fields.

Working groups of the ESONE Committee are considering further hardware and software aspects of systems for measurement and control, and maintain close liaison with similar working groups of the USAEC NIM Committee and also with International Electrotechnical Commission.

CAMAC is a non-proprietary specification which can be adopted and used free of charge by any organisation and without any form of permission, registration or licence action.



WAS IST CAMAC?

CAMAC ist ein Vorschriftenwerk für den Entwurf und den Einsatz modularer elektronischer Datenverarbeitungsgeräte. Seine Regeln ermöglichen die Anwendung einer Standardschnittstelle für den Anschluß von Rechnern an Datenabgabeeinheiten und Befehlsübernahmeeinheiten in on-line Anlagen. Die Vorschriften sollen die einheitliche Handhabung und die Kompatibilität von Erzeugnissen (Hardware und Software) verschiedenen Ursprungs und für verschiedene Einsatzbereiche fördern.

CAMAC ist ursprunglich vom ESONE Komitee, einem multinationalen Zusammenschluß von E.D.V. Fachleuten aus Kernforschungsinstituten, definiert worden. Die Anwendbarkeit von CAMAC beschränkt sich aber nicht auf DV-Probleme welche für die Kernforschung spezifisch sind; CAMAC wird bereits in vielen anderen Bereichen verwendet.

Arbeitsgruppen des ESONE Komitees untersuchen weitere Hardware- und Software-Aspekte von Systemen mit Mess- und Steuerfunktionen. Sie halten dabei engen Kontakt mit den zuständigen Arbeitsgruppen des NIM-Komitees der USAEC und auch mit der Internationalen Elektrotechnischen Kommission (IEC).

Die Anwendung der CAMAC Spezifikationen ist frei und setzt keine Lizenz oder eine andersartige Erlaubnis voraus.

QU'EST-CE QUE CAMAC?

CAMAC désigne un ensemble de règles pour la conception et l'utilisation d'un système électronique modulaire de traitement de l'information. Il définit un modèle standard de connexion des calculateurs aux organes d'acquisition de données et aux organes de commande dans les systèmes en ligne. Son objectif est de promouvoir l'utilisation de techniques communes et d'assurer la compatibilité entre des produits (tant hardware que software) d'origines différentes et utilisables dans différents secteurs d'activités.

CAMAC a été élaboré par le Comité ESONE, organisme multinational regroupant des spécialistes en traitement de l'information appartenant à des instituts de recherche nucléaire. Toutefois, son utilisation n'est pas limitée au secteur nucléaire, mais est déjà effective dans de nombreux autres domaines.

Des groupes de travail du Comité ESONE étudient des aspects complémentaires, hardware et software, de systèmes de mesure et de contrôle, en liaison étroite avec les groupes correspondants du Comité NIM de l'USAEC ainsi qu'avec la Commission Electrotechnique Internationale.

Les spécifications CAMAC ne sont pas brevetées; elles peuvent être adoptées et utilisées gratuitement par tout organisme sans aucune permission, enregistrement ou licence.

INHALTSVERZEICHNIS

CONTENTS

TABLE DE MATIÈRES

ERÖFFNUNGSSITZUNG

OPENING OF SESSION

SÉANCE D'OUVERTURE

I

EINFÜHRUNG

INTRODUCTION

INTRODUCTION

Part 1 : Chairman, K. Zander

Part 2 : Chairman, M. Mall

II

SOFTWARE FÜR CAMAC DV ANWENDUNGEN

SOFTWARE FOR CAMAC COMPUTER APPLICATIONS

SOFTWARE DANS LES SYSTÈMES CAMAC AVEC CALCULATEUR

Chairman, I.C. Pyle

III

AUTOMATION IN LABORATORIEN

AUTOMATION OF LABORATORY INSTRUMENTATION

AUTOMATISATION DE L'INSTRUMENTATION DE LABORATOIRE

Chairman, P. Gallice

MESSEN, STEUERN, REGELN IN DER INDUSTRIE

MEASUREMENT AND CONTROL IN INDUSTRY

MESURE ET CONTRÔLE INDUSTRIELS

Chairman, R. PATZELT

V

AUTOMATION IN MEDIZIN UND GESUNDHEITSDIENST

AUTOMATION IN MEDICAL AND HEALTH SERVICES

AUTOMATISATION DES SERVICES MÉDICAUX

Chairman, R.S. LARSEN

SCHLUSS-SITZUNG

CLOSING SESSION

SESSION DE CLÔTURE

ERÖFFNUNGSSITZUNG

OPENING SESSION

SÉANCE D'OUVERTURE

BEGRÜSSUNGSANSPRACHE - WELCOME ADDRESS
DISCOURS DE BIENVENUE

J. Lannoy

Director, General Directorate Scientific and Technical Information and Information Management, CEC

Ladies and Gentlemen,

It is a pleasure for me to welcome you here in Luxembourg for the 1st International Symposium on CAMAC in Real-Time Computer Applications.

The Commission of the European Communities has organized this symposium in collaboration with the ESONE Committee.

The ESONE Committee is an inter-laboratory organisation of representatives of European nuclear research centres.

Ten years ago, Dr. GIANNELLI, Head of Electronics at the research establishment at Ispra, encouraged a movement towards compatibility and interchangeability of electronic equipment in all the nuclear laboratories in the European Community. The outcome was a definition, by the ESONE Committee of the "European Standard of Nuclear Electronics" in a EURATOM report, published in 1964.

Since 1964, other laboratories have joined the ESONE Committee. The enlarged Committee has initiated the CAMAC specifications as you might know already, first for applications within their own institutes to have useful methods for an improved handling of measurement and control tasks by the automation of procedures.

Later on the idea was to encourage the spread of the proven methods into areas other than nuclear research for the promotion of automation in scientific research in general but also f.i. in industry and in public services.

This idea was well in agreement with the policies of the Community, including its general policy in favour of dissemination of scientific and technical information in Europe. The Community decided in 1971 to reinforce this general sector of activity because of the ever growing dependence on the availability of a broad spectrum of information well handled and adapted to reach the potential user.

For instance, industry relies on information to rationalize technologies and to improve products for a better competition.

All public services depend on useful information as one preliminary condition to improve their operations. Also the growing research activities in many different areas rely especially on suitable methods of information handling and evaluation to overcome the problem of rational information selection.

I believe that the availability of efficient documentation systems is of great importance for the Commission in order to support the existence of a Common Market for Information, especially by coordinating activities.

But one should not forget that personal contacts and exchange of experiences as it will be possible during the following meeting can not be replaced completely by other methods of information exchange.

For this first Symposium on CAMAC I wish you good success and I can assure you that the Commission will be anxious to support also in future initiatives for CAMAC promotion.

INTRODUCTORY REMARKS

B. Rispoli
Chairman of the ESONE Committee 1972/73

Ladies and Gentlemen,

As Chairman of the ESONE Committee, I have noted with pleasure the great interest shown in this Symposium. I would like to thank very much first of all the C.E.C. for organizing the Symposium in collaboration with our Committee.

Special mention and thanks should also be directed to the services of the Directorate General Scientific and Technical Information and Information Management for the local organization here in Luxembourg together with the Conference services and particularly the General Directorate for Industry and Technology for their generous financial support.

This total C.E.C. support has been an important stimulus for the ESONE Committee members who have been actively engaged in the preparation of this symposium and for those who are to present papers.

Historial Review

By 1959 the NUTRA standard for transistorized equipment had been developed by collaboration between the Euratom laboratory at Ispra and the French laboratories at Saclay and Grenoble.

German standardisation work had proceeded independently, and in spring 1961 the "Deutsche Studiengruppe für Nukleare Elektronik" joined in discussions with the NUTRA people.

In the United Kingdom there had been an independent development of the Harwell 2000 Series standard, then mainly for electron tube equipment.

However, because electronic systems had been very rapidly developing in the recent past years, following the tremendous developments of new electronic components and materials, the work on standardization in a technological fascinating field became very difficult and could not achieve any real success without collaboration between people with similar interests, requiring the specification of a system for their own work as well as for others.

In July 1961 there was a conference at Ispra, the ESONE Committee was founded, and the first working text of the ESONE standard was defined. The Committee is an association of representatives from institutes, mainly nuclear research centres financed by governments. It was created to define common specifications of electronic equipment needed for applications in these institutes. Common multinational specifications were desirable because of the collaboration between institutes for performing joint experiments and in order to stimulate manufacturers to offer equipment well adapted to their needs and to promote the availability of inexpensive equipment by supporting mass production of equipment often needed only in small quantities by a single institute.

The founders of the Committee, i.e. nuclear laboratories from France, Italy, Germany and the research centres of Euratom in Ispra, Italy and Geel, Belgium, named these specifications "ESONE" - European Standard on Nuclear Electronics.

Discussions resulted not only in a well defined modular system, but also an improvement in knowledge and experience for the participants.

In the meantime the USAEC NIM Committee generated the Nuclear Instrument Module System which followed a similar philosophy to the ESONE and 2000 Series standards. However the NIM standard received substantial support from industry, and this adversely affected the growth in production of ESONE equipment.

But already around 1965 the growing needs for measurement and analysis of data became enormous; also the first small computers for on-line uses were offered and the institutes could not ignore, any longer, the availability of these computers.

In consequence the definition of CAMAC was started in Europe by the ESONE Committee. For CAMAC, the modular concept was again applied but really a new system philosophy had to be followed in order to interface many measurement sources to a computer and to exert program control on-line.

The name "CAMAC" was originally not an abbreviation like "ESONE". The bidirectional readability of the word CAMAC symbolises the bidirectional flow of data in CAMAC systems. Today the name is often explained by the expression Computer Aided Measurement And Control. Mainly as a consequence of the close collaboration with the NIM Committee in the USA, since 1968, CAMAC has become a world-wide standard during the last few years. This collaboration during which the NIM Committee largely influenced the choice of certain key characteristics of the CAMAC specifications led to the endorsement by the NIM Committee of CAMAC specifications; EUR 4100 in February 1970 and EUR 4600 in November 1971.

The 1969 version of EUR 4100 was published in four languages, English, French, German and Italian. This involved much work, became to a certain extent translation also means interpretation.

Since then the Committee has continued and increased its activities by extending and refining specifications on hardware and putting effort into the consideration of the software aspects of CAMAC, as will be explained during the following sessions.

ESONE Organisation

Intense activity by member organisations of the ESONE Committee had been essential in the early stages of formulating CAMAC. By 1968 the number of members had increased and it was essential to have a Committee structure which would ensure the continuity of CAMAC and at the same time have the minimum amount of rules and regulations. The resulting structure and activities are shown in Fig. 1. The primary aims are to formulate, recommend and promulgate practices in areas where

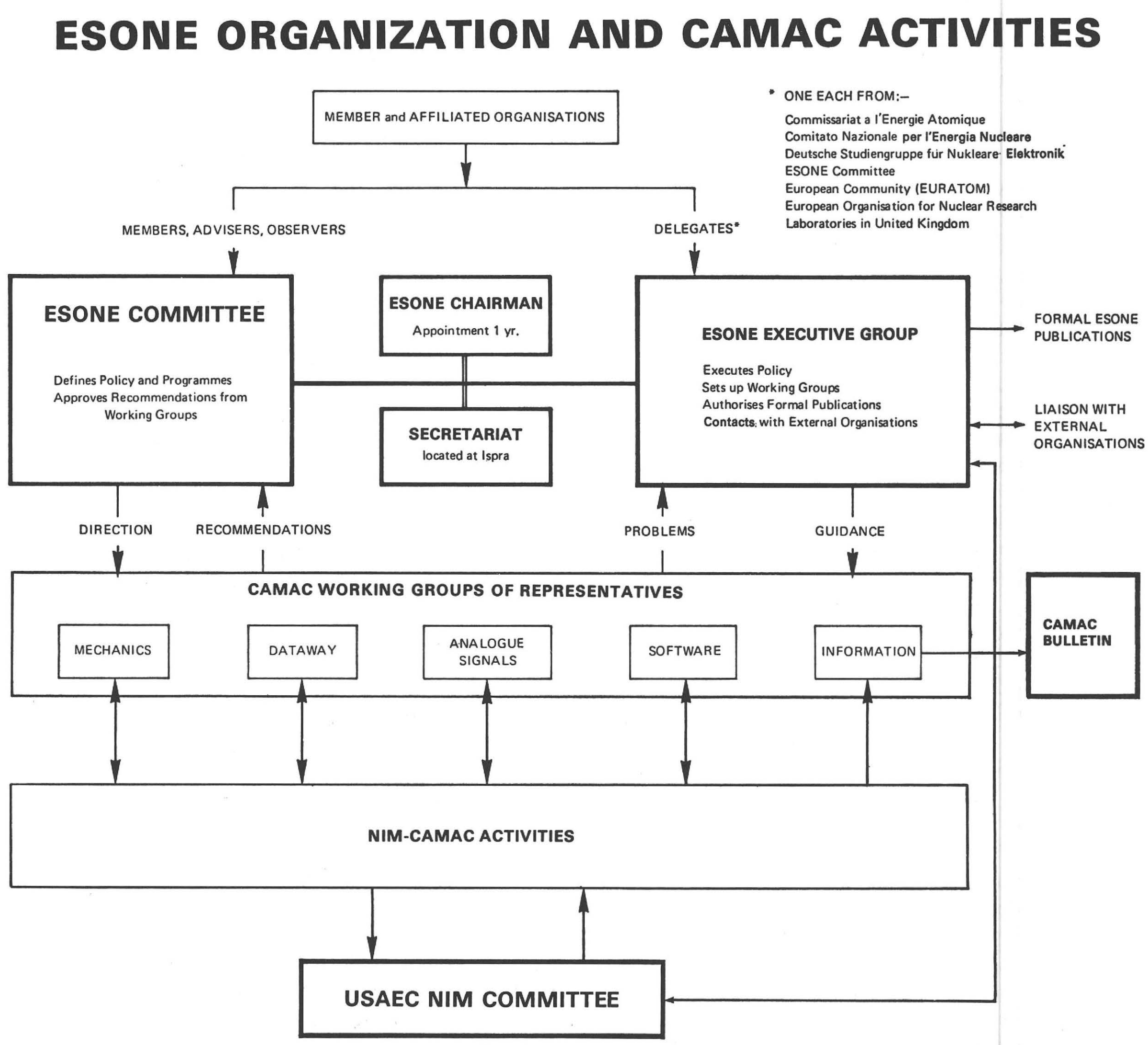


Fig. 1

international standards do not exist, for application in the design and operation of electronic equipment with a view to encouraging interchangeability and therefore common use of equipment and practices. Collaboration with formal International Standardisation Organisations is a keynote of the Committee's policy. The work programme of the Committee is carried out by Working Groups that receive guidance and report back to the full ESONE Committee. The Executive Group is responsible to the Committee for the management of ESONE interests.

Extension of CAMAC Application Areas

The Information Working Group, mentioned on the figure, is the most recent activity of the ESONE Committee.

Gradually over the last few years the ESONE and NIM laboratories have become aware of the universal applicability of their CAMAC specifications and consider that a wider application of CAMAC could promote technological progress in many areas and also utilize the enormous financial investment made during the CAMAC development by national governments. But the introduction of CAMAC into new fields was another type of multinational activity far beyond the scope of most laboratories and even the ESONE Committee as a free association of laboratory representatives with no official status. Support was therefore sought from CEC as an official multinational body. First of all it appeared necessary to disseminate as much information on CAMAC as possible not only to the growing number of CAMAC users but also to potential users. The outcome was the publication of the CAMAC Bulletin in early 1971 by the ESONE Committee in collaboration with, and financially supported by, the CEC.

5000 copies per issue are distributed at the moment, and copies of Issue No. 8 are available at this Symposium.

One year ago the Commission of the European Communities agreed to support the promotion of CAMAC in a second way: by organising meetings with potential users and the first attempt has resulted in this Symposium. The idea is to allow, after a general consideration of the characteristics of CAMAC, some deeper understanding of its potential by a description of applications in fields familiar to the participants of the Symposium. I sincerely wish you to enjoy this Symposium and hope that, in consequence, there will be an increase of CAMAC applications and users.

THE PROMOTION OF AUTOMATION IN INDUSTRY, RESEARCH AND TECHNOLOGY BY THE COMMISSION OF THE EUROPEAN COMMUNITIES

C. Layton

Director, General Directorate for Industry and Technology, CEC

My first task is to pass on to you the personal good wishes of Commissioner Spinelli for the success of this congress. As Commissioner responsible for industrial and technological affairs and the initiator, last year, of the Community's first proposals for a common science and technology policy, now the responsibility of R. Dahrendorf, A. Spinelli takes a great interest in the work of your congress, and wishes it success. R. Dahrendorf the Commissioner responsible for science, education and research, and also of the Directorate General Scientific and Technical Information and Information Management, which has organised this congress, also, I know, wishes you well in the very practical work you have to undertake this week.

The congress takes place in a climate, in European affairs, which has considerably evolved in the last two years. A year ago, the summit conference spelt out a major new series of goals for the Community — the development, in particular, of common policies for industry, science and technology.

During the past year, the Commission has followed up that mandate by making a series of specific proposals in these fields. In May, we sent to the council an outline for a programme of action on industrial and technological policy. In August, the action programme for science and technology policy was put forward, if implemented, it means the setting up of permanent machinery to coordinate national policies and develop common actions. Two weeks ago, the Commission put forward first proposals for a pooling of resources in the field of data-processing. These cover joint work on applications, coordination of policy in the field of standards, and procurement, and a real effort to strenghten the European based industry by collaboration between the major companies.

The work of this congress and the effort made by a small group of people to spread the CAMAC gospel in recent years fit in closely with our aims in data processing and wider industrial policy. A central Community objective is to ensure that the full potential of modern data processing capabilities is applied to users needs in the Community.

A second objective is to make markets for equipment as open and transparent as possible through standardisation. A third related aim is to give manufactures the possibility of achieving higher productivity through economies of scale.

All these general aims will be served indirectly by your congress. The CAMAC club has been a spontaneous creation by users, arising from the needs of scientific laboratories for a standard modular system for instrumentation and process control. In their case, it has already demonstrated that it can bring important operating economies and enable manufactures to provide economical equipment. The Community has supported this work in the past through the publication of the CAMAC Bulletin and the modest but essential support provided by the Joint Research Centre we have given this support not least because in the development of a standard, or in international collaboration generally, it makes much more sense to support an activity for which users have clearly indentified a need and competent and able people are already at work, than to try to promote blue-prints from outside.

This is valuable too however that—the CAMAC system—now seems to have reached a critical phase in its development, characteristic of many innovations. CAMAC systems clearly have relevance to a growing range of applications, as the papers on medical applications show. Nor should the wider industrial potential of CAMAC systems be ignored.

Inevitably, however, users ask many questions before they embark on the application of a new modular standard, and though it is possible to demonstrate the potential economic benefits of applying CAMAC systems, new users often prefer others to do the demonstrating. The user carries extra costs of adaptation if he is a pioneer. Or he may simply be unaware of the potentialities of the system.

Meanwhile, the potential manufacturers of CAMAC equipment may be reluctant to embark on new applications until they see that the standard is likely to be widely adopted.

The essence of the industrial innovation process is of course that innovators perceive future markets needs, develop products to meet them and persuade the potential users to buy. But when the innovation consists of a standard whose use is spreading gradually, there is a more complex interaction between the spread of utilisation and the possibilities for innovating manufacturers to develop the sale and output of equipment.

In the Commission's view, the spread of the CAMAC standard to new fields could, potentially, raise productivity in a number of fields of application, and at the same time provide a more open and transparent market. The aim of this congress is to encourage this spread: by acquainting new possible users with the potential of the CAMAC system, by confirming this from past experience, and by examining ways in which it can be applied in new fields, by showing manufacturers potential areas in which they may usefully develop new products, by enabling manufacturers and users to examine all these possibilities together.

We hope that as a result of this congress it may be possible to see what further steps need to be taken to bring users of CAMAC systems together and support the further spread of its applications.

The essential work has to be done by you, but the Commission is a benevolent friend.

I

EINFÜHRUNG
INTRODUCTION
INTRODUCTION

Teil 1 - Part 1 - Partie 1

Präsident - Chairman - Président:

K. ZANDER, HMI Berlin, Germany

Teil 2 - Part 2 - Partie 2

Präsident - Chairman - Président:

M. MALL, Dornier GmbH, Friedrichshafen, Germany

THE FUNDAMENTAL PRINCIPLES OF CAMAC

H. Bisby
Electronics and Applied Physics Division, Harwell Laboratory,
Harwell, United Kingdom

ABSTRACT

Some aspects are reviewed of the methods of interconnecting a computer with the physical world of transducers, actuators and other peripherals. The CAMAC interface standard is a solution to the problems that arise and its design principles are examined.

ZUSAMMENFASSUNG

Es wird eine Übersicht über einige der Gesichtspunkte gegeben, die bei der Zusammenschaltung eines Computers mit seiner realen Umwelt von Messwerterfassungsgeräten, Wirkungsgliedern und anderen Peripheriegeräten zu beachten sind. Die im CAMAC-System standardisierte Schnittstelle ist eine Lösung für die dabei auftretenden Probleme. Ihre Konstruktionsprinzipien werden aufgezeigt und erläutert.

RESUME

Après un examen de quelques aspects des méthodes d'interconnexion entre un ordinateur et les éléments tels que organes d'acquisition et de commande et autres périphériques, on montrera que les normes CAMAC constituent une solution aux problèmes qui se présentent dans ces cas. On en examinera les principes de conception.

1. Introduction

1.1 The use of small dedicated computers in on-line data processing systems for automation of analytical instruments, for the control of industrial plant, for data acquisition in the medical context or elsewhere, encounters problems that are similar and associated with interfacing the physical world outside and computer to the computer itself such that the desired interaction between the two is obtained.

1.2 The external world, Fig. 1, comprises a varied range of 'peripheral devices' that includes transducers for measurement purposes, actuators for control purposes, and devices for communication between the human operator and the system, or between systems. The internal world of the computer contains a digital processor, a store, input/output facilities and a program that may be required to react to stimulii from the external world. The nature of the interface can exert considerable influence on the extent to which the potential of both worlds can be realised in an integrated real-time system.

2. Interfacing Strategies

2.1 In many real-time, on-line applications, computer interfacing is concerned with connecting peripheral devices to the input/output facilities of the computer. These facilities may be classified into the two types of 'programmed I/O' or 'direct store access'.

2.2 Whichever type is used, the computer requires an adaptor to connect it physically, electrically and operationally to the peripheral. This adaptor may be a combination of hardware and software and the hardware will have an interface to the computer and an interface to the peripheral. One of the following solutions to this interfacing problem may be chosen:-

2.2.1 A Standard Option

The computer company may have and offer with the computer its own proprietary 'standard option' whose external interface will suit the peripheral. For example, a 'standard option' could provide a parallel, digital, input or output transfer facility that employs standard TTL signal conventions at the external interface.

2.2.2 A Special Option

The computer supplier, or the user in some cases, may provide adaptor hardware equipment having one interface to the computer's I/O channel and the other interface to the appropriate external peripheral, Fig. 2. The adaptor has to follow the electrical and organisational conventions set by the computer manufacturer and will often be built from hardware similar to that of the computer itself. The adaptor hardware connected to the programmed I/O bus typically includes gates and registers to control the data transfer and, additionally, means by which the peripheral recognises when it is addressed, means for accepting instructions to enable and disable interrupts and means to test the readiness and other status aspects of the peripheral. As with the manufacturer's standard option, the special option adaptor may be shared between a number of peripherals. For example, a typical adaptor for a number of magnetic-tape devices may include various control and status registers appropriate to the devices with instructions to load, read and test them. In the general case, where 'p' different peripherals each with different proprietary interface conditions are to be connected in various systems to 'c' different computers each again with their own proprietary interface conditions, the special option approach of making an adaptor from each peripheral to each computer requires (p x c) different adaptors. Each new peripheral requires 'c' new adaptors; each new computer requires 'p' new adaptors.

2.2.3 An Independent Standard Interface

An alternative method is to introduce an intermediate independent interface standard (Figs. 3, 4 and 5) so that now only 'p' peripheral-standard adaptors and 'c' standard-computer adaptors are needed, i.e. (p + c) in all. For each new computer and each new peripheral the addition of only one new adaptor is required. The intermediate independent standard may be designed for a 'one-to-one' connection(1), as in Fig. 3. Each peripheral device is associated with two simple adaptors linked through the independent standard; the adaptors could be physically part of the peripheral, on the one hand, and part of the computer on the other, but not necessarily so. The independent standard, CAMAC, is designed(2) as a multiplexing interface for many-to-one connections (Fig. 4) so that although each peripheral has an associated adaptor or 'module', the standard also links these to a common computer/adaptor or 'controller'.

2.3 In broad terms and ignoring other differences between these two particular interface standards, the one-to-one standard has an advantage where only a few peripherals are involved and each is connected invidiually to the computer, while the many-to-one interface is more attractive for systems having many peripherals. The two can complement each other in situations where some of the peripherals are fitted with one-to-one adaptors and the rest are interfaced directly via CAMAC.

2.4 The many-to-one type of interface could employ either a continuous highway (as in Fig. 4) having tee-connections to the peripheral adaptors or a chain highway

(Fig. 5) in which the computer adaptor is linked to the first peripheral adaptor, and this in turn is linked to the next adaptor, and so on. The physical arrange-ment of peripherals may dictate the nature of either highway. If the peripherals are close to the computer, the highway could use a multi-wire bus for parallel-bit transmission. If they are more scattered, there may be a case for constructing a number of nodes, each serving a group of peripherals, with a common connection back to the computer. Other important considerations arise if the peripherals are not with the computer in the same suite of cabinets (or same room) but are distributed throughout a large building, factory or process plant, or even more remotely. In these situations line-transmission problems and cable-cost considerations could dictate that continuous and chain highways are different. For example, the continuous highway could be for bit-parallel, word-serial transmission on a multi-wire highway compared with bit-serial byte-serial on a 2-wire chain highway. Clearly, such different interface standards should have data transfer structures that are compatible in order to achieve both equipment and programming benefits.

3. Fundamental Design Principles

3.1 The initial concept of CAMAC was based on certain design principles and these have enabled the CAMAC standard to develop over the years, so that it embraces the best of the interfacing methods.

3.2 These design principles are:-

3.2.1 Peripheral and Computer Independence

The structure, organisation and management of the data transfer methods had to be computer and device independent because of the nature of the multi-national forum in which the need for CAMAC arose. A standard biassed to the proprietary methods of any one computer would have been totally unacceptable not only for nationality reasons but because no one computer family could be expected to be adequate for all the possible applications even in nuclear laboratories. Although many specific applications of CAMAC were clear in the nuclear field, even here the diversity of transducers/actuators was sufficient to warrant a standard that would not be restricted in its field of application by being device dependent.

3.2.2 Anticipation of Developing Technologies

As early as 1964 when the need for CAMAC was clear and components using small-scale integration were just becoming commercially available, there was some doubt as to when large-scale integrated circuit technology would become available. Nevertheless since CAMAC was being designed for major usage in the 1970 decade, it had to anticipate at least the possibility of medium-scale integration and computers having 24-bit word-lengths.

3.2.3 Modular Adaptors for Flexibility

To maximise the user advantages of flexibility in system design, in system configuration and size and of the cost economies arising from commonly available data processing functions[3], the standard had to employ adaptors that were modular in physical format. Thereby a 'library' of modular units, each having specific data processing functions or peripheral tasks could be available with which to build up a systems interface for application in many fields. The physical format had also to take into account the low cost considerations of integrated circuits and printed wiring technologies.

3.2.4 Many-to-One Multiplexing

It was recognised that by the 1970s the small dedicated computer would become predominant in data acquisition and control situations which employ many sources/receivers of data. Because these systems would require many peripheral adaptors in comparison to the fewer computer adaptors, a basic principle was to design CAMAC such that the peripheral adaptors should be simple and inexpensive so that any complexity involved in many-to-one multiplexing, such as addressing decoding, transfer-cycle timing, system-error checking etc., should be concentrated in the computer adaptors.

3.2.5 Efficient Demand Handling

On-line, real-time systems are such that the computer program must be interrupted when a real-time demand occurs in that part of the system external to the computer. This demand may be for either a service routine or for the facilities provided for direct store access. Rapid and efficient demand handling is therefore a fundamental requirement of a system and the interface arrangements must not impose serious delays in identifying the source of a demand.

3.2.6 'Downwards' Compatibility

Over the years it was expected that the CAMAC Standard would be extended to cover additional features and requirements. To safeguard therefore against rapid obsolescence of equipment, the principle was established that the mandatory parts of the CAMAC specifications should apply to only those features which were essential for interchangeability and compatibility and so reduce the risk that, after publication of the first specification, (EUR 4100) any additional specifications for more complex requirements would not render incompatible existing equipment that complied with EUR 4100.

This principle also allowed the specifications to leave plenty of opportunity for many imaginative implementations to make the impact which they deserve, not only on the facilities available in CAMAC, but also on their commercial prospects.

4. General Comment

The design principles of CAMAC have made possible a unique and non-proprietary method of interfacing the computer with the physical world external to the computer in such a way that flexible and economical systems for data transfer can be achieved in measurement and control.

The presentations which follow in this Symposium will expand on why and how by describing CAMAC, its implementations and a selection of the many and varied existing applications.

A standard that has interchangeability and compatibility as its primary aims must be rigidly implemented in all its applications if the full benefits of use are to be realised. This can impose restrictions on the methods of implementation that may render it unsuitable for certain areas of application or, at most, of no outstanding technical or economic merit over other methods available. In what follows therefore I am confident that my colleagues will be clear to point out not only those areas where CAMAC can be used to advantage but also where its very nature may impose restrictions.

References

(1) British Standard Interface BS 4421.

(2) CAMAC Bulletin Issue No. 6 onwards. List of Specifications and supplementary information.

(3) BISBY, H. The Design Principles and Role of a Comprehensive Unit System of Electronic Equipment. JIERE, 29 No. 3, pp. 185-195, 1965.

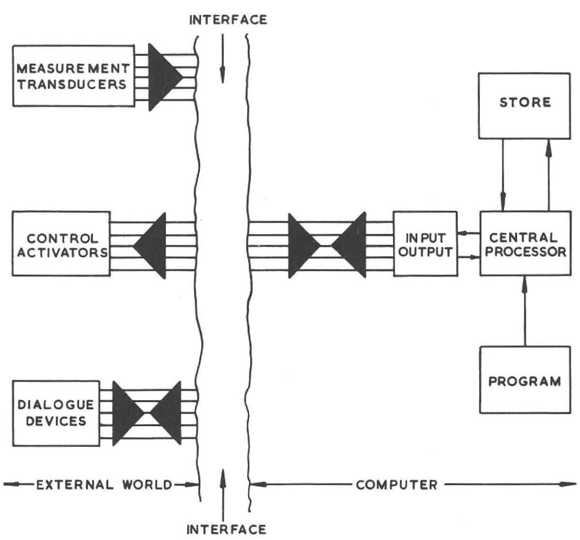


Figure 1. Generalised System Schematic.

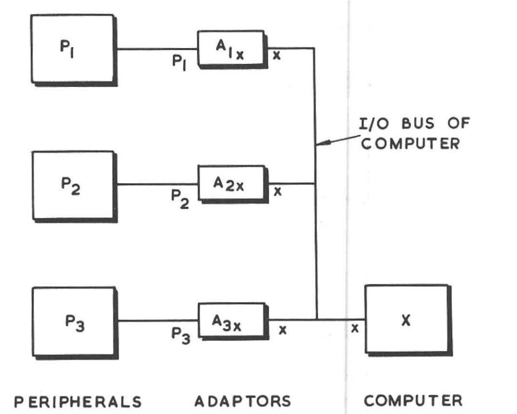


Figure 2.
Interfacing by "Special Option" Adaptors.

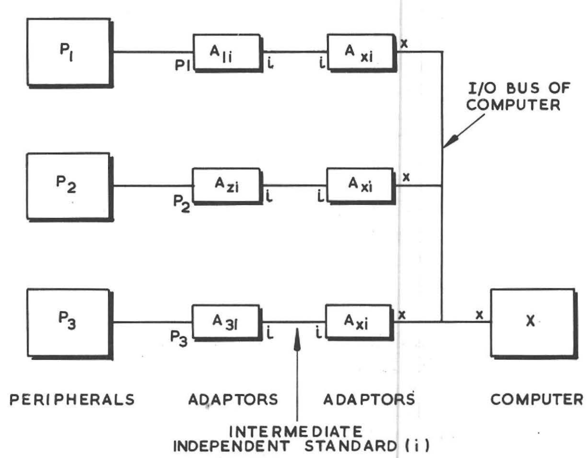


Figure 3. Interfacing through an Intermediate "One-to-One" Standard.

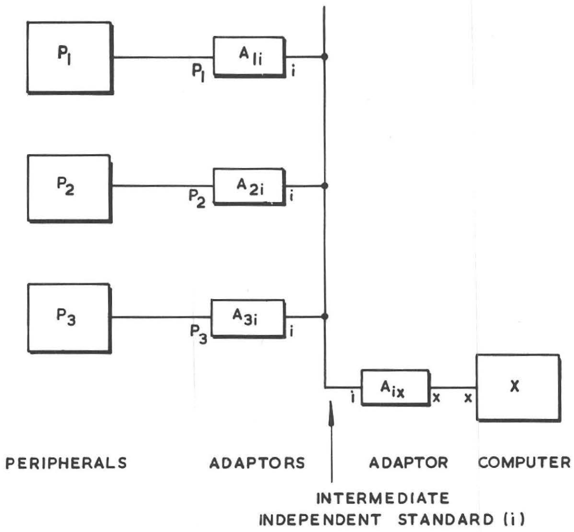


Figure 4. Interface through an Intermediate Many-to-One Standard Highway.

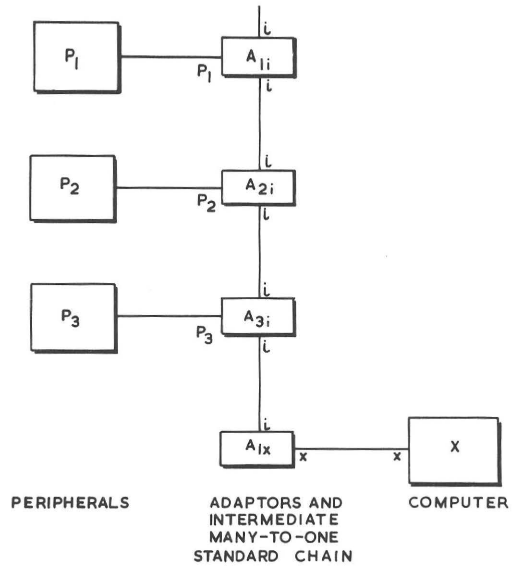


Figure 5. Interfacing through an Intermediate Many-to-One Standard having Chain Connections.

DISCUSSION

Q - R. L'Archeveque

It would appear that a combination of figs. 2 and 4 in your paper would give a more realistic representation of a practical system since generally, a number of peripheral units are available already interfaced to the computer bus by the manufacturer. This is particularly true when a number of DMA channels are needed. In such cases, the use of CAMAC or any intermediate standard leads to a dual bus system.
Would you care to comment on the desirability (advantages/disadvantages) of the single bus system illustrated in your fig. 4 as opposed to the dual bus situation from an overall system concept point of view.

A - H. Bisby

Yes, I would agree with your first comment. It is very simple when one is starting to build up a new system to decide on accepting either the solution of Figure 2 or that of Figure 4. If you already have employed the solution of Figure 2 and wish to expand its capability using CAMAC, one nevertheless has a dual-bus situation. On your second comment, the solution of Figure 4 is the CAMAC solution and one of its major advantages is, that the system user is isolated from or made independent of the computer manufacturer or supplier in many ways. Also attendent with this solution is the whole range of predominantly CAMAC advantages, e.g. multi-sources of supply, wide choice of products, system open-endedness etc. etc.

Q - G. Weijers

Can the panel give an indication of the problems associated with the use of standard CAMAC equipment in an avionics environment (low power available 500 W, low cooling capacity)?
Is it feasible to modify the actual design to make CAMAC modules applicable in future spaceprograms.

A - H. Bisby

There already exists an airborne single-crate CAMAC system that was described in CAMAC Bulletin No. 6 ("An Airborne γ-scintillometer", E.M. Christiansen and P. Skaarup) which demonstrates that CAMAC equipment can be installed in aircraft provided the crate is suitably mounted against vibration etc. Furthermore, a single CAMAC crate can contain modules/controllers dissipating up to 200 watts and this allows a very large system to be used. If you are suggesting that the CAMAC standard should be changed to satisfy the avionics environment, I do not think this would be useful but I see no reason why a series of CAMAC, low-powered, COSMOS modules should not be developed for avionics and for installation in space vehicles.

CAMAC HARDWARE

F. Iselin
CERN, Genève, Switzerland

ABSTRACT

The basic elements and features of CAMAC are described: modules, dataway, crate, etc.

ZUSAMMENFASSUNG

Der grundsätzliche Aufbau und die Eigenschaften von CAMAC Geräten werden beschrieben (Module, Datenweg, Überrahmen, usw.).

RESUME

Description des éléments de base et des principales caractéristiques de CAMAC: modules, interconnexion, châssis, etc.

ABSTRACT:

This paper will describe CAMAC in a very simple and introductory way to help newcomers to get a first idea of the system. The basic reference is EUR 4100e revised 1972 "CAMAC a modular data handling system"[1].

The first major advantage of CAMAC is to provide users with standard instruments (or modules) which may be inserted into a Crate and interconnected in this crate via a so-called Dataway. Many further advantages come out of this standardization such as commercial availability, easy communication between laboratories through their designers of hardware and software.

1. Standard CAMAC instruments (or modules)

Fig. 1 shows a "one unit width" module without side cover. All dimensions for compatibility are mandatory. The main feature is the fact that all signals entering or leaving the module (at the rear connector) are precisely defined in 4100e, making the module compatible with other modules also inserted in the crate.

The module (= instrument!) compatibility - mechanically and logically - is the basic feature of CAMAC. It is realized by the Dataway which links the modules together in the crate in a standard way. The Dataway consists of standard interconnections, mainly bus lines for Address, Data, various controls and two individual lines per plug-in station (Module Address + Call Facility L). The most important signals used in the Dataway are indicated in Fig. 1 which otherwise is self-explanatory. Note the extreme compactness of the module (width 17.2 mm) and the good fit to integrated circuits. Wider modules are simply multiples of the single width module shown.

The front panel makes the required connection to the external equipment: motors, counters, display, etc. (In the module of Fig. 1, counts may be received into 4 parallel scalers of 16 bits at rates above 30 MHz/sec.). Users - or their computer! must generate the necessary commands in view of producing the required action. A Command defines where to execute and what: "Where" is the address C (crate) N (module) A (subaddress) "What" is the Function F to be executed. The data (if any) related to the command is transferred on the - separate - Read/Write lines from or to the computer.

The very important and individual L signal (one per basic module width) allows a module to call the computer for attention, for ex. in case of overflow, alarm synchronisation etc. A proper "conversation" may therefore exist between the computer and the modules, using the L signals in conjunction with the interrupt system of the computer.

2. Standard CAMAC crate (chassis)

Fig. 2 shows modules from various manufacturers, inserted into the CAMAC crate. The crate has 25 slots or stations of which the 2 rightmost stations (at least) are reserved for a "crate controller" (CC). This CC may be type CC-A1 as defined in document (EUR 4600 CAMAC Organisation of Multicrate Systems) in view of providing higher compatibility level (in the control) or it can be a "dedicated CC". The dedicated CC is so designed as to best fit the characteristics of a given computer. It finds increasing applications.

In Fig. 2 only a few modules are inserted to allow a view on the internal Dataway (standard connectors + standard wiring). As an example a Crate Controller A (with no cover) is inserted (Stations 24 and 25).

The Crate Controller has access to the Dataway and to the external world (other crates, computer).

The preceding and following paper treat the various possibilities of connecting crates to the computer (Branch highway with CCA, dedicated CC, serial highway, radial highway). Note the great efficiency and compactness of electronics in one crate (up to 23 users' modules). The power supply is mounted - in this example - on the rear of the CAMAC crate itself.

3. Typical features of CAMAC

- Data, or various control information, may flow from or to the central controller (generally a computer, or any other controller).
- Commands may initiate all sorts of actions required by the installation. (Read, Write, enable, start, measure, move, print, display, record, etc..).
- The speed is up to one million "words"/sec. of up to 24 bits/word.
- The flexibility of CAMAC allows the construction of complex systems, immediately or progressively. Operations may be very fast, interlaced, sometimes autonomous, and may control concurrently a very large number of receivers/transmitters of data and execute multiple control operations.
- The total cost of a typical operational system with one crate and one minicomputer, including some 20 "medium cost" modules and the crate controller, is of the order of magnitude of a designer's salary for one year.

These few parameters - or features - should help in defining the field in which CAMAC can be applied such as typically where computers are organizing complex data transfers and multiple control, often simultaneously with calculations, data filtering and recording. Fig. 3 gives a partial view of the CAMAC equipment in a large experiment at CERN[2].

1) Obtainable from "Office for official publications of the European Communities", case postale 1003, Luxembourg 1 or as TID 25876 report in USA from Louis Costrell, National Bureau of Standards, Washington, D.C. 20234.

2) See Laboratory Automation Session III, paper "A self checking system...".

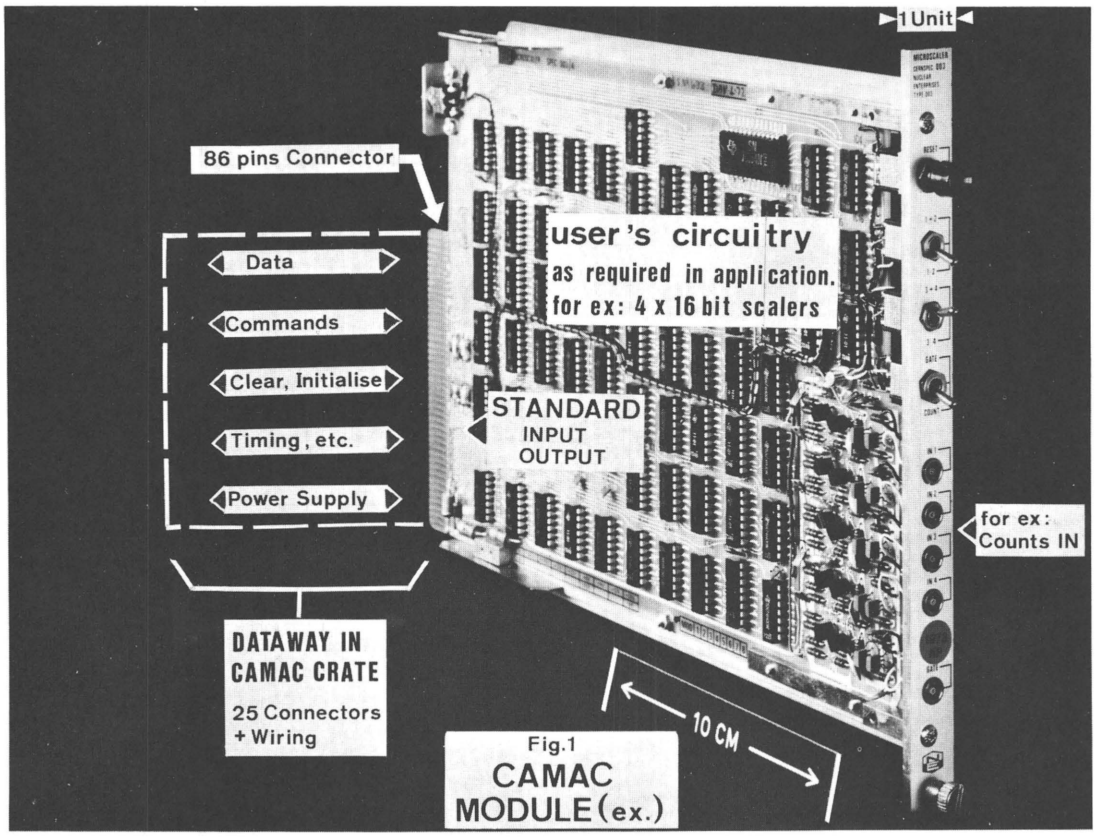


Fig.1
CAMAC
MODULE (ex.)

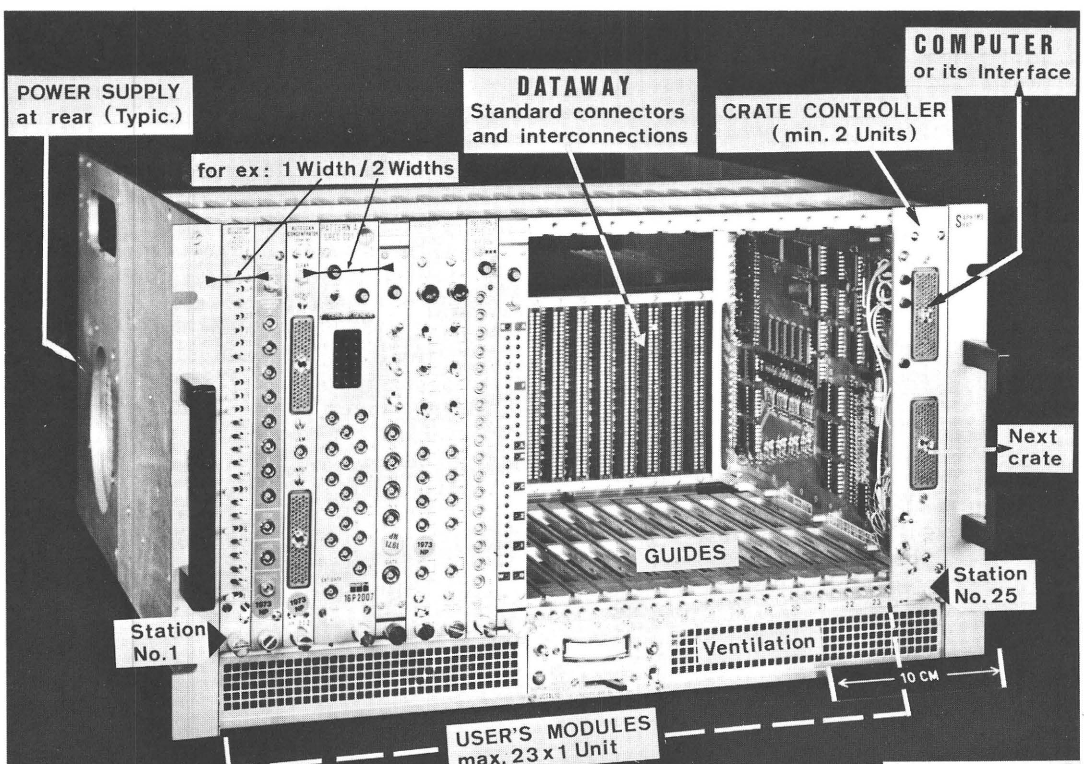


Fig. 2 CAMAC CRATE

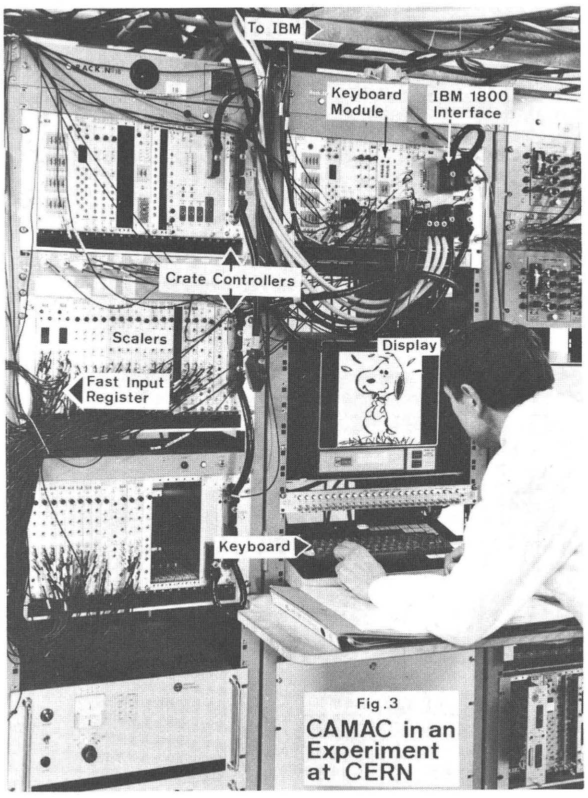


Fig.3
CAMAC in an Experiment at CERN

DISCUSSION

Q - R. L'Archeveque

1) Is the back-mounted crate power supply a necessary standard of CAMAC as opposed to a large remote power supply distributing power to regulators in many crates?
2) Does the back-mounted approach lead to efficient use of the power supply in practice, considering that often the loading on each crate is dictated by the front panel space needed more than the logic capacity of the dataway?

A - F. Iselin

1) Common block power supplies at the bottom of crates are no doubt a good solution and the preference - by users - is probably due to a certain need for independence.
The aim to have NIM and CAMAC systems (or others) powered by identical power supplies may again raise interest in a common power supply at the bottom of the rack.
2) The front panel space situation is not really in contradiction with the individual rear power supply of corresponding size. In fact it seems that the 300 W power supply now to be used for example in CERN (for example with ± 6 V at 30A max.) will represent quite an efficient use of the volume available. I guess this is approximately true for other power supplies too.

Q - N. Des Ordons

Pouvez-vous justifier le choix du format des données (mot de 24 bits) du "Dataway" CAMAC?

A - F. Iselin

The choice of 24 bits was mainly due to an influence from our US colleagues in the very early stage of CAMAC design. ESONE had planned 16 bits but asked for comments - in a very informal way - from our various NIM colleagues. Since the subject was not particularly hot at that moment there was no real discussion about going over to 24 bits since this would anyway cover the 16 bit solution.

Q - J. Waidelich

My question concerns fault recognition within a CAMAC module or a crate. Are there any options in CAMAC modules so that a hardware failure within the module can be recognized, for example by some additional redundant hardware?
What possibilities exist, so that a CAMAC system can be checked on line (that means with the process running)?

A - F. Iselin

Troubleshooting is not specific to CAMAC but applies to all large and complex systems. This problem is not yet entirely solved but is under study and is being improved. One critical point seems to be at the "junction" hardware - software, where it is not always clear in which group the error is. The software should as much as possible have provision for checking the overall system or any other checking facility. Because it is often not the case hardware/software people must be on the spot to provide some support.
It may consist of giving special commands via special subroutines. Often the computer is disconnected and replaced by a special hardware driver to simulate programs that are otherwise too bulky - by a simple group of instructions produced by an easily loaded memory (for example by a card written with pencil).
Other hardware feature are used such as a special "block transfer module" to simulate responses to software, also Branch test boxes have been produced.
A difficulty is the individual N address to modules and here maybe a special CAMAC function ("Test N") could help testing the various N's.
A standard feature of all CAMAC modules, specified in EUR 4100 (1972), is that they generate a Dataway signal, X = 1, to indicate that they are able to carry out each command. Some hardware and software failures are thus indicated by X = 0.

CAMAC SYSTEM CONFIGURATIONS

H. Klessmann
Hahn-Meitner-Institut für Kernforschung GmbH, Berlin, Germany

ABSTRACT

The CAMAC instrumentation system is defined by a precise set of rules in order to facilitate modularity and compatibility. Much freedom is left to allow system designers to arrange CAMAC systems in many configurations in order to accommodate a wide range of different applications and complexity. Typical examples of basic configurations are reviewed.

ZUSAMMENFASSUNG

Das CAMAC-Instrumentierungssystem ist durch genaue Regeln spezifiziert, die einen modularen Aufbau und die Kompatibilität von Geräten ermöglichen. Für den Systementwickler besteht dennoch weitgehend Bewegungsfreiheit, die unterschiedlichsten Systemkonfigurationen mit der CAMAC-Instrumentierung zu verwirklichen, um die Anforderungen bei den verschiedensten Anwendungen mit unterschiedlicher Systemkomplexität zu erfüllen. Typische Beispiele von grundlegenden Konfigurationen werden beschrieben.

RESUME

Le système d'instrumentation CAMAC est déterminé par un ensemble de règles précises, en vue de faciliter la modularité et la compatibilité. Ce système CAMAC laisse une grande liberté d'utilisation aux concepteurs de système pour l'adapter à une large gamme d'applications de compléxités différentes. Des exemples caractéristiques des configurations fondamentales sont passés en revue.

1. INTRODUCTION

Basically, the CAMAC instrumentation system is specified in terms of three different ports. These are (Tab. 1):

(1) The Dataway port, specified in EUR 4100e (1972) [1], which provides data communication between CAMAC modules and a crate controller, within each CAMAC crate.

(2) The Branch Highway port, specified in EUR 4600e (1972) [2], which may be used for standard interconnection of several CAMAC crates to a computer, with bit-parallel transfer of data via a limited distance.

(3) The Serial Highway port, defined in a preliminary ESONE/NIM Description EDWG 25/73/NDWG 73-10 [3], which complements the parallel Branch in applications where simple interconnections of many crates to a computer with bit- or byte-serial transfer of data are particular important, or where long distance is involved [4].

All these ports are defined by a precise set of rules in order to facilitate modularity and compatibility of CAMAC instrumentation. However, a high degree of freedom is left to the system designer so that CAMAC systems can be arranged in many configurations to accommodate a wide range of different applications and complexity.

As shown in Tab. 2, there are at least three very general areas of applications where CAMAC instrumentation is used:

(1) Data acquisition and process control in on-line instrumentation where CAMAC provides the means for communication between transducers or actuators and a computer. Originally, the CAMAC system was intended primarily for this application area to provide a standard method for interfacing all the exotic process devices which may be required.

(2) If CAMAC is used for process instrumentation, it is an attractive consequence to interface also standard computer peripherals to the computer via CAMAC modules, as an alternative to the traditional approach of dedicated interfaces linked directly to the I/O channel with special-purpose software for support of each device. This allows to select peripherals like alphanumeric and graphic displays, keyboards, paper-tape and magnetic-tape units etc. from a wide market, considering performance and cost. In addition, these developments may easily be transferred to or duplicated at another computer [5].

(3) Finally, it is obvious to extend the power of a small computer by coupling it to a larger processor. This concept of linking satellite computers to central computing facilities becomes more and more attractive, as the transfer of data by means of a data link can be handled much more effectively in terms of time and cost than by means of magnetic tapes or other data carriers. Again, CAMAC instrumentation is used for module-to-module data transfers via private lines

- in high speed data links if real time feedback of experimental or instrumental parameters is required,
- or in low speed data links, if only the facilities of a remote large central computing system is needed.

Besides application, system configurations are determined also by many other factors like the required performance or task, the size of the system, the distance and noise of the transmission lines involved and the rate of data transfers. The CAMAC specifications are written deliberately that almost any of these requirements or conditions can be accommodated. To illustrate this some typical examples will be given of how CAMAC systems can be arranged.

2. DATAWAY-PROGRAMMED CONTROLLER SYSTEMS

A very simple single crate system with no computer involved is shown in fig. 1. Its major task is to transfer the data of a set of user modules to a permanent record. Data-logging systems are a typical example. The controller generates pairs of commands - one command to read data from a module to a register in the controller and another to send this datum to a peripheral interface module driving a teletypewriter, a paper tape punch or a magnetic tape recorder.

Sucha simple controller can automatically scan and transfer data to or from an array of modules by using a special feature of CAMAC called "Address Scan Mode". In this mode each module generates a Q = 1 response signal during a read or write operation at all sub-addresses at which data registers are present, but generates a Q = O response at the first unoccupied sub-address. The controller then acts accordingly to access the next data register: When Q = 1, the sub-address is incremented and again a read or write operation is performed. When Q = O, the sub-address is set to A(O) and the station number is incremented. This allows to detect unoccupied stations within an array or that the scan has passed the last data register in a module, at the expense of just an unsuccessful transfer operation. And the block transfer is terminated by the controller when a specified word count has been reached.

A system as shown in fig. 1 may employ a controller which has the capability to read all commands to be performed in sequence which can be a fixed or a programmable read only memory. Although the program is of rather limited size (e.g. 256 or up to 1 K words), it may contain statements to perform CAMAC operations in any sequence and to any module, including branching on Q-response or jump to a subroutine. The start of the programmed sequence may be initiated by an external trigger event to the controller or by a Look-at-Me-signal from a user module. The program may stop in a wait-condition and continue upon another trigger signal. In order to operate character-oriented peripheral devices Binary-to-BDC conversion is performed either by means of the controller hardware or by an optional plug-in module.

Such a small stored-program system can be considered and is used as a low-cost remote terminal: Data can easily be transferred to or from a computer instead of to a peripheral device (see fig. 2). For this purpose a simple, serial, asynchronous data link via a Teletype I/O-module may be sufficient. Then, the computer can trigger the program sequence in the remote autonomous subsystem which runs at full CAMAC cycle time of about 1μs/transfer, allows independent front end recording or monitoring of data and causes only few program interrupts to the computer [7, 8, 9].

3. DATAWAY-COMPUTER SYSTEMS

A standard method of connecting a CAMAC crate to a computer is by means of a special "computer-dedicated" crate controller linked directly to the I/O bus as shown in fig. 3. The crate controller serves as a CAMAC-computer interface and appears as one or more standard peripheral devices to the computer. The most common requirement is to have programmed I/O data transfer in which each CAMAC operation is under control of the computer program, that specifies the CAMAC address, the direction of data flow and the operation to take place. In a write operation, for example, write data are transferred to the data register of the controller, then the CAMAC command is loaded into the command register and the Dataway operation executed. In order to use only single word instructions for data transfers (as well as for dataless functions like Test, Enable or Execute) some designs are based on the principle of transparent operation whereby each module address and sub-address in a CAMAC Crate corresponds to a peripheral address of the computer. With 5 bit for the station number and 4 bit for the sub-address, in this case each crate requires 512 peripheral addresses.

This configuration of direct interfacing a CAMAC crate to the Input/Output channel of the computer is most economic for small and simple instrumentation systems, with preferably a simple crate only, and if future expansion of the system is not encountered. If several crates are required, it is possible to have a radial configuration with a dedicated crate controller for each crate adapted to the channel structure of the specific computer used. This may turn out to be less economical

- if another type of computer is used for the same system,
- or if more complex system requirements, for example, direct memory access or address scan facility, necessitates other, more powerful types of dedicated controllers for the same computer.

In addition, there are often limitations due to the number of peripheral addresses available and due to the physical length of the computer I/O bus [6].

4. BRANCH HIGHWAY - COMPUTER SYSTEMS

Another, more powerful and well-known basic configuration, shown in fig. 4, makes use of the CAMAC Branch Highway, a bidirectional data bus for information transfer of up to 24 bit in parallel. Up to 7 CAMAC crates may be linked to the Branch Highway which is defined - like the Dataway-independent of the type of computer used. Thus, in each crate a standard crate controller designated Type A-1 (CCA-1) is used, which is available from many manufacturers. Thereby, this approach has often proved to be an economic solution even for small systems, especially if the inherent capability for easy expansion to larger and/or more complex systems is taken into account. The Branch Highway, being an interconnection for multicrate systems, is defined as a port for cable link with 66 twisted pairs of wires and may be used directly if the physical length does not exceed about 50 m.

The Branch Highway conveys in principle the same signals as the Dataway. However, the time sequence of a Branch operation is controlled by a handshake mechanism similar to a dialogue. That is, a data transfer between the Branch Controller (at the computer end) and the crate controller involves several phases, and each controller starts the next phase of the transfer only, after having received an acknowledge from its partner that the preceeding phase has been terminated. This handshake mechanism accommodates automatically any signal delays and therefore any length of the Branch Highway. Therefore, CAMAC Branch driver/receiver modules for balanced signal transmission are commercially available, thus allowing the Branch Highway concept for high rates of data transfer (bitparalel, wordserial) to be used up to about 1.000 m. Both the Branch Highway for CAMAC multicrate systems and the standard Crate Controller Type A are defined in the CAMAC specifications [2], adopted in Europe and in the USA.

The Branch Highway provides a computer-independent port which allows any computer to communicate with the CAMAC multi-crate instrumentation system via a single interface only, which is called the Branch Controller. These may have very different capability - for example program controlled transfer only or, in addition, block transfer of data, or add-to-memory mode, or automatic incrementing the external address and so on. For most of the small process computers in use today, CAMAC Branch Controllers are commercially available with a wide spectrum of functional performance.

5. MULTIBRANCH/MULTISOURCE SYSTEMS

Although seven crates full of CAMAC units amounts to about 160 user modules and represents a very large system there are applications which require an even greater number of crates. Thus, for large instrumentation problems multibranch-systems have been considered. There are no CAMAC specifications or even recommendations for multibranch-systems, although the problem and many different solution have been discussed in great detail.

Obviously, part of all Branch Controllers is a simple unit with very similar characteristics, called a Branch Driver, which serves the Branch Highway port and is independent from a particular computer. Thus, a conventional approach is to provide a separate, simple Branch Driver for each Branch and multiplex these via a common interface to the computer (see fig. 5). Such a system may be employed not necessarily to permit a large number of crates but possibly also to allow a star configuration of Branches starting from the computer in different directions.

There are also applications where more than one command source share the same instrumentation system. These can be two minicomputers with a different task for each, like data acquisition and process control. Or simply an additional fixed program hardware controller, a manual controller or separate controllers for the I/O bus and the Direct Memory Access Channel of a computer. Each command source is linked to the system by separate interfaces, called System Controllers or Couplers. The common problem in both multibranching and multisourcing is to coordinate the communication between a System Controller and a Branch Driver. This multiplexing is effectively done on the basis of a master-slave relationship: Any System Controller may request to become master of the system. In a master-select operation a specific system controller is selected by means of a priority network. The master obtains control of the system and then selects one of the Branch Drivers as a slave by a slave-select operation. Normal Branch operations, conforming fully to the CAMAC specifications, can then take place between the master System Controller and the seven crates associated with a selected Branch Driver.

An attractive implementation is the concentrated multibranch/multisource solution as shown in fig. 6. It uses a standard CAMAC crate as a system crate, with CAMAC modules as Branch Drivers and Computer Couplers. An Executive Controller at the two right-most stations of the crate controls the multiplexing scheme by responding to a master-request and selecting which system controller shall have exclusive control of the system at a time. Although the System Crate approach allows large and very powerful systems, it is a rather economic solution: Due to the modular structure a system is only equipped with Branch Drivers and Computer Couplers actually required. Furthermore, the Dataway is used, on a time-sharing basis, for intercommunication of system crate components as well as in the very conventional manner according to the CAMAC specifications. Therefore, standard CAMAC modules can be employed in the System Crate which allows application even for single crate instrumentation [10].

6. SERIAL HIGHWAY-COMPUTER SYSTEMS

The parallel Branch Highway has proved to be a powerful means for high data rate transmission in compact multicrate systems. Its application in simple or geographically distributed systems is however limited due to the complexity of the Branch Controller required and the cost and installation of a multiwire cable. The CAMAC Serial System provides another means for interconnections in multicrate instrumentation, with serial data transmission. The Serial Highway is intended to complement the Branch Highway in applications

- where long distance links between crates are necessary,
- where considerable noise on the transmission line has to be encountered and protection against transmission errors is required,
- or where simplicity of interconnections between crates and to a computer is particular important.

These features of the Serial System are most attractive for industrial applications and will serve to introduce CAMAC instrumentation more rapidly into other non-nuclear areas. A detailed Description of the Serial System [3] has been completed, just in time for this Symposium, as a result of a joint effort of both the NIM and ESONE organisation and will be used as the basis for a final specification.

This Serial Highway is configured as a closed loop between the output port and input port of a Serial Driver (SD) at the computer, as shown in fig. 7. This provides an unidirectional path for all messages. Each Serial Crate Controller (SCC) therefore need only have two ports, and all messages are retransmitted at the output port except those addressed to the crate controller itself. There are three types of messages: a Command message, transmitted from the Serial Driver to the addressed crate controlled; a Reply message generated by the crate controller as a response to a Command; and a Demand message generated by the crate controller as a response to a LAM signal in its crate. All messages are organized as sequences of bytes, and may be transmitted either in bit-serial or in byte-serial mode, on choice of the system designer. The logic of the system is defined to be virtually independent of the speed of transmission, ranging from 110 Bit/s (Teletype speed) to about 5 MByte/s. The Serial System Description defines the format of all messages, the message procedure and the signal standards at the input-output ports of Serial Crate Controllers, just to name the most important items. The defined (balanced current-mode) signal standards may be utilized directly for intercrate transmission via private lines over distances up to about 1000 m. Other, specific types of communication channels and modulation may be used in any section of the highway, for example employing modems for operation over longer distances or via the public telephone network. The communication channel actually used will be determined by factors such as distance, speed, noise environment and cost. However, the port of the Serial Highway and the message format are defined such, that it can be connected very simple through a low-cost adaptor to standard communication interfaces (with internationally accepted standard ports like TTY, CCITT/V24 or EIA/RS 232) which are available for all modern computers. Thus, the Serial Highway can be applied equally well in very simple systems, with one or a few crates next to the computer only, without the need for a special CAMAC System Controller designed for a particular computer.

REFERENCES

(1) CAMAC - A Modular Instrumentation System
Revised Description and Specification
ESONE - EUR 4100e (1972)
AEC NIM - TID 25875 (1972)

(2) CAMAC - Organisation of Multicrate Systems
Specification of the Branch Highway and Crate Controller Type A
ESONE - EUR 4600e (1972)
AEC NIM - NDWG 73-10

(3) CAMAC - Serial System Organization, A Description
ESONE - EDWG 25/73
AEC NIM - NDWG 73-10

(4) R.C.M. BARNES
The CAMAC Serial Highway - A Preview
CAMAC Bulletin No. 8, November 1973

(5) B. ZACHAROV
CAMAC Systems: A Pedestrian's Guide
DNPL/R23, DNPL Daresbury, England, 1972

(6) CAMAC Tutorial Issue
IEEE Transactions on Nuclear Science
Vol. NS-20, No. 2, April 1973

(7) H. BISBY
The CAMAC Scheme
AERE-R 6713, AERE Harwell, England, 1971

(8) G. HERDAM
Ein CAMAC Branch Controller (PROM-BDC) für programmierbare Festwertspeicher
HMI-B 128, HMI Berlin, Januar 1973

(9) F. BAL et al.
CADET - A CAMAC Single Crate Read Only System
CERN-NP CAMAC Note 48-00 Sept. 1973

(10) K.R.E. SMITH, D.M. DRURY, N.V. TOY
The CAMAC System of Data Handling
Journal of the Institution of Nuclear Engineers
Vol. 13, No. 1, Jan./Febr. 1972

CAMAC System Port		Communication
DATAWAY	EUR 4100 e	Module / Crate Controller
BRANCH HIGHWAY	EUR 4600 e	Crate Controller / Branch Controller
SERIAL HIGHWAY	EDWG 25/73	Crate Controller / Serial Driver

TAB. 1: *CAMAC System Ports*

Application Area	Communication
Measurement and Process Control	Transducer, Actuators / Computer
Standard Peripheral Interfacing	Man / Computer
Data Links	Satellite / Central Computer

TAB. 2: *General Application Areas*

System Configuration	System Size
Dataway-Programmed Controller Dataway - Computer	Single Crate Systems
Branch Highway - Computer Serial Highway - Computer	Multicrate Systems
System Crate - Computer	Multibranch/Multisource Systems

TAB. 3: *CAMAC System Configurations*

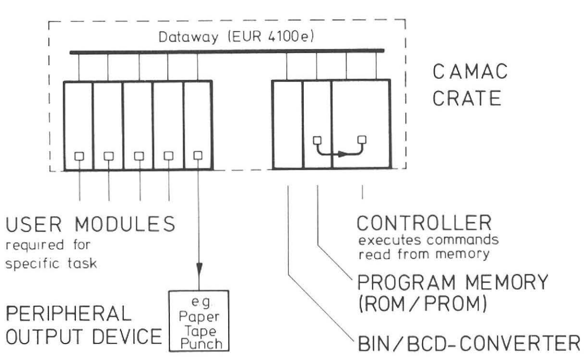


Fig. 1:
Dataway - Programmed Controller System

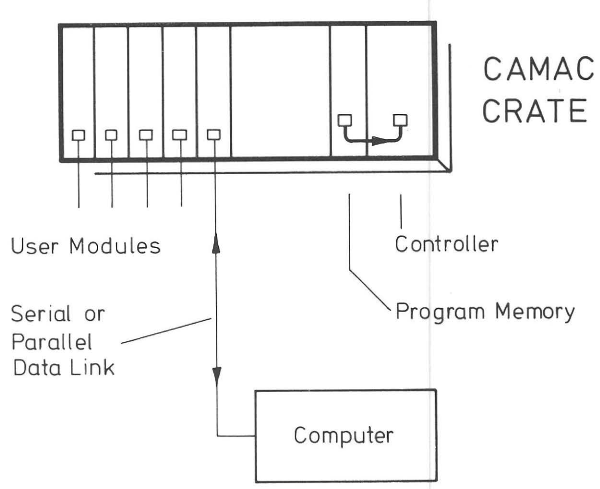


Fig. 2:
Dataway - Programmed Controller System with computer communication

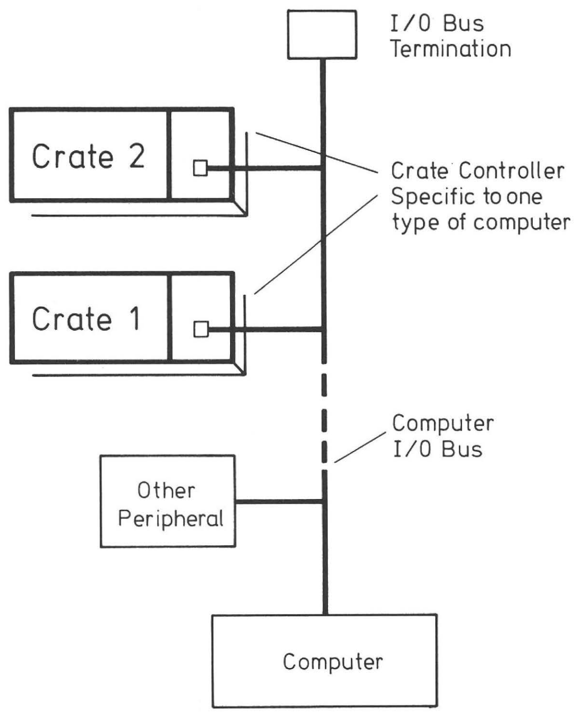


Fig. 3: Dataway - Computer System with dedicated crate controller

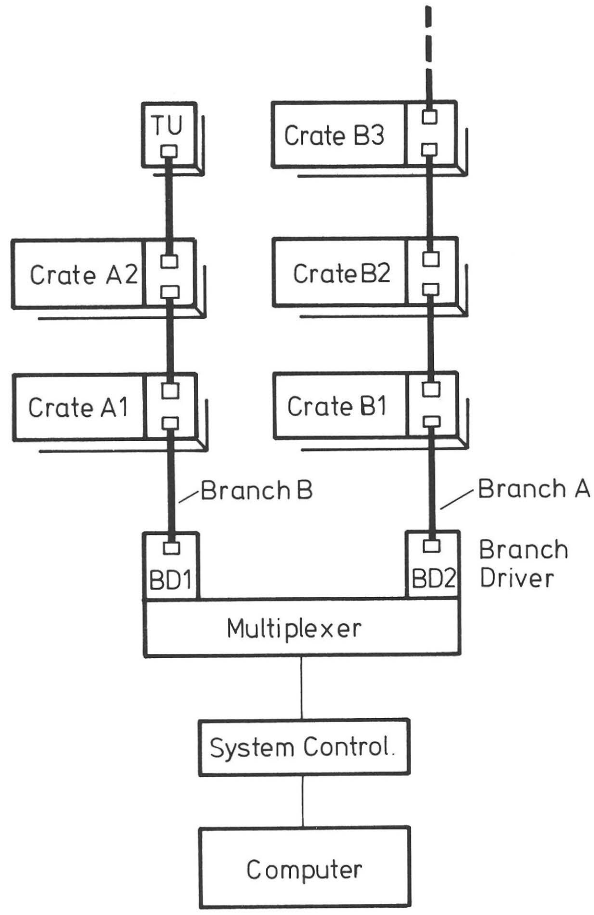


Fig. 5: Multibranch System

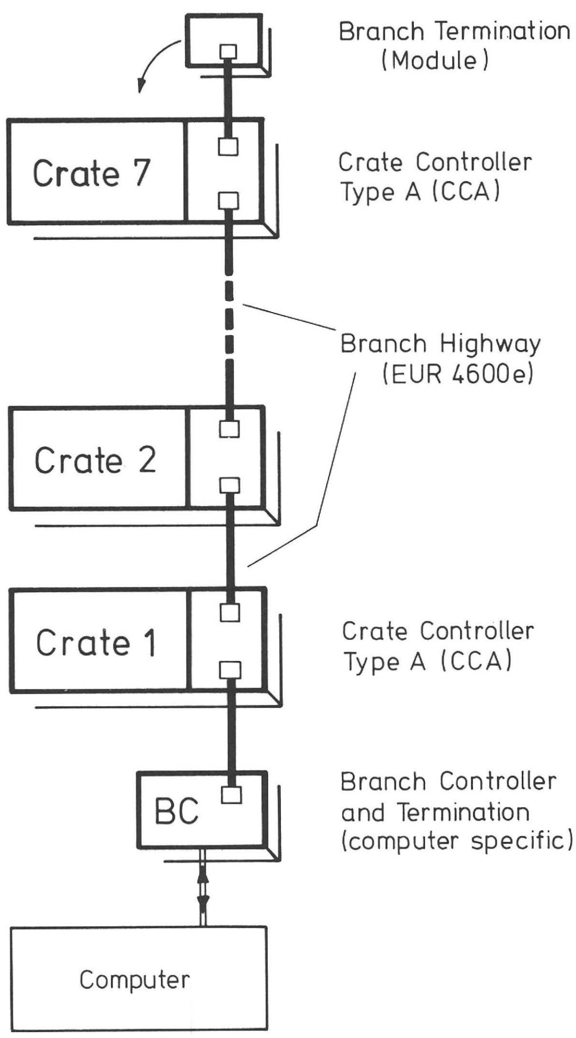


Fig. 4:
Branch Highway - Computer System with standard Crate Controller Type A

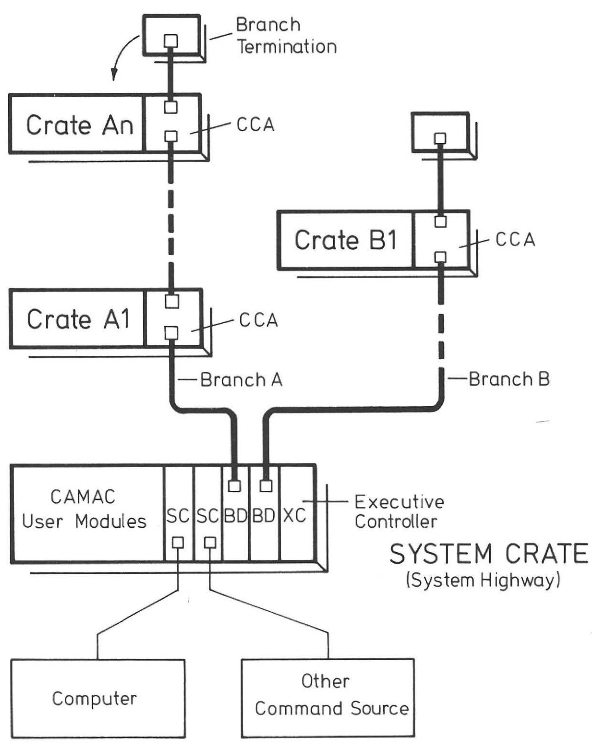


Fig. 6:
Concentrated Multibranch / Multisource System with a standard CAMAC crate as System Crate

XC Executive Controller (Priority Sorter + Crate Controller)
BD Branch Driver
SC System Controller (Command Source Coupler)

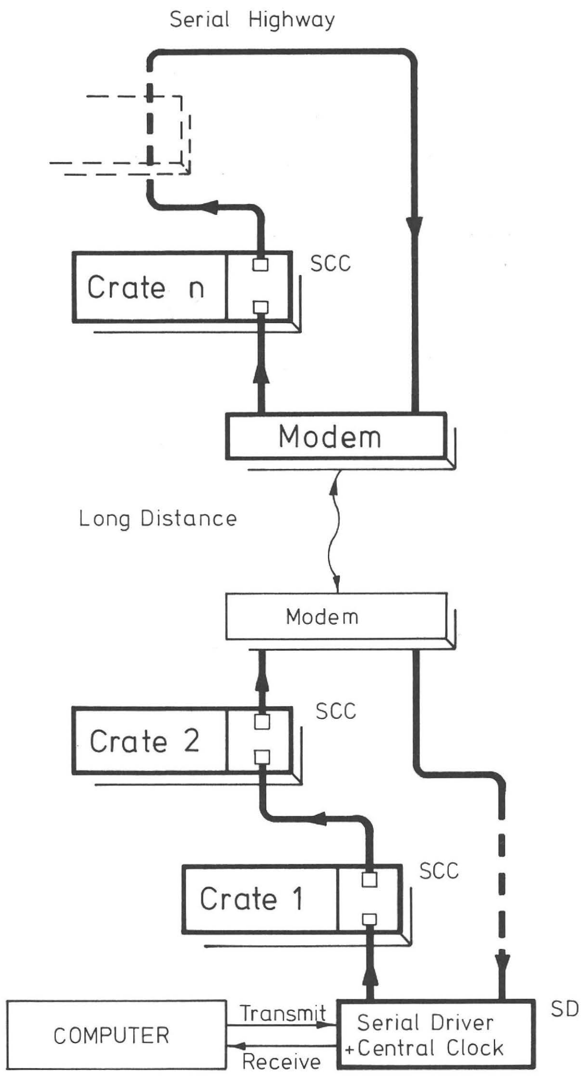


Fig. 7:
Serial Highway - Computer System

SCC Serial Crate Controller
SD Serial Driver

DISCUSSION

Q - R. Patzelt

How do typical applications fit into the different CAMAC system configurations?
Typical applications are:
1) Program controlled data collection
2) The same, initiated by interrupts
3) Program controlled process control (experiment, etc.)
4) The same, influenced by interrupts
5) Any combination of 1...4
Relevant parameters are:
1) Data-flow-rate (bits/sec)
2) Number of interrupt sources and handling routines
3) Permissible reaction-time on interrupts

A - H. Klessmann

There are many more factors involved than those given in your question.
This makes a short, but valid answer difficult.
In principle CAMAC is an instrumentation system, which covers both data acquisition and process control. Branch Highway configurations are best suited for data acquisition with high data rates and many interrupt sources. It allows also fast feedback for parameter control of instrumentation and process devices. Interrupt handling is fast and flexible, due to the facilities of LAM-grading and GL-operation, which provides priority sorting and a 24-bit Graded L-pattern from a multi-crate-system in a single operation. The overall performance of the configuration is very much dependent on the capability of the Branch Driver, which may allow program controlled transfer only, or in addition block transfer of data, address scan operation, or may perform CAMAC operations automatically. The limitations of the Branch Highway configuration are primarily given by the distance, noise environment (no error protection) and the cost of the computer-specific Branch Driver as well as of the multiwire cable link.
Configurations using autonomous sub-systems, and the serial Highway are best suited for process control systems which, in general, tend to be distributed systems and to be remote from the computer.
Means are provided for protection against transmission errors in noisy environments, the data rates required are much lower, so that long distance transmission channels with low rate can be used. The Command-Reply response in a small system, operated in bit-serial mode via a 9.600 baud channel takes about 13 ms. The Demand response is much slower as compared to the Branch Highway, however is still powerful, as the source crate Address and the GL-field in a Demand Message may be used as a direct-vector to the LAM-source and the corresponding interrupt routine. The major advantages are due to simple interconnections between crates and to standard ports of computers by means of a low-cost TTY-port adaptor.

Q - J. Laver

With regard to the serial branch. When this is to be used over practical long distance transmission links subject to errors and breaks.
1) What is the hardware recovery system proposed to enable the serial branch system to recover quickly from errors and breaks?
2) Why make it a prerequisite of operation that commands must pass right round the system before execution? This would mean no part of the system can be used if any one link is disturbed.
3) What facilities and system is proposed for local autonomous crate controllers?

A - H. Klessmann

1) The Serial System provides the means
- for bypassing a crate ("suicide") in case of failure of an SCC, or if withdrawn from service
- or for loop-collapsing ("murder") if a section of the loop is circumvented, which

allows testing of the loop and continued operation of the rest of the system.
If an SCC has detected a transmission error in a command message it returns an Error Reply message to the Serial Driver.
If an SCC has lost message synchronisation it retransmits at its output port all information received at its input port. It acquires byte synchronisation by means of the byte clock in conjunction with the framed END- and WAIT-bytes (in bit-serial mode). An SCC asserts message synchronisation after receiving two or more consecutive DELIMITER-Bytes (END-, ENDSUM - or WAIT-). The next non-Delimiter Byte will be interpreted as the first byte of a message.
2) The Serial Highway is a unidirectional loop in order to cope with the problem of asynchronous demands and propagation delays. A Command is executed in an SCC after the error check at the end of the Command Message indicates that the message has been received without transmission error. The Command therefore does not pass around the whole system before execution. The loop concept of the Serial Highway requires however an intact loop for proper operation (see 1).
3) The Description of the Serial System does not describe an autonomous Serial Crate Controller at this time. Various features, however, allow such an implementation, e.g. for block transfer operations data fields of other length than 4 bytes, etc.

C - K. Müller

At the KFA Jülich an autonomous controller for a serial loop is at the present time under development which follows the Description of the Serial System with respect to the input and output ports of the Serial Highway and conforms to the Specification EUR 4100e for its connection to the Dataway.
The main elements of the unit are however a CPU chip (Intel 8008, which will be replaced by the Intel 8080 in April 1974) and 4K bytes of RAM and 4K bytes of PROM memory. A further expansion of the memory capacity can be accomplished by the use of additional CAMAC memory modules.

THE NEED FOR CAMAC SOFTWARE

I. N. Hooton
A. E. R. E., Harwell, England

ABSTRACT

CAMAC Software consists of statements which enable a user to call for CAMAC activities directly without reference to the mechanism of the controller. Different implementations of CAMAC Software can be expected to have different syntactic forms but to have similar semantic features.

ZUSAMMENFASSUNG

CAMAC-Programme haben Angaben zum Inhalt für deren Form der Anwender nicht auf den Aufbau der von ihm verwendeten Steuereinheiten Rücksicht nehmen muss. Beim Einsatz von CAMAC-Programmen in unterschiedlichen Datenverarbeitungs-systemen kann damit gerechnet werden, dass wohl Unterschiede in der Syntax vorkommen können, dass aber in bezug auf die Semantik die Programme ähnlich sind.

RESUME

Le software CAMAC est constitué d'instructions permettant à l'usager d'appeler directement des activités CAMAC sans être tenu de se référer au mécanisme du dispositif de contrôle. On prévoit que les diverses applications du software CAMAC auront des syntaxes différentes mais que les caractéristiques de la sémantique seront similaires.

1. Introduction

The Camac specifications define the operations performed within the Camac hardware. They also standardise the hardware "Camac commands" which control these operations. However they do not define the detailed sequence of computer instructions required to set up and execute Camac commands. This is because the instructions are implementation dependent, being a function of the computer programming system and of the controller (Camac-computer interface).

With the expansion in the use of Camac an increasing number of users do not have expert knowledge of the internal working of their controllers and hence cannot be expected to generate the required sequence of instructions. Camac programming statements are therefore provided which enable the user to specify Camac activities directly without reference to the mechanism of the controller. These statements constitute 'Camac software'.

It is the job of a system programmer who understands CAMAC, the controller and the computer to provide the sequence of instructions that corresponds to the Camac statement in the user's program. The sequence is implementation dependent; the Camac statement can be implementation independent.

There are many ways of providing the user with CAMAC software. One method is for the system programmer to generate a set of macro-expansion or sub-routine calls. The user is told the facilities provided by each call and the arguments that he must supply. These will include information relating to the Camac hardware; for example, which register is to be addressed and the function to be performed. For data transfers between Camac and the computer a computer memory reference must also be supplied.

An example of a user statement (in the IML macro-expansion syntax) is

SA READ,TIMER,TIME

The macro-name SA defines that this is a single-action statement. The action required is for the content of the hardware register TIMER to be transferred to the computer memory reference TIME. At program assembly time the corresponding sequence of instructions required for the particular installation is generated.

2. Semantics

When defining any form of Camac software the system programmer must decide what facilities will be provided, that is, the semantics of the software. His selection will be largely influenced by the applications he has in mind.

A possible attitude is that, since Camac hardware is essentially general purpose, it is sufficient if each individual Camac operation is available to the user. However both execution time and program space are reduced by combining operations and some additional features may be required. The current interest in 'block transfers' is an example.

3. Syntax

Camac software acts as a bridge between Camac operations and computer processing. The computer system and the programing language used set constraints on the syntax of Camac statements. For example, the Camac statement previously written in macro-expansion syntax as

SA READ,TIMER,TIME

might be written as

CALL SA(READ,TIMER,TIME)

for use in a Fortran sub-routine call environment.

Both these examples are Camac statements that can be added-on to existing languages. There are processing languages in which Camac facilities are or can be incorporated. Here the syntax is even more closely controlled.

Another aspect of the syntax is the attitude of mind it assumes in the user. Consider

SA CLEAR,TIMER

and

SA 9,1,2,3,0

These two Camac single-action statements both clear (F9) the contents of the register TIMER (located at Branch 1, Crate 2, Station 3 and Sub address 0). The first form concentrates attention on the software, the second on the hardware.

4. Discussion

Camac software exists. Some of the methods of making it available to users will be presented in the software session of this conference. The syntactic forms of the software will show considerable variation. As long as there are many different processing languages in common use we must expect to see many different forms of Camac syntax.

However each approach is aimed at giving the user the facilities he needs to incorporate Camac in his computer system. We may therefore hope to see some similarity in the semantics. An agreed semantic definition of Camac software would be of great importance. It would establish guide lines for those implementing new forms and give a reference for describing existing forms. It would also help the user to specify his requirements to the system and controller designers.

THE COMMERCIAL CASE FOR CAMAC - A MANUFACTURERS VIEW

D. M. Drury
GEC-Elliott Process Automation Ltd, Leicester, England

ABSTRACT

A study is made of the structure and practises of both customers and suppliers of the market "for interfacing computers to data transducers and actuators in on-line systems", and the effects of the adoption of an international standard on those practises is discussed. The arguments are used as the basis for a commercial justification of manufacturing support for the CAMAC standard.

ZUSAMMENFASSUNG

Für einen Markt "von Systemen zur Zusammenschaltung von Messwert-Erfassungsgeräten und Wirkungsgliedern mit Digitalrechnern für Echtzeit-Datenverarbeitungsanwendungen" werden die Art und die Verhaltensweisen von Verbrauchern wie Lieferanten untersucht; wie sich die Übernahme von internationalen Regeln für solche Systeme auf die Verhaltensweisen auswirkt, wird erörtert. An Hand der gegebenen Argumente wird die Herstellung von Geräten nach den CAMAC-Regeln kommerziell begründet.

RESUME

Etude de la structure et de la pratique du marché des utilisateurs et des fournisseurs dans le domaine de la "connexion des ordinateurs à des transducteurs de données et à des dispositifs de commande à distance dans les systèmes on-line" et discussion des répercussions exercées sur cette pratique par l'adoption d'une norme internationale. Les arguments mis en cause sont utilisés pour justifier la promotion de la fabrication du standard CAMAC.

1. Introduction

1.1 To date CAMAC equipments have found wide application in solving interfacing problems for which by and large no other solutions exist. Though this is a considerable market in its own right, it is a small fraction of the market "for interfacing computers to data transducers and actuators in on-line systems" in which CAMAC could have application. To analyse whether there is any commercial case to be made for the use of CAMAC in application areas that do have existing solutions, it is necessary to study the structure of that market and the current practises, of the customers and the suppliers.

2. Structures of the existing market

2.1 The customer

The most obvious place to start in looking at the structure of a market is with the customer. Who is the customer for interface products? A simple answer would be the end-user of the computer system, in whatever application the system is being used. But the end-user will probably not be aware of the interfacing details, however they are implemented. He is concerned only in obtaining the desired overall system performance. It is the function of the system engineer, via the applications programs, device control programs and the interface structure he employs, to make the system appear transparent so that the end-user is unaware of the details of the system construction. It is therefore the systems engineering organisation that is the customer for interface products, whether this organisation is a department of the end-users organisation, or an independent systems company. In both cases the function they serve is identical.

2.2 The Systems company

2.2.1 It is an organisation whose business is in providing a complete computer system, including all the interfacing hardware, and all the operating software. The responsibility for the correct functioning of the system rests normally with this organisation, and this is a major undertaking for it commits the systems organisation to finding the cause of any system malfunction, whatever and wherever it is.

2.2.2 Each system is by its very nature unique, and will require expertise in the application area for which the system is intended. This then is the service provided by a systems company, but because work unique to a project is expensive in both time and money, a major part of the design function of the systems company is to construct the system from catalogue items wherever possible, and hence reduce to a minimum those functions that are necessarily unique to each project. (1)

2.2.3 Fig.I illustrates all the principal functional and physical blocks involved in constructing an on-line system. All the functions must be performed before the system can operate, and so the diagram represents on the one hand, the total commitment of the systems engineer, and on the other shows the various catalogue items that can be employed.

2.2.4 At this point it is necessary to distinguish between systems companies that are departments of the end-users organisation, independent commercial systems-only companies, and systems companies of a manufacturing group, as their methods of operation is based on different premises. In particular the systems company of a manufacturing group is (normally) constrained for obvious commercial reasons, to use in its systems the products of the rest of the group, wherever possible. The independent or end-user systems company can (in theory) select the "best buy" for each type of catalogue unit that makes up his system. In both cases however the selection of the particular units requires a system knowledge to select the best buy such that the units are compatible with the requirement and also with each other.

2.2.5 This latter point represents a major difficulty in the absence of any international standardisation, especially as far as process peripheral interfaces are concerned, to the extent that with existing commercial practises the systems company is forced (or obliged) to go to one supplier in order to guarantee compatibility between interface products. In this area he (currently) loses any freedom of choice, and his ability to reduce the unique content of a system is determined to a considerable extent by the range of process peripheral interface products from which he must do his selection.

2.3 It should perhaps be noted that this method of organising system design has only developed over the last few years, and has resulted from the rise of minicomputer suppliers providing hardware at very low costs. Prior to that the computer and its associated hardware were the major part of the system cost. The system design and interfacing were often tackled in a unique manner since the cost was small in comparison with that of the computer hardware.

3. The Catalogue item suppliers

3.1 Computers

The dramatic change in the use of on-line computer systems also stems directly from the availability of computer hardware from "mini computer" suppliers at low costs. The low costs were in turn achieved by the computer manufacturer producing large quantities of a standard range of equipments, and by eliminating virtually all of the systems support normally associated with the larger machines. The development of component technology (I.C's) and printed circuit wiring techniques all contributed to reducing replication costs at the expense of increasing the initial production design costs.

3.1.1 Most of the applications at which the machines were aimed could be resolved with a small range of the so called "business" peripherals, i.e. paper tape, card, magnetic tape, disc, drum, line printers etc. (the character orientated devices).

3.1.2 Because each mini computer is a standard product of a particular company, there is obvious commercial pressure, indeed necessity, to avoid international standardisation, as the differences in architecture in the I/O Bus, the C.P.U. and the instruction set are the only selling features available other than price. The inevitable corollary is that even though most mini computers employ 16 bit data words, and I/O structures, they are all totally incompatible. If this were not so, the problem of interfacing would be by now trivial.

3.2 The business peripherals

The central processing unit represents a small and reducing cost for any given system. The next area of costs are the business peripherals, and the trends in the development of these devices has gone further than that of process peripheral interfaces. Most computer manufacturers support a range of 'business' peripheral devices, each with its necessary interfaces and operating software (sometimes) to make them compatible with the computers own software. Here the pressure is to provide as few items as necessary, for each new peripheral requires developing, with its interface etc.

3.2.2 In practise the last few years have seen the emergence of the specialised unit suppliers, for such business peripherals, and the adoption of an International non-proprietary standard (CCITT V24) as the "one-to-one" interfacing standard to remove the interfacing constraint. (2)

Again the commercial logic is identical to that for mini computers. The development cost of say a tape transport, or a line printer is very great, and the product cost can only be minimised by sales in as many application areas as possible, i.e. maximising the selling base. This cannot be done by restricting the sales to one particular mini computer, and hence the inevitable separation of the computer supplier and the peripheral supplier. This in turn inevitably leads to the adoption of a standard, as has already been explained.

3.3 Process peripheral interfaces

3.3.1 The situation in this field up to recently was totally different. In order to cope, even moderately well, with the vast range of process peripheral devices available to the system designer, at least 50 device interfaces must be developed and possibly many more. Now because the requirements are so different from business peripherals in terms of the ability to assemble an infinite variety of configurations of device interfaces, and to connect these to plant, a complete scheme of peripheral highway structures, mechanical packaging, cubicling and powering philosophies have to be developed. This represents an enormous investment, and two clear approaches to it can be determined, dependent on where the interface company sees its main customers.

3.3.2 Where mini computer manufacturers are principally hardware specialists, and are production, not systems orientated they normally side step the issue by not interfacing all the way up to the process peripheral device. Typically one will find "general purpose" digital inputs and outputs coupled with "general purpose" analogue inputs and outputs. The systems engineer is then left with the problem of interfacing these interfaces to his real world, very often an extremely difficult task, and certainly very costly and time consuming.

3.3.3 Where the customer is a systems company, and this applies particularly to companies where systems groups are responsible for a significant percentage of the companies business, the market for dedicated process peripherals can be sufficiently large to warrant the development expenditure, on such a peripheral system and hence reduce the front end systems engineering to a major degree. Even here though, because systems tend to require low quantities of a wide variety of device interfaces, and because the development costs of each are very high in comparison with the replication costs, some general purpose interface units have to be used to limit the scale of the development.

3.3.4 In both cases the design of the process peripheral interface structure is usually based on the computer with which it is intended to be sold. It follows that all the current process peripheral device interfaces from different companies are, like the I/O Buses of the computers totally incompatible, and it is again obvious that the market for the specific device interfaces is limited to selling with the chosen computer.

3.3.5 There is a third category of supplier, the manufacturer of specialised I/O equipment. Companies specialising in analog equipment, both ADC's, DAC's and multiplexers are most common in this field, since the technology is a complex one and the design problems are far less predictable than on pure logic systems. The products tend to be self-contained systems, and cope with specialised and yet common problems such as data logging and pulse height analysis.

3.3.6 In this manner the products are offered as special peripherals often with stand-alone capability, falling somewhere between the full process I/O system and the "general purpose" interfaces of the computer hardware supplier. It is normal for such companies to offer interfaces to several popular machines for product sales on the one hand, and to offer a systems capability on the other. Whilst the special product is a high cost device, and the construction techniques are specialised due to the technology used, the burden of interfacing to several computers is small. However should any of their expertise be more widely available, say through monolithic analog circuit developments, the product of their business could appear threatened,

3.4 <u>Software products</u>

3.4.1 So far we have referred only to items of hardware as the products (or catalogue items) of a particular range of process peripheral interfaces. However when such interface assemblies are designed to function with just one specific computer, (as is normally the case) and when the systems customer is another member of the same group of companies, then additional product software support can also be provided on a catalogue item basis.

3.4.2 The chosen computers real time operating system will access all peripheral devices through a standard I/O routine, using a specific Device Control Program (DCP) for each peripheral. The DCP is peripheral dependent, however as the system is intended for real-time use both the DCP and parts of the operating system have to be written in assembly level language, and are therefore also machine dependent.

3.4.4 The other area in which program segments specific to a peripheral can be provided is system test, in this case designed to run with a system test operating system. Both DCP and STP (system test programs) can therefore be provided as product software with each device interface, when a specific machine is chosen.

3.5 Configurations

3.5.1 Configurations represent the highest level of product based support that it is possible to offer a systems company, and consist at the hardware level of an assembly of peripheral interfaces, mounted in their housing, in cubicles, complete with all wiring up to the terminations for plant cables, the necessary power supplies, mains distribution, and cooling, that is required for a specific project. In other words all the functions from (1) to (7) on Fig. I plus the undertaking that any configuration rules of the scheme have been obeyed. To the hardware configuration can then be added the specific computer configuration, [(8) and (9) on Fig. I] and the two together can be checked in a configuration test using the STP's [Fig. I (10)] . The systems company is then left with the very minimum of unique project work, in adding the application programs and the site cables to complete the system.

3.5.2. It is important to remember that these functions must be performed however the system is assembled, and with particular reference to purely catalogue items of hardware from hardware oriented manufacturers, all the configuration work, normally falls to the customer. It is common for this to be considerably underestimated as a problem - and easy to make errors if the configuration rules are not well defined. Both of which can seriously affect the time scale and hence the cost of a project.

4. The function of standards in process peripheral interfaces.

4.1 The ability to configure a virtually infinite variety of unique assemblies, with the very minimum of effort particular to any specific arrangement, is one of the fundamental design objectives of any process peripheral interface structure, since not only are there a vast array of process peripherals to interface to - but in all probability no two systems will use the same combination of quantities and types of peripheral interfaces. The ease which any given system can be configured will have a major bearing on the total cost of each configuration, which is borne by the supplier if he supplies configurations, or the customer if he purchases a set of catalogue items only. Each supplier does this by conforming to his own in-house mechanical and signal standards to achieve compatibility throughout his range. What then are the benefits of his using a non-proprietary international set of standards, on which to base his range of process peripheral interfaces?

4.2 The products will now be compatible, within the limits of the international standard, with all the other peripheral device interfaces made for that standard. This has two direct effects, both of which can offer significant commercial advantages.

4.2.1 All his hardware products at the unit level, now have available a purely product hardware market, to all assemblers of configurations of standard equipments. Product sales can be supported very, very much more simply in overseas territories than can system sales as one is marketing a well defined item of hardware only. Further if the standard truly is international, and internationally employed, he will get his product sales over a vastly greater market area than for a proprietary product as he can leave to the local configuration supplier the task of selling the total package. Indeed an international standard would probably encourage specialist unit suppliers supplying only catalogue items of hardware to emerge, as a parallel to the business peripheral market mentioned earlier.

4.2.2 Secondly if the manufacturers principal function is to provide configurations, he has then available for his use in his configurations all the units available from all the other sources of supply. Even if the degree of compatibility of the international standard does not entirely solve the total configuration problem, such problems are much cheaper and much faster to resolve than to design a new unit as would be the case with a proprietary standard. It must inevitably be true that, if the standard becomes established, far more distinct interface units will be available than any one manufacturer could support. In addition, even configuration sales need not necessarily be tied to the sales of one computer, since the existence of the standard would bring into existence a range of I/O Bus to process I/O highway interfaces for many different computers. Mainly for software reasons, the computer already has influence over systems grossly out of balance with its costs.

4.3 The existence of the standard will in all probability also change the customer situation. If the task of configuration reduces, then the predominant task falls on the system software. For software - only systems houses the attraction of in effect having the biggest catalogue of peripheral interface items in the world available to them with easy well defined configuration rules means that they can now undertake hardware supply as well as software. The ultimate benefit will come when applications software is also transferable, so that once a new application has been solved somewhere with the standard interface, that solution is available to all users or suppliers.

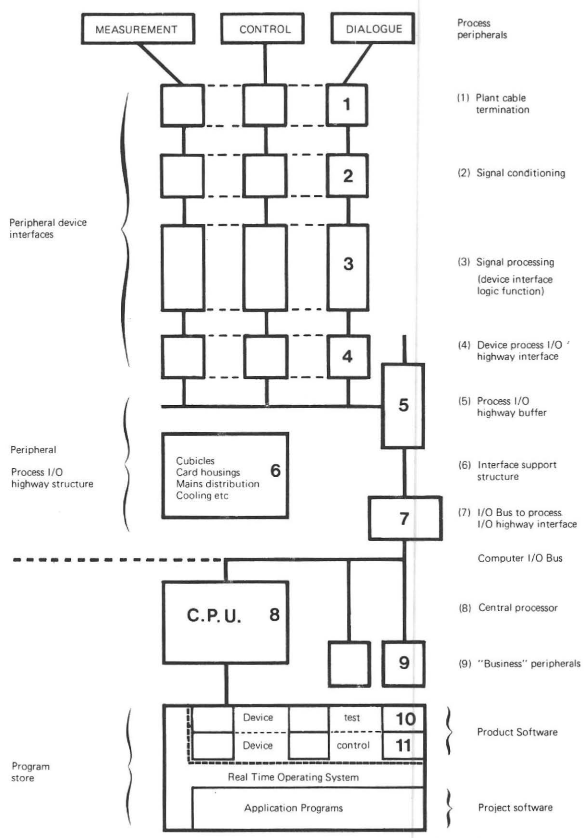
4.4 The disadvantages of a standards solution are:-

4.4.1 The economic penalty on hardware cost for following a standard rather than "minimum cost" solution.

4.4.2 The dependence on a standards authority to maintain, evolve, promulgate, and (particularly to) interpret the standards so as to maintain the standards discipline. This task of international communication is an enormous and vastly expensive one that must be satisfactorily resolved if the standard is to get conformity over a significant area in which it has potential.

5. CAMAC

This then is the commercial case for an international standard, and since CAMAC is the only specification of that status in existence it is also the case for CAMAC.
It would appear in practise that CAMAC is on a vital threshold, economic analysis puts the economic cost penalty at up to 10% on hardware, which is already seen as not too significant in comparison with the potential savings in system costs that result. Provided the standards organisation has the resources to adequately control the standard, with its help and that of enlightened customers, CAMAC can be transformed from an interesting scientific product to a vital component in on-line computer systems internationally.



Note: that although the peripheral device interface is made up of all the functions (1), (2), (3), and (4), the plug-in unit of any particular scheme may not contain all four functions. In this case a method of interconnecting the separate physical blocks has also to be provided.

Fig. 1 Principal physical and functional blocks of on-line systems.

References

(1) St. JOHNSTON, A., COX, H. Software specification and the changing pattern of system procurement. Conference on trends in on-line computer control systems. (Sheffield 1972) IEE Conference Publication Number 85.

(2) BISBY, H. Fundamental principles of CAMAC.
1st CAMAC Symposium.

Part II

I-6

STATUS AND FUTURE OF CAMAC IN WESTERN EUROPE

K. Zander
Hahn-Meitner-Institut für Kernforschung GmbH, Berlin, Germany

ABSTRACT

An action of questionnaires led to an updated review of CAMAC status and prospected futural applications. Problems of supply, use, advantages and difficulties of CAMAC employment within "classical" fields are discussed as well as chances and bottlenecks of introduction to new applications. Proposals are made for an intensified utilisation of CAMAC system.

ZUSAMMENFASSUNG

Eine Umfrage ergab eine Übersicht über den neuesten Stand und die künftigen Anwendungsmöglichkeiten von CAMAC. Lieferprobleme, Fragen des Einsatzes sowie Vorteile und Schwierigkeiten der Anwendung von CAMAC in "herkömmlichen" Bereichen werden erörtert. Die Aussichten für eine Einführung von CAMAC in neue Anwendungsbereiche und die dabei zu bewältigenden Probleme werden ebenfalls diskutiert. Der Bericht enthält Vorschläge für eine intensivere Nutzung des CAMAC-Systems.

RESUME

Les réponses à un récent questionnaire ont permis de faire le point de la situation actuelle de CAMAC et de ses perspectives d'application futures. Les problèmes de fourniture et d'utilisation, les avantages et les inconvénients de l'emploi de CAMAC dans les domaines "classiques", sont discutés en même temps que les chances et les difficultés de son introduction dans de nouveaux champs d'application. Des propositions sont présentées pour développer l'utilisation du système CAMAC.

1) INTRODUCTION

Western Europe is the birthplace of CAMAC as it was for the ESONE-System. The benefits of long-term brain stormings of so many intelligent scientists of different nations creating a modular and flexibel, internationally standarized computer controlled system are obviously - although not self-evident.

My report covers the Western European hemisphere of countries like Austria, Belgium, Denmark, Great Britain, France, Germany (West), Italy, Netherlands, Sweden, Switzerland and international organisations like CERN and EURATOM.

Within this region 48 manufacturers (Table I) produce a variety of about 560 types of modules according to CAMAC specifications (Table II). This figure of 560 types of modules does not include a large list of CAMAC-components and an unknown number of home developed special modules but it shows a remarkable 50 % increase in numbers of types of commercial CAMAC equipment within an one year period from June 1972 to June 1973. This trend underlines one most important feature of CAMAC: the independency of an user from only one manufacturer or a small monopolitic number of them.

On the other hand, the astonishing large number of types should encourage manufacturers to think about an economical shrinking in numbers of types and hopefully decrease of costs. And even more: since manufacturers are often obliged to offer whole systems, I should recommend a much closer cooperation between manufacturers to supply one another.

2) CAMAC STATUS

In order to give an updated review and prospecting view of CAMAC in the area of Western Europe, I started an action of mailing 50 questionnaires, about 30 came back properly filled in and supplemented by comments from A, B, CH, D, DK, F, GB, I, N, S. Parts of the answers were to be foreseen, namely: the actual main areas of application are the nuclear fields of high and low-energy physics, plasma-physics, nuclear and radiation chemistry (included x-ray diffraction), reactor and (mainly) accelerator control (Table III). For the later a remarkable example is the 100 % control of the Super Proton Synchrotron (SPS) via CAMAC at CERN. Remarkable also the large range of computer applications. An pioneering example for the attachment of CAMAC to computer networks is given at Daresbury Nuclear Physics Lab by means of CAMAC modules, which interface to data links directly. The trend of further application within these fields for instance in storaging experiments and heavy ion research is characterized as increasing; the number of used CAMAC-crates varies from 2/2 (It's hard to understand what is 1/2 crate?) up to 83 crates at Rutherford Labs, 120 at Daresbury and 215 at CERN I and II, Table IV.

The answers to my questionnaire concerning bottlenecks in hard- and software and arguments *against* CAMAC are somewhat conform:

a.) concerning hardware: PRICE, COMPLEXITY, TRADITION

Too expensive and too complex were the almost always given answers, with some exceptions: CERN people find CAMAC not complex enough, French and German colleagues find it not too complex.

The analysis seems to be clear: CAMAC-designed for a wide variety of complex applications is obviously too expensive and too complex for a small basic system. And intrinsic redundancy of course adds costs (and unreliability). But this is the price to be paid for a large modular system, which shows on the other hand declared (and surprisingly in their conformity) estimated 20-40% savings in costs and up to 60% savings in installation time. Members of companies state an introductional saving of time of about 2 years in the field of process control. In addition, there are CAMAC moduled ADC's available, which are less expensive than comparable others; and it is a fact that CAMAC prices are contra-inflationary. And prices are relative: of course CAMAC is more expensive than NIM; but who knows a more powerful and actual available computer controlled system orientated towards future which is less expensive than CAMAC if employed as *system*?

For those, who need only a small part of the whole system, I would like to make a proposal: the easy interchangeability of CAMAC equipment should lead to the foundation of regional electronic equipment pools, which could save a lot of money, if intensively used, maybe on the base of "rent a crate or module".

The threshold TRADITION stands for 'not invented here syndrome'. This has various, psychological and technical reasons. To some extent CAMAC suspends the direct visible principle of causality for the user and offers other modes of dialogues with the system. This may form a psychological barrier for those experimentators, who like 'handdriven' systems and their romantic and relaxing performing of experiments.

Experiences with computer controlled systems teach us that the performance of an experiment can be accelerated by a factor up to 40 (!) by early detecting and avoiding mistakes in measurement procedures etc.

On the other hand, there is a variety of actual demands to the present system, which will influence successful further development and large scale application of CAMAC in the future. Let me mention some of these problems almost in correspondence to my questionnaire:

- depending on history of CAMAC, there was a lack of attention to design for use in engineering and industrial environments as opposed to laboratories.
- almost total absence of publicity for CAMAC in the engineering world as opposed to the nuclear physics world.
- absence of vigorous support by computer industry who have no incentive to encourage use of CAMAC's computer independency
- weakness of CAMAC manufacture in systems engineering and subsequent support capability and in total over all large industrial systems responsability
- sometimes poor quality control and inadequate final testing of hardware
- failure to meet the full specification EUR 4100
- lack of vision by manufacturers in the provision of certain modules, which are frequently required in industrial applications
- difficult mechanics and unreliable connector as well as 'flimsy' front panel connectors
- no provision for autonomous remote controllers and bad interrupt concept
- problems of flexibility in conjunction with operating speed (no powerful block transfers in branch highway)
- inadequacy of present means of long-distance reliable intercommunication between CAMAC systems
- the 'not invented here' attitude and the belief that CAMAC is something used only by nuclear physicist and therefore is eo ipso not qualified for industrial applications.

b.) concerning software:
The answers to my question concerning software show an alarming indication of resignation rather than of ignorance and in very few cases of autonomy.

There should be no doubt, that only progress in superior, easy-to-use, problem-orientated real-time languages will achieve the breakthrough necessary for successful applications of the CAMAC system in all the different fields. If we don't concentrate all our attention and the thought of our abilities in solving these most important software problems, we will fail I confess that, due to a lack of documented or at least listed real-time programmes - even written in assembler language - I find myself unable to give here a detailed report on this matter.

Indeed, there are activities in Western Europe in the fields of
- CAMAC-Language, published in 72, a problem-orientated language, which handles I/O routines (no arithmetic routines)
- CAMAC-IML, an intermediate language between CAMAC and I/O assembler codes,
- CAMACRO, an aid to CAMAC interfacing programming (DNPL).

Other real-time languages in Western Europe are:
- PEARL (Germany), PROCOL (France), RTL/2 (Great Britain), PL11 for PDP-11 computers with statements for primitive I/O-CAMAC routines (CERN) and extended versions of ISA-standards for FORTRAN, and other trials.

My question is, whether it would make sense, to have a periodic and updated index of generally useful software products and packages. My demand is to avoid duplication of efforts and provide sufficient manpower to implement very soon a CAMAC-language, embedded in a suitable host language.

A review of the status of CAMAC should not be closed without mentioning a highlight in the history of CAMAC, namely the fact, that - with a midwife function of H. Bisby - the trademark 'CAMAC' found entrance into the Encyclopaedia Britanica and therefore will be unforgettable for future generations. And I should gratefully point to the CAMAC-Bulletin which is a publication of the ESONE-Committee and supported by CEC: I consider this publication as a tremendous help in disseminating the idea of CAMAC! Besides this, however, there is no effective quick information service in Western Europe.

3) CAMAC'S FUTURE
In the sense we are more or less satisfied with the development of CAMAC in the past, we should watch and control the steps toward futural applications very carefully. First of all, most nuclear research centers are subject to diversification of their tasks. This means, we all are challenged not only to widen the application of CAMAC within our classical nuclear fields, but even more to transfer our know-how to new and non-nuclear fields of application and to do this in a sophisticated manner. Table V shows applications of CAMAC within non-nuclear fields. The analysis of this Table gives a different picture: as the nature of CAMAC is that of an *experimental system*, the introduction into new experimental and testing fields or pilot installations seems to be easier than it is for classical industrial fields like industrial control and automation of factories. Though some companies characterize the CAMAC-System as *the* idea for automation in fields like space- and oceanic research and applications to hybrid computers.

CAMAC's future looks quite promising, because
- for introduction into new fields, the flexibility of the modular structure of CAMAC can meet all requirements of new applications and guarantees the adaptibility to unforeseen difficulties
- the transfer of know-how and experience in handling even complicated structures helps to save a lot of money and manpower, compared with a begin from other origins

- the application of CAMAC as *one system* within the fields of medicine, environmental research and monitoring or computer applications respectively may lead to transferable structures and comparable results.

Demonstrative CAMAC applications to such new fields are already performed and will be extended very soon. The implementation of such programmes should observe some rules:

1) We have to make sure that there will be one and only one CAMAC-System.

2) Of course CAMAC is influenced on its interior by progress above all in semiconductor technology, in the increasing use of MOS- and COMOS-devices and - in consequence - large scale integration. At the ports of dataway or branch highway, however, the specifications of CAMAC have to be strictly observed, regardless, whether a module contains pneumatic or liquid controlled Flip-Flops.

3) In addition, to this, CAMAC is tied to progress in computer technology; so it might occur some day that there will be no visible on-line computer any more because it is integrated into a crate or system controller.

4) For industrial application some people don't need the total flexibility of CAMAC and hate the chaos of analogue wire connections on the front-panels. So we have to find a civil and modified version of proper looking CAMAC frontsides. This corresponds with the demand to transfer CAMAC from playgrounds of superintelligent scientists to technical fields of applications in engineering.

5) Trends of further developments of CAMAC to multi-branch, multisource systems and the urgently demanded bit and byte-serial data transfer in using low-cost long-distance telephone lines will achieve new applications of CAMAC.

6) We should avoid, that almost all ESONE-solutions are too complicated for users.

Let us see, whether we can solve these sometimes contradictionary problems.

The answers to my questionnaire for non-nuclear appliactions of CAMAC show dominating fields of

- medical health service
- computer applications
- environmental monitoring and control

Maybe, this catalogue is uncomplete and subject to revision. Let me mention an example of introduction of CAMAC to new fields from my own country:

The German Government installed two Project Managements to introduce CAMAC to new applications in medical health service and process-automation. The former project management asked the Hahn-Meitner-Institut to arrange the system management for the application of CAMAC in the fields of nuclear medicine, lab. automation, intensive care and biosignal handling. This is a chance to take care that all the different applications in several hospitals are to be considered as an *one system-experiment*, the successful parts of which and proved results can be easily transferred to other hospitals or research institutes. In addition, the Senat of Berlin (West) intends to introduce CAMAC into many public hospitals, gaining the benefits and savings of an economized maintenance. The same might be true for other countries and other fields, e.g. environmental monitoring and control etc.

There is another unifying breakthrough function of CAMAC: while introducing CAMAC into medical applications, the dataports of analyzing analogue apparatus-till now of course very different for different manufacturers - are to be new determinated, defined and standardized. That new definition for data ports will be accepted by several larger firms under the pressure of wanted applications of CAMAC in the field of medicine.

This again is a highly political function of CAMAC with tremendous advantages in costs and operational demands for public users. I hope the CEC will realize this effect and pay suitable attention to the fact, that CAMAC replaces firm standards by international standards. This is exactly what CEC should want.

A more detailed look for other new applications related to my questionnaire informs us that these activities are small at the moment, but show increasing trends over all. The manpower within our nuclear centers involved in activities for preparing new applications is relative small. I count for the area of Western Europe about 50 people. This figure is not very imposing - even the real number may be as twice as high - and shows the fact that national governments don't pay too much attention to this kind of diversification. Governments should realize that diversification is of course a function of manpower too.

My proposal is that just nuclear centers should form system managements to support the introduction of CAMAC to new fields, convincing doubtful men and gaining from the transfer of accumulated know-how.

Another new field of application is that of employing CAMAC for computer applications (CAI, CAP etc.) links and networks. CAMAC's structure is aggresive enough to allow the combination of computer peripheral equipment of different manufacturers via this system. If there was ever tossing a hat into the arena from outsiders here is one. This fact is realized by some manufacturers who did equip their smaller computers with CAMAC compatible ports. This way CAMAC overrides an old traditional guarantee for manufacturers to sell their good products in conjunction with their expensive bad ones. Who counts the savings of costs and gains of effectiveness of this CAMAC function?

4. CONCLUSIO

The status of CAMAC in Western Europe has reached some kind of threshold, on which the further development of CAMAC doesn't depend only on the ESONE-Committee and its active working groups. CAMAC became too large an operation to handle it within the frame of the past. We should be aware that

a) other areas where CAMAC is being or will be applied are not represented and have no influence on the future development of CAMAC-standards

b) there is not enough interaction between the Supply Industry and the ESONE-Committee for the implementation of actual problems.

c) we have to find new ways for the promotion of CAMAC utilisation.

Futural activities (a) call for more support by the national governments, perhaps acting through the IEC; activities (b) and (c) demand help and sources of the European Community too.

This way this first International Symposium may become a milestone in the landscape of further development of CAMAC: till now CAMAC was our own child; but it is grown-up and begins to realize that there are other relatives too: the suppliers and non-nuclear or industrial users respectively. We will have to find new organisational forms which will broaden development and applications of CAMAC. But which undoubtely will moderate the speed and mobility of former working groups.

It is, like a beloved and grown-up child will leave the parents house.

Let me come to the end with the most urgent demand for further widened and intensified cooperation, supported by the effective help of the Commission of the European Community.

Let us demonstrate that again electronics can be the frontier to diversification with the challenging goal to attribute to improvement of quality of life, not only by better computer controlled measurement techniques and faster installation of pilot systems in various fields, but above all, using the advantages of automation features to make working places more worthy of a human being.

If there is any real and common benefitting, peaceful penetrating and separating interests overriding, worldwide introduced system: the CAMAC-System is!

Table I

	B	CH	D	F	GB	I	N	TOTAL
1972	1	4	9	9	17	1	2	43
1973	1	5	11	9	18	2	2	48

Table II

	1972	1973
System units, Test equipment	78	110
I/O Registers, Displays	148	219
Multiplexer, Converters	44	85
Other modules	25	30
Crates, Supplies	83	113
total	378	557

Table IV

CRATES	RESEARCH CENTER/INSTITUTION
80	CERN (I + II) 215 (73), 355 (74), 600 (Saturation) DNPL Daresbury 120, SRC Rutherford Lab. 83
40 - 80	CEA Grenoble, Saclay 60 KfA Jülich 50, GfK Karlsruhe 50 DESY Hamburg 45 (73), 100 (74)
20 - 40	HMI Berlin 30, GSI Darmstadt 20, CNEN Roma 20
10 - 20	SGAE Seibersdorf, AEC Risø CEGB Centr. El, Res. Lab. Sticht. Reactor Centr. Ned.
5 - 10	AEE Winfrith, UKAEA IPP Garching
1 - 5	Acc. Lab. Garching, Studsvik AB Atomenergie Universities of Erlangen, Frankfurt, Freiburg, Groningen, York

Total ~ 700

FIGURES

CAMAC-APPLICATIONS IN CLASSICAL NUCLEAR FIELDS (30 QUESTIONNAIRES)

(Table III)

	introduced	1973	projected	Savings cost %	Savings time %
high energy physics	*******	**	****	10 - 30	30
low energy physics	**********	**	**	20 - 25	50
plasma physics	*	*			
nuclear chemistry	***	**	**	30	50
radiation chemistry (x-ray-diff.)	***	**			50
reactor control	*				
accelerator control	****	**	***	40	20 - 50 (several months)
nuclear medicine	*	*	*****		
other applications neutron spectroscopy, fission nuclear geology	***	*	*		
computer applications (data links, connection of peripheral equipm., networks)	*******	*****	************	10 - 40	50 - 60

CAMAC-APPLICATION IN NON-NUCLEAR FIELDS (30 QUESTIONNAIRES)

(Table V)

	introduced	1973	projected	Savings cost %	Savings time %
1. astronomy	*	**	**	30	40
2. biology	*				
3. computer aided design and teaching		**	***		
4. meteorology, microscopy logging, checkout routines space, oceanic	****	**	***		
5. chemical industry			*		
6. industrial laboratory automation	***	*	***		
7. process control	******	*	****	20 - 50	60; 2 years (industrial)
8. enviromental monitoring and control	****	***	***	20	
9. medical health service		****	***********	20	50

DISCUSSION

Q - J. van Gorp

- In your table V, point 4, you give the application of CAMAC in the meterological and oceanic field.
- Can you - or perhaps somebody else - give examples and experiences with such an application.
- In particular I have in mind "automatic weather stations", but it can also be a research project for example "boundary layer" research for reducing air pollution.

A - K. Zander

a) Meterologie:
In den Antworten auf die Fragebogenaktion wurden 3 Anwendungen erwähnt, eine davon aus Harwell (AERE, Grossbritannien).
Von einer anderen Anwendung berichtete die Firma Dornier (Friedrichshafen, Deutschland, Herr Mall). Diese befasst sich besonders mit Fragen des Umweltschutzes.
Die dritte Anwendung muss ich meinen Unterlagen in Berlin entnehmen.

b) Ozeanic Research
Eine solche Anwendung erwähnte ebenfalls die Firma Dornier.

Q - K. Killian

Können Sie die Gründe dafür nennen, warum, wie Ihre Tabelle zeigt, unter Punkt 9 - Medical Health Service - bei Vorhandensein keines im Routinebetrieb laufenden CAMAC-Systems in der Medizin das Schwergewicht in der Projektierung auf diesem Gebiet liegt?
Bisher laufen in Deutschland meines Wissens (aber ohne CAMAC):
-Lehrsysteme in: Tübingen, Erlangen, München, Hannover und Hamburg
- Ein Nuklearsystem in Bonn und
- Systeme für EKG in: Hannover, Giessen und München.

A - K. Zander

Meine Umfrage betraf ausschliesslich CAMAC-Anwendungen in der Medizin, keine anderen.
Das CAMAC-System wird jetzt erst in die verschiedenen Bereiche der Medizin eingeführt.

STATUS AND FUTURE OF CAMAC IN EASTERN EUROPE

R. Trechciński
Institute of Nuclear Research, Świerk, Poland

ABSTRACT

This review gives information on the state of the development, production and application of the CAMAC system in Eastern Europe countries (Bulgaria, Czechoslovakia, Poland, Roumania, Hungary, USSR) and on the collaboration of these countries in international organisations such as Joint Institute of Nuclear Research (Dubna), Nuclear Instrumentation Working Group of the Atomic Commission (Conecon), Interatominstrument (Warsaw). The paper contains some remarks concerning CAMAC system development in these countries in the nearest future.

ZUSAMMENFASSUNG

Es wird eine Übersicht gegeben über den Stand der Entwicklung, der Produktion und der Anwendung des CAMAC-Systems in den osteuropäischen Ländern (Bulgarien, Tschechoslowakei, Polen, Rumänien, UdSSR, Ungarn) und die Zusammenarbeit dieser Länder in internationalen Organisationen wie dem Joint Institute of Nuclear Research (Dubna), Nuclear Instrumentation Working Group der Atomkommission (Conecon),Interatominstrument (Warschau). Einige weitere Anmerkungen betreffen die Entwicklung des CAMAC-Einsatzes in diesen Ländern in der nächsten Zukunft.

RESUME

Cet exposé donne des informations sur l'état actuel des études, de la production et des applications du système CAMAC dans les pays d'Europe de l'Est (Bulgarie, Tchécoslovaquie, Pologne, Roumanie, Hongrie, URSS) et sur la participation de ces pays à des organisations internationales telles que l'Institut Commun de Recherche Nucléaire de Dubna, le groupe de travail Instrumentation Nucléaire de la Commission Atomique (Conecon), l'Interatominstrument de Varsovie. On fera quelques remarques relatives au développement du système CAMAC dans ces pays pour le proche avenir.

1. GENERAL

Beginning from 1971 Camac system is being introduced to the developments, production or applications in the Eastern Europe countries as: Bulgaria, Czechoslovakia, Poland, Roumania, Hungary, USSR.

/In USSR a system called "Vector" is being developed. It differs from the Camac system with a mechanical solution and the multipin connector applied to connect modules with dataway. All other features of the Camac system have been fully maintained in a "Vector" system./

The problems dealing with Camac system were included into the work programmes of the international organisations such as:

- Joint Institute of Nuclear Research /Dubna/,
- Nuclear Instrumentation Working Group of the Atomic Commission /Conecon/,
- Interatominstrument, Warsaw.

To the most significant international enterprises connected with a Camac system is to include :

- conference relating to collaboration problems in the field of a Camac system, Warsaw, October,23-25 1972 with participation of delegates from Bulgaria, Hungary, Czechoslovakia, Poland as well as Joint Institute of Nuclear Research and Interatominstrument ;
- symposium and exhibition on nuclear electronics /mainly Camac system/, Warsaw, April, 5-7, 1973, with 68 experts from Bulgaria, Hungary, East Germany, Poland, Roumania, USSR, Czechoslovakia and Interatominstrument ;
- VII international symposium on nuclear electronics organised by the Hungarian Academy of Science, Joint Institute of Nuclear Research /Dubna/, Budapest, September, 17-23, 1973 ;
- meetings of the Nuclear Instrumentation Working Group of the Atomic Commission /Conecon/ - twice during a year and experts meetings of a/m Working Group - twice during a year.

2. DEVELOPMENTS, PRODUCTION,APPLICATIONS

Bulgaria

Developments of several digital modules, among others development of the crate controller adapted to operation with a type EC 1020 computer are being carried out. It is expected that in the first period the Camac system will be applied to nuclear areas. The developments dealing with Camac system are being realised in a Bulgarian Academy of Science.

Czechoslovakia

About 40 digital and analogue modules are now being developed in the Institute of Nuclear Instrumentation. It may be expected that the production of these modules will start in 1975. The expected application - also outside nuclear technique. In addition to the above some works are carried out in the Institute of Nuclear Research of the Czechoslovak Academy of Science.

Hungary

About 25 modules of various types have been designed and recently are being produced. In 1972 10 crates /60 modules/ and in 1973 40 crates /300 modules/ for various applications were made. It is expected that Camac will equip a series of small type TPA computers, which are widely used also outside nuclear technique.

Roumania

Camac mechanics has been developed /crates, plug-in units/. A power supply and several modules are recently being designed. The expected application - nuclear technique.

USSR

Several Institutes for Research of the USSR Academy of Science are interested in Camac. No information is provided about the applications.

"Vector" system is being developed in the Union Instrumentation Institute for Science and Research - Moscow. Modules with the basic width of 20 mm are applied. A crate of 520 mm width has 23 stations for plug-in units. The 24th station is occupied by the AC switch, fuse and power-on indicator. In order to apply Camac modules in the Vector system a special adapter has been desiged. About 40 modules of various types have been designed. The production will start probably in 1975. The expected application - in various fields of science and technique. The developments of about 20 types of modules is still going on.

Poland

The mechanics /crates, plug-in units/ is produced since few years. The numbers cover all orders. The modules production has been initiated in 1972 /about 10 types/. In 1973 and in the next years about 20 modules of different types yearly are being passed on for production. About 60 types of various modules /analogue and digital ones/ are being recently elaborated. Camac system finds application outside nuclear technique in the mining, ship-building industry in power generating stations and so on.

Joint Institute of Nuclear Research - Dubna

In the Joint Institute of Nuclear Research about twenty various types of modules have been developed. These modules are being produced in the workshop of the Institute. System Camac is being applied in the Laboratory of High Energies and other laboratories with HP 2116B, TPA 1001, TPA-70 computers and other ones.

3. STANDARDISATION

Two international standardizing recommendations dealing with a Camac system are recently being developed within the Nuclear Instrumentation Working Group of the Atomic Commission /Conecon/. These recommendations cover the statements contained in the documents 4100 and partially 4600.

In addition, a Polish standard PN-73/T 06530 relating to Camac system has been issued. The further developments of the international standardizing recommendations and national standards are expected to be carried. In 1971 and 1972 several of the users wanted to introduce some changes into the Camac system. They were mainly relating to the type of the multipin edge connector and to the other constructional changes, which would be able to guarantee a better mechanical reliability. Recently these tendencies are not to be observed.

4. INTERNATIONAL COLLABORATION

Symposium and exhibition, Warsaw, April, 5-7, 1973

9 papers connected with Camac system were presented. Camac instrumentation was exhibited by: Czechoslovakia, Hungary, Roumania, USSR, Poland. Among others two working measuring assemblies with K202 computer /Poland/ and 1001-TPA /Hungary/ were presented.

Symposium, Budapest, September 17-23, 1973

About 30 papers dealing with Camac were delivered. They proved increasing interest in Camac.

Problems represented by the Nuclear Instrumentation Working Group of the Atomic Commission

The activity of the Working Group /and experts meetings/ apart from the standardizing topics is involved as well in the organisation of symposiums and exhibitions. The similar symposium and exhibition as in April 1973 in Warsaw will be organised in 1975.

From the other more important questions resolved in 1973 it should be mentioned the preparation of a list of computers types recommended to "on line" operation with the Camac system. This list contains now 12 types of computers manufactured by Bulgaria, Hungary, USSR, Czechoslovakia and Poland. The aim of this work is to secure production of the suitable controllers and branch drivers and to secure programming. It is expected as well that the Camac language and IML will be applied. In view of the fact that there several other digital systems of industrial automatics, to the work programme of the Working Group a problem to develop interfaces to join these systems with Camac has been introduced. The other question relates to the preparation of the preferential list of elements /mainly integrated circuits/ used in modules developments of Camac system. The application of these lists in practice will eliminate a number of elements types and will simplify service.

Interatominstrument

Interatominstrument as an organisation of commercial character will carry out among others the commercial information of the Camac system equipment. Interatominstrument will act as an intermediary among users, producers and research organisations.

5.FUTURE

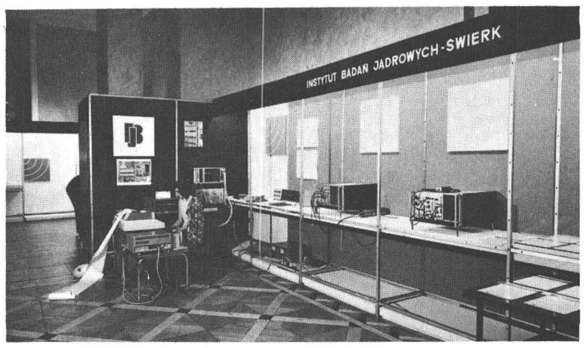
In the Eastern Europe countries a strong and a growing demand for the products of the Camac system is to be observed. To the main obstacle in the further effective development of applications it should be included the insufficient number of the suitable types of computers on the market. The progressive elimination of this main difficulty should in the nearest time affect the considerable growth of the number of applications.

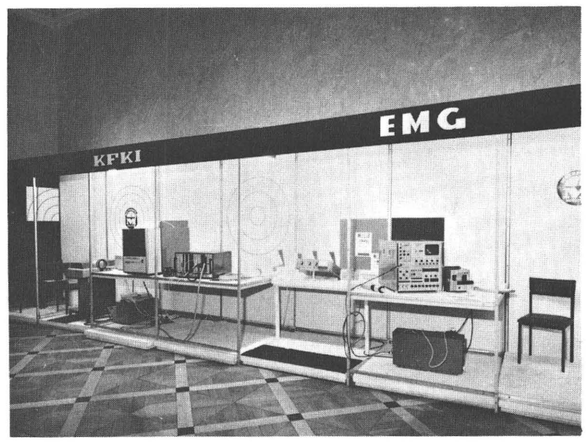
In the sphere of industrial automatics some interest is arising in the multicrate serial systems. The completion of these works in the ESONE Committee will create possibilities for the new Camac applications.

It is further a problem to get sufficient information on the ESONE Committee works. A small number of experts from the Eastern Europe countries in the Working Groups of ESONE Committee causes that the information relating to the new developments can be received by the interested users only after the issue of final documents.

The development of the Camac system in the Eastern Europe countries should in the nearest future affect the growth of a number of experts from these countries participating in the works of ESONE Committee Working Group.

It is to expect, that in the course of the nearest years about 80 % of the Camac system application will be employed outside nuclear technique.





Camac system products presented at the nuclear electronics exhibition, Warsaw, April, 5-7,1973

DISCUSSION

C - G. Chmielenski

My comment concerns developments of modules and CAMAC systems in the High Energy Laboratory of the Joint Institute of Nuclear Research in Dubna near MOSCOW.
More than 30 different types of modules have been designed, e.g. ADC's, scalers, code converters, DVM's, punch- and print-interfaces as well as crate system controllers for various computers as HP 2116B, TPA1, TPA70, BESM-4. Systems are in use for on-line data acquisition, processing and control in high energy physics experiments with the Dubna 10 Gev proton synchroton. A small, autonomous system of digital data recording has been also designed.
A data link, 1,2 km long, with a BESM-4 computer has been modified recently using CAMAC.

C - R. Trechcinski

The comment of G. Chmielenski is an extension of the information given in my paper. I would like to thank G.Chmielenski for this.

C - J. Böhm

CAMAC is used in Yugoslavia too.

C - R. Trechcinski

The comment is correct. Yugoslavia was a member of ESONE Committee a long time before Poland, Hungary, Romania. Therefore it was decided to include in my report only the information about new members of the ESONE Committee and new users. The status of CAMAC in Yugoslavia is very well known in West Europe.

STATUS AND FUTURE OF CAMAC IN NORTH AMERICA

R. S. Larsen
Stanford Linear Accelerator Center, Stanford University, Stanford, USA

ABSTRACT

The progress of CAMAC in North America is traced from the time of the endorsement of CAMAC by the NIM Committee in 1970, to the present. Developments in laboratory data acquisition and control, medical applications, and industrial measurement and control are described. Possible directions of future development are discussed.

ZUSAMMENFASSUNG

Der Vortrag verfolgt die Fortschritte von CAMAC in Nordamerika seit der Annahme von CAMAC 1970 durch das NIM- Komittee bis zur Gegenwart. Die Entwicklung bei der Datenerfassung und Steuerung in Laboratorien, bei medizinischen Anwendungen und in der industriellen Mess-, Steuer- und Regelungstechnik wird beschrieben. Mögliche Richtungen der zukünftigen Entwicklungen werden diskutiert.

RESUME

On retrace la progression de CAMAC en Amérique du Nord depuis l'adoption de cette norme par le Comité NIM en 1970, jusqu'à maintenant. On décrit les études dans les laboratoires, les applications médicales, les systèmes d'acquisition de données et de contrôle-commande ainsi que les systèmes industriels de mesure et commande. Les grandes lignes du développement futur sont discutées.

STATUS AND FUTURE OF CAMAC IN NORTH AMERICA*

R. S. Larsen
Stanford Linear Accelerator Center
Stanford University, Stanford, Calif. 94305

Abstract

The progress of CAMAC in North America is traced from the time of the endorsement of CAMAC by the NIM Committee in 1970, to the present. Developments in laboratory data acquisition and control, medical applications, and industrial measurement and control are described. Possible directions of future development are discussed.

Historical Background

The first large-scale instrument standardization effort among U. S. nuclear laboratories was the well-known NIM (Nuclear Instrument Module) system,[1] begun in 1964 and reaching full fruition about 1966. The AEC NIM Committee, sponsored by the AEC National Laboratories and National Bureau of Standards, and including representation from university research laboratories, concentrated on developing a standard which would arrest the rampant proliferation of incompatible hardware and signal standards taking place at that time. The success of NIM is history and will not be belabored here, except to note that the NIM program subsequently provided a most favorable climate for the introduction of CAMAC.

During the time that NIM was being successfully introduced to the AEC Laboratories, a number of digital modules, such as analog-to-digital converters (ADC's), time digitizers (TDC's), high speed scalers, and multi-channel coincidence latches, began to appear in module format. These designs usually consisted of a low density of digital electronics per module; thus the digital data, either serial or parallel, was transmitted via low density front or rear panel connectors conforming to no particular standard. A number of laboratories developed discrete wire-harness dataway systems specially designed to suit their particular needs. However, it was generally felt that NIM would never see usage as a high density digital interface system; thus the matter of a standard data connector for NIM was much discussed but never resolved.

CAMAC arrived at an opportune time, since new high speed integrated circuits were having a major impact on nuclear physics instrumentation, and a standard dataway was becoming imperative. The NIM Committee, which endorsed CAMAC officially in May, 1970,[2] viewed CAMAC as a timely adjunct to NIM: it felt that NIM would continue in popularity for high-speed circuits having little or no slow-speed digital I/0 requirements, and that CAMAC would serve admirably for data-collection modules, such as ADC's or scalers, having many digital outputs. In the U. S. nuclear laboratories today, NIM and CAMAC systems are indeed usually integrated in a typical experiment, partly of course because of the presence of large stocks of NIM equipment which existed prior to the advent of CAMAC.

In the same way that the NIM system was introduced and promoted, CAMAC has been presented by the NIM Committee to both users and industry and actively promoted as a desirable standard. The NIM Committee formed several sub-committees, parallelling the ESONE organization, to help develop the standards further; these include an Executive group for CAMAC, and Dataway, Software, Mechanical and Analog working groups. The purpose of these groups is to work with, as well as in parallel with, the corresponding ESONE groups to develop further standards in areas presently undefined or unspecified; and especially to interpret and clarify specifications for users, as well as industry, including specification of preferred practices where options are possible.

Working Group Activities

The various working groups have maintained very close liason with ESONE in all of their efforts. The earliest activities of the working groups centered around participating in the revision of EUR-4100e, and producing National Bureau of Standards equivalents of the main CAMAC specifications.[3,4] The Dataway and Mechanical groups have also cooperated on a Supplement[5] which contains notes on preferred practices, and includes such items as dataway crosstalk measurements and a typical power supply specification. The most recent activity of the Dataway group revolves about specification of the Serial Crate Controller. The serial controller is an important question and it is hoped that a final specification acceptable to both NIM and ESONE is near at hand.

The Software group is considering the problems of CAMAC standardized software on a broad front.[6] Its major tangible product to date has been the endorsement of six recommended CAMAC subroutines[7] suitable for use with, but not restricted to, the FORTRAN programming language. These subroutines involve essentially the generation of CAMAC commands which reflect the complete generality of CAMAC hardware, and which hopefully can be used with a variety of procedural and assembly languages. The subroutines include such titles as CMCBSC (any legal command to a branch, single module or multi-crate address, repeatable a specified number of times); CMCASC (address scan using the Q-response mode); and so on.

The Software group is also cooperating with the ESONE-initiated effort to specify a high-level language for the definition of CAMAC systems and procedures. Progress in this area is being coordinated and interpreted via meetings and reports such as (6, 7) previously cited.

The Analog group has collaborated in a series of proposals leading up to the currently proposed revision of EUR5100 (1972) "Specification of Amplitude Analogue Signals."[8]

Some promotional activities involving other organizations have taken place or have been scheduled. Committee or Working group members have exchanged meetings with the Purdue Workshop on Automatic Control, a group involved in instrumentation standards in the industrial control field. In addition, some CAMAC equipment will be demonstrated at the IEEE Industrial Applications Society meeting, October 8-11, 1973, in Milwaukee, Wisconsin. Also in October, the Dataway Group chairman will participate in a panel called "Computer Industry Standardization Efforts—Their Impact on Process Control" at the Instrument Society of America (ISA) Conference, October 15-18, 1973, in Houston, Texas.

* Work supported by the U. S. Atomic Energy Commission.

CAMAC In Laboratory Data Acquisition, Measurement and Control

The greatest impact of CAMAC to date of course has been in the nuclear laboratories. In the established laboratories, where data acquisition systems and interfaces already existed, the main impact has been at the module and crate, rather than the branch, level; that is, CAMAC equipment has been interfaced to existing systems via simplified branch drivers and sometimes simplified crate controllers in systems requiring usually a read-only operation. In these laboratories, implementation of full CAMAC data acquisition and control systems has been slow, mainly because of this common requirement to interface with older non-CAMAC equipment.

In the newer laboratories such as the National Accelerator Laboratory (NAL), Los Alamos Meson Physics Facility (LAMPF), and Tri-University Meson Facility (TRIUMF), where clear choices could be more readily made, CAMAC has enjoyed wide application and a growing popularity in both data acquisition and control. At NAL, for example, CAMAC is used for complete experimental data acquisition systems, as well as monitoring and control of the extracted proton primary beam and meson and neutrino secondary beams.[9, 10, 11, 12] Additionally, the Synchrotron Booster control system uses CAMAC for control and monitoring of magnet power supplies, etc.[13] Because of the extremely long distances involved at NAL (3-mile circumference of the 400 GeV main ring), it has been necessary to develop serial crate controllers[14, 15] which are now in wide use. A recent order has been placed by NAL for fifty serial controllers based on the existing NIM-ESONE draft specification for a CAMAC standard serial crate controller. Approximately 250 CAMAC crates are presently in use at NAL.

At LAMPF,[16] similarly, CAMAC systems are used in conjunction with minicomputers for data acquisition and control in each secondary beamline; for some special primary beamline diagnostics systems; and for data links between satellite computers and a main control computer. Most of the present systems utilize a Microprogrammed Branch Driver[17] (MBD) interface to provide a very powerful and versatile system, while some of the simpler systems use single crate (Type U) interface controllers.

At the TRIUMF laboratory, CAMAC based on the Elliott Executive System is used to interface a central control console to a SuperNova computer; this computer is in turn linked to a second SuperNova having six CAMAC branches to separate remote control console areas. These remote areas control such items as the ion source, injection, ring magnets, etc. Ultimately the entire machine control system will be operated from the central console. A special optically-coupled serial system[18] has been developed for control of the ion source, which operates at a potential of 300 KV. About 30 CAMAC crates are involved in this system.

These and other laboratory activities have stimulated the development of new CAMAC hardware among manufacturers in North America. Essentially all the important module and controller functions, as well as crates and power supplies, can be purchased from domestic sources. This includes a variety of branch drivers, Type A and Type U controllers, multi-channel ADC, TDC, and register modules, relay control and multiplexer modules, etc. Further details of specific components can be found in the various CAMAC products guides such as the periodic listings in the CAMAC Bulletin.

The laboratories themselves have developed special modules and readout schemes for such devices as proportional and magnetostrictive wire chambers,[19, 20] scalers and scaler displays,[21, 22] magnetic tape interfacing,[23] etc., to name but a few examples.

In addition to the National Laboratories, CAMAC has been introduced into many user groups at major universities throughout the United States and Canada.

One non-nuclear laboratory where CAMAC is being broadly implemented is at Kitt Peak National Observatory (KPNO) in Tucson, Arizona. Here, CAMAC is being used in a new installation to interface a total of eleven telescopes to individual control and data acquisition minicomputers.[24] A special branch driver has been designed for this system.[25] To date, six of the eleven telescopes are fully operational.

Another observatory using a small CAMAC installation is the Lick Observatory, operated by the University of California at Santa Cruz, California.

CAMAC In Medical Applications

CAMAC has had some significant beginnings in the field of medical computing. At Vanderbilt University, CAMAC has been used to implement a nuclear detector scanning system using a small computer.[26, 27] The computer both programs and controls the scanning table, as well as processes the data and generates visual displays. A variety of different detectors can be accommodated; in addition an Anger type camera is being implemented. In future, this group is also proposing to utilize CAMAC in developing a computer controlled radiological therapeutic facility.

At the Los Alamos Meson Physics Facility (LAMPF), a biomedical group is in the process of implementing a system for pion radiotherapy.[28] This system, consisting of a Microprogrammed Branch Driver (MBD) connecting three CAMAC crates to a minicomputer, will use the computer real-time operating system for control of treatment and planning as well as for beamline diagnostics.

At the University of Toronto Department of Medicine, CAMAC has been used to implement a large data acquisition system involving a 128K core time-shared central computer serving a total of five individual medical research laboratories over a standard CAMAC branch.[29] The remote crates in turn are used to interface to local minicomputers through autonomous controllers for real-time high speed data acquisition. The central and local computers communicate through a special Link Module. A local CRT display is also interfaced via CAMAC. The system also provides background batch processing in the central computer. A balanced long-distance branch, as well as a serial crate controller, have also been implemented. The system is used to support various experiments in neurophysiology, pharmacology, otolaryngology, and biochemistry.

More recently, two independent CAMAC systems are being installed in the laboratories of the Zoology Department, for behavioral experiments and neurophysiological monitoring, and in the Addiction Research Center, similarly for neurophysiological studies.[30] The former system involves three minicomputers sharing a CAMAC branch, while the latter involves a single minicomputer.

CAMAC In Industrial Applications

CAMAC has made some modest beginnings in the field of industrial control and measurement. A prototype system being implemented to control and monitor natural gas demands for an aluminum plant is to be described later in this conference.[31] As has already been mentioned, CAMAC is being studied by the Purdue Workshop on Automatic Control, and will be discussed by a panel at the aforementioned ISA Conference in October. The planned demonstration of CAMAC equipment at the October IEEE Industrial Applications Society meeting is also significant.

The application of CAMAC to industrial environments introduces new problems of electrical noise, isolation of high voltages and currents, suitability of current software developments to industrial applications, and overall system reliability. It is clear that although a few of these industrial environmental problems have received attention in the National Laboratories in applications such as the control and monitoring of very high current power supplies or high voltage ion sources, the operational requirements for CAMAC in a non-laboratory environment are in general much more stringent, and some important questions remain to be resolved.

Future of CAMAC

The immediate future of CAMAC in nuclear as well as other laboratories seems assured. CAMAC, because it has been a well-engineered system from its inception, has indeed become the desired standard for future data acquisition and control systems. The development of commercial controllers and branch drivers for a wide selection of computers, and at reasonable prices, has been instrumental in promoting this trend. As more practical operating systems are demonstrated, the advantages of CAMAC become obvious even to those who may initially oppose the system. The major point in favor of CAMAC is that in many cases it offers an off-the-shelf solution to complete data acquisition or control system problems, thus relieving the experimental user of much repetitive and costly design effort. This feature is of particular value to small and medium sized user groups who usually have minimum engineering resources at their disposal, and also in large laboratories where many similar but flexible systems must be implemented.

In medical applications of CAMAC, the situation seems to encompass features of both laboratory and industrial environments. In general, it appears that CAMAC is capable of solving a host of problems in medical system monitoring and control; however from a user standpoint the need for reliability and availability of outside maintenance and engineering services would appear even more important than in many industrial applications, both because of the possible interruption of a vital medical service, as well as because of a possible complete lack of in-house technical service or engineering personnel. Hospital personnel in general are not electronics specialists; hence the practical problems of operating and supporting sophisticated electronics equipment in such an environment must be met as a prerequisite to widespread acceptance of CAMAC as an instrumentation system.

Similarly, certain environmental problems must be solved if CAMAC is to receive widespread popularity in the industrial control field. The situation here is somewhat better in that the tasks of automatic control and measurement are relatively straightforward from the computer standpoint, and a minimum interaction with technically unskilled personnel is necessary in many applications. CAMAC appears capable of meeting the environmental requirements, possibly needing some expansion of existing hardware options to do so;* however its success will depend mainly upon the development efforts of independent industrial users such as Alcoa. Again, such efforts require teams of engineering personnel, and many industries will not see CAMAC until complete systems engineering and servicing are available commercially because of an in-house lack of such personnel.

It should be noted in all of the foregoing that there are no indications that CAMAC is incapable of meeting the technical requirements in all of the fields mentioned, only that certain practical conditions need to be met for CAMAC to continue to expand into these and other areas of application. Thus the problems mentioned apply equally well to any competitive system or systems which would hope to operate in these areas, and CAMAC may well have a significant advantage because of its current widespread acceptance in the nuclear field.

In considering the long-range future of CAMAC, the question of its suitability as a packaging standard for future integrated circuits inevitably is raised. Even in the very beginnings of CAMAC it was argued by some that the CAMAC packaging concept was too restrictive for large-scale integrated circuits (LSI) and hence would be quickly doomed to failure. That CAMAC has in fact become a practical and economic success in many areas, with much promise of further success, simply demonstrates that such standards do serve a useful purpose even in the face of change, due to the finite time lag between initial development and widespread availability and utilization of new, more complex families of integrated circuits. Furthermore, the advent of higher density, more complex integrated circuits does not automatically outmode a given packaging scheme, since real systems are more often influenced by other constraints such as the available connector, cable or mechanical hardware, than by the introduction of a new configuration of integrated circuit into the system.

Still, it must be recognized that just as CAMAC initially received impetus from new developments in integrated circuits, so also CAMAC may eventually become obsolete because of the advent of totally new logic packaging or organizational concepts at the device level. If and when this happens will be dictated by compelling technical and economic factors. Whatever the outcome, however, the lessons of NIM and CAMAC will surely serve as an invaluable model for future standardization efforts in the field of instrumentation.

Acknowledgment

This brief status report could not possibly convey the extend of the effort required by the NIM-CAMAC groups to bring CAMAC to its present status in North America; nor could it properly convey the extent of the mutual benefits derived from the NIM-ESONE collaboration. We on the NIM Committee are grateful for past associations with our ESONE colleagues and pleased to share in the continuing successful development of CAMAC.

References

1. "Standard Nuclear Instrument Modules", L. Costrell, National Bureau of Standards, USAEC, TID 20893 (Rev. 3), December, 1969.
2. See p. v, Section B, Reference 3.

* It should be noted that in an electrically harsh environment, certain problems of machine control interfacing may be more readily solved using a NIM module; thus the potentiality for combined NIM-CAMAC systems should be considered for such applications.

3. "CAMAC, A Modular Instrumentation System for Data Handling", USAEC TID 25875, July, 1972.
4. "CAMAC, Organization of Multi-Crate System", USAEC TID 25876, March, 1972.
5. "Supplementary Information on CAMAC Instrumentation System", USAEC TID 25877, December, 1972.
6. "Standard Software for CAMAC", S. Dhawan and R. F. Thomas, IEEE Trans. Nucl. Sci., NS-20, No. 1, February, 1973.
7. "Specification for CAMAC Subroutines", R. F. Thomas, LAMPF Report LA-5059-MS, October, 1972.
8. "CAMAC...Specification of Amplitude Analogue Signals", Draft Text for EUR5100e (rev.), ESONE Working Group on Analogue Signals, Geel, Belgium, July 6, 1973.
9. "CAMAC Experimental Beam Line Control System", L. J. Hepinstall, E. J. Barsotti and H. C. Lau, IEEE Trans. Nucl. Sci., NS-20, No. 3, June, 1973.
10. "A Modular Control System Utilizing CAMAC Hardware for Interfacing Remote Components at NAL", W. H. Deluca, et al., IEEE Trans. Nucl. Sci., NS-20, No. 3, June, 1973.
11. "CAMAC Stand-Alone Control System (CAMSAC)", R. G. Martin, to be published in IEEE Proceedings of the 1973.
12. "Digital Monitoring System for NAL Secondary Beam Line Instrumentation", R. G. Martin, ibid.
13. "An Accelerator Instrumentation and Control System Using CAMAC", L.A. Klaisner and J. M. Stephenson, Jr., IEEE Trans. Nucl. Sci., vol. NS-18, No. 1, February, 1971.
14. "CAMAC Crate Controller for a Serial Control System", E. J. Barsotti, IEEE Trans. Nucl. Sci., vol. NS-20, No. 3, June, 1973.
15. "Operational Aspects of a Serial CAMAC System", E. J. Barsotti et al., to be published in IEEE Proceedings of the 1973 Nuclear Science Symposium, San Francisco, November 14-16, 1973.
16. Private Communication, D. R. Machen, LAMPF. Summary report to be published in a future issue of the CAMAC Bulletin.
17. "A Long-Distance CAMAC Branch via Data Link and Microprogrammed Branch Driver", L. Biswell et al., IEEE Trans. Nucl. Sci., vol. NS-20, No. 1, February, 1973.
18. "An Optically-Coupled Serial CAMAC System", D. R. Heywood and B. Ozzard, to be published in IEEE Proceedings of the 1973 Nuclear Science Symposium, San Francisco, November 14-16, 1973.
19. "Fast Readout and Geometrical Reconstruction System for Proportional Wire Chambers", S. Dhawan, ibid.
20. "CAMAC Magnetostrictive Readout System Using Shottky Memories", B. Bertolucci, IEEE Trans. Nucl. Sci., NS-20, No. 1, February, 1973.
21. "Collaboration on a 100 MHz CAMAC Scaler Specification", Dale Horelick, SLAC-TN-70-13 (Rev), LBL Spec. LRL-03-34A, (Rev), September, 1970.
22. "Versatile, Modular Readout System for CAMAC Scalers", D. Horelick, IEEE Trans. Nucl. Sci., NS-20, No. 1, February, 1973.
23. "A Variable Wordsize CAMAC Tape Controller", A. Oakes et al., ibid.
24. Private Communication, L. Paffrath, Kitt Peak National Observatory.
25. "A CAMAC Branch Driver for the Varian 620 Series Computers", Lorne B. Mortora, IEEE Trans. Nucl. Sci., NS-20, No. 1, February, 1973.
26. Private Communication, A. B. Brill and J. Patton, Vanderbilt University.
27. "CAMAC Applications in Nuclear Medicine at Vanderbilt: Present Status and Future Plans", A. B. Brill, et al., to be published in IEEE Proceedings of the 1973 Nuclear Science Symposium, San Francisco, November 14-16, 1973.
28. Private Communication, D. R. Machen, LAMPF, Los Alamos, N.M.
29. "MEDAC-CAMAC, A CAMAC System for Medical Data Acquisition and Control", William K. B. Sie and John N. T. Potvin, CAMAC Bulletin No. 7, July, 1973.
30. Private Communication, J. N. T. Potvin, Faculty of Medicine, University of Toronto.
31. "An Evaluation of CAMAC Equipment in Industrial Environment", W. T. Lyon, ALCOA, Paper to be presented at First International Symposium on CAMAC in Real Time Computer Applications, Luxembourg, December 4-6, 1973.

NORMALISATION INTERNATIONALE DANS LE DOMAINE DES SYSTEMES MODULAIRES POUR LE TRAITEMENT DE L'INFORMATION

A. Rys
Président du Comité d'Etudes 45 de la CEI, Paris, France

RESUME

Les travaux de Normalisation entrepris dès 1963 par la CEI ont abouti à deux publications internationales relatives aux dimensions des châssis et aux signaux.

L'apparation du CAMAC a conduit à inclure le tiroir et le châssis CAMAC dans deux autres publications en cours et à décider une publication globale reprenant tout le système CAMAC : châssis, tiroirs et interface qui devrait pouvoir être éditée en 1975.

ZUSAMMENFASSUNG

Die Normungsarbeit der IEC auf diesem Gebiet begann 1963 mit der Ausarbeitung zweier internationaler Normen über Abmessungen von Chassis und elektrische Signale.

CAMAC wurde zunächst dadurch berücksichtigt, dass CAMAC-Rahmen und -Einschübe in zwei weitere Veröffentlichungen aufgenommen wurden. Danach wurde entschieden, eine einzige Veröffentlichung herauszugeben, die das gesamte CAMAC-System mit Rahmen, Einschüben und Schnittstellenspezifikationen umfassen soll. Ihr Erscheinen ist für 1975 vorgesehen.

ABSTRACT

The work of the IEC on standardization in this area, started in 1963, has generated two publications related to chassis dimensions and signals.

CAMAC has been introduced by including the CAMAC unit and chassis in two documents and subsequently it has been decided that a single document should deal with the complete CAMAC system (units, chassis and interface) and should be issued in 1975.

1-LA CEI

L'organisme officiellement chargé de la normalisation sur le plan international est l'Association Internationale de Normalisation dont l'appellation anglaise abrégée est ISO. L'ISO a délégué ses pouvoirs dans le domaine de l'électricité et de l'électronique à un autre organisme affilié à l'ISO la Commission Electrotechnique Internationale (CEI). La CEI comporte de nombreux comités d'études (actuellement 77) travaillant chacun dans un domaine déterminé.

La normalisation des systèmes modulaires qui couvre un domaine très vaste aurait pu être entreprise par différents Comités d'Etudes.

Il s'est trouvé, comme pour le CAMAC que c'est dans le domaine nucléaire que la nécessité d'une normalisation est apparue le plus vite.

C'est donc le Comité d'Etudes n° 45: "Instrumentation Nucléaire" de la CEI qui a le premier démarré les travaux de normalisation dans cette voie.

2-TRAVAUX ANTERIEURS AU CAMAC

Les travaux entrepris ont porté chronologiquement sur les sujets suivants : châssis, signaux, tiroirs.

2-1- Concernant les châssis, le CE 45 émettait en 1963 un premier document international fixant certaines dimensions des châssis 19 pouces. En 1965 un autre document complétait ces dimensions et il fut décidé de fusionner ces deux projets qui aboutirent en 1969 à la Publication CEI n° 297 : Dimensions des Panneaux et Bâtis (pour appareils d'électronique nucléaire).

Malgré la restriction apportée dans ce titre, une préface indiquait que le document était également applicable à d'autres domaines que le nucléaire.

En raison de cette extension la suite des travaux sur ce sujet fut confiée au Comité d'Etudes n° 48 "Composants électromécaniques pour équipements électroniques".

Ce Comité a révisé la Publication 297 et le nouveau projet vient d'être soumis au vote des différents pays membres de la CEI.

2-2- Concernant les signaux, le CE 45 émettait un premier document international en 1966 qui aboutissait en 1969 à la Publication CEI n° 323 "Domaines de tension analogique et niveaux logiques pour appareils nucléaires alimentés par le réseau". Les niveaux logiques pour signaux binaires définis dans ce document sont respectés par le CAMAC sauf une légère distorsion concernant la tension de sortie du niveau bas.

La Publication 323 va faire l'objet d'une révision dont le texte, actuellement soumis au vote des pays membres de la CEI, est maintenant rigoureusement en accord avec le CAMAC sur ce point.

2-3- Concernant les tiroirs, deux documents internationaux étaient émis en 1968 l'un concernant le tiroir NIM et l'autre concernant un système à base 20 mm en usage dans les pays de l'Est.

En réponse, les pays faisant partie du Comité Esone demandèrent l'inclusion dans cette normalisation du tiroir CAMAC.

Un document définissant ces 3 tiroirs fut alors publié et sa dernière version est actuellement soumise au vote des pays membres de la CEI sous le titre "Dimensions des tiroirs d'appareils électroniques (pour appareils d'électronique nucléaire)".

3-TRAVAUX SUR LE CAMAC

3-1- L'apparition du document EUR 4100 en 1969 et le succès indéniable rencontré dans le monde par le CAMAC ne pouvaient pas laisser la CEI indifférente. La question fut évoquée pour la première fois en mai 1969 lors de la réunion de l'ACET (Advisory Committee on Electronics and Telecommunications).Ce Comité placé sous les auspices du Président de la CEI réunit les Présidents et les Secrétaires de tous les Comités de la CEI qui s'occupent d'électronique (une vingtaine).Son but est de coordonner les travaux de ces Comités et de recommander des actions dans des domaines déterminés.

C'est à sa réunion suivante en décembre 1970 que l'ACET prit la décision d'établir la liaison technique avec le Comité Esone en vue de la normalisation du CAMAC et chargea le CE 45 de cette liaison.En fait, comme on l'a vu précédemment, la liaison existait déjà puisque plusieurs membres du Comité Esone travaillaient au CE 45 et souhaitaient voir la CEI entériner les travaux sur le CAMAC. Appliquant la décision de l'ACET, le CE 45, lors de sa réunion de Bucarest en 1971, désignait M. BISBY alors Président du Comité Esone pour assurer la liaison entre ESONE et la CEI.

3-2- Lors de la réunion du CE 45 à Londres en 1972, un groupe de travail ad-hoc présidé par M. BISBY se réunit avec mission de déterminer les parties du document EUR 4100 qui pouvaient faire l'objet d'une recommandation internationale de la CEI. Les conclusions de ce groupe ad-hoc furent qu'en plus du tiroir CAMAC déjà prévu dans le document mentionné ci-dessus en 2-3, il y avait lieu d'établir deux autres documents, un sur les châssis et un sur les interfaces. M. BISBY fut chargé de les rédiger en accord avec le Comité Esone. Ces 2 documents furent prêts en février 1973.

Il fut décidé que le document sur les châssis qui définissait le châssis NIM et le châssis CAMAC serait examiné par le groupe de travail habilité du CE 45 avant diffusion aux pays membres de la CEI. Cet examen eut lieu à la Haye en octobre 1973 et le document légèrement amendé va être diffusé incessamment.

Le document sur les interfaces reprenait les prescriptions obligatoires du nouveau document EUR 4100, version révisée de 1972. Il fut envoyé au Bureau Central, organe directeur de la CEI en lui demandant de désigner le Comité d'Etudes qui serait chargé de ce travail.

Pour ne pas perdre de temps, le Bureau Central diffusa aussitôt le document aux pays membres de la CEI dans le cadre du CE 45 en informant également les autres Comités intéressés.

4-PERSPECTIVES D'AVENIR

4-1- En raison de l'importance de plus en plus grande prise par le problème des bâtis et panneaux, le Comité Directeur de la CEI vient d'organiser une réunion consultative pour discuter de la répartition du travail dans ce domaine.

Cette réunion vient de se tenir à Zurich en septembre dernier et groupait les représentants de 7 Comités d'Etudes intéressés par ces problèmes.

Les décisions suivantes furent prises :

a) Pour ne pas perdre de temps, le CE 45 terminerait les travaux entrepris dans le cadre du nucléaire en préparant 3 publications : Une sur les tiroirs faisant suite au document dont il est question en 2-3-.

Une deuxième sur les chassis NIM et CAMAC et une troisième décrivant l'ensemble du CAMAC conformément au document EUR 4100.

b) Un sous-comité serait créé au sein du Comité d'Etudes n° 48 qui, mis à part la terminaison des travaux du CE 45, aurait désormais l'exclusivité des travaux dans ce domaine.

Dans un premier temps, son activité serait limitée aux problèmes dimensionnels : Il aurait à reprendre les publications du CE 45 pour déterminer dans quelle mesure elles sont applicables à d'autres activités que le nucléaire et à étudier d'autres standards éventuels, son étude ne se limitant pas aux châssis de 19 pouces.

Par la suite le domaine d'activité de ce sous-comité pourra être étendu à d'autres problèmes liés aux bâtis et panneaux par exemple celui des cartes imprimées. Toutefois, il a été précisé que les questions d'affectation des contacts pour des utilisations spécifiques (comme pour le CAMAC) resteraient du domaine des Comités spécialisés pour leurs besoins particuliers.

4-2- D'autre part le CE 45 s'est réuni à La Haye en octobre dernier et appliquant les décisions du groupe consultatif ci-dessus a décidé d'accélérer au maximum les Publications qui lui sont demandées. Des mesures seront prises, pour le CAMAC notamment, pour activer au maximum les procédures.

5-CONCLUSION

Compte tenu néanmoins du délai nécessaire à la rédaction du nouveau projet et à sa diffusion, du temps de réflexion de 6 mois laissé obligatoirement à tous les pays membres de la CEI pour faire part de leur vote et de leurs observations et du temps nécessaire à l'édition de la Publication, on peut espérer raisonnablement que le CAMAC deviendra une Recommandation CEI en 1975.

Ce délai paraîtra certainement très long à tous ceux qui depuis des années ont travaillé pour mettre au point le CAMAC. Mais la longueur des procédures de normalisation internationale est nécessaire pour obtenir l'accord le plus large donnant aux normes toute leur valeur.

Je ne m'étendrai pas sur l'intérêt supplémentaire que la normalisation CEI va apporter au CAMAC en élargissant considérablement le cercle déjà important des sympathisants du Comité Esone.

Je voudrais pour terminer exprimer au nom de la CEI ma gratitude à ce Comité.

Il n'est pas courant que des utilisateurs sentant le besoin d'une normalisation s'unissent d'eux-mêmes pour la créer et surtout y parviennent avec un tel succès.

Cette normalisation "sauvage" qui va maintenant rentrer dans le cadre officiel a permis certainement de gagner plusieurs années sur les phases préliminaires des procédures habituelles et à mon avis elle doit être encouragée. Je me permets donc de féliciter le Comité Esone et d'espérer que d'autres documents pourront être édités de la même façon.

DISCUSSION

Q - L. Richard

Puis-je demander à l'orateur de nous dire quelques mots sur le CAMAC et les accords CENELCOM visant à l'harmonisation des normes nationales, signées par les 13 gouvernements de la Communauté Economique Européenne.

A - A. Rys

L'harmonisation des normes nationales pose des problèmes au CENELCOM lorsque ces normes existent et qu'elles sont divergentes.
Ce n'est pas le cas du CAMAC puisque la Normalisation débute à l'échelon international.
Il est probable que la plupart des pays de la CEE voteront pour le CAMAC et qu'aucun ne sera contre.
Or quand un pays vote oui pour une Recommandation Internationale, il s'engage à adopter cette Recommandation dans ses normes nationales.
Il est donc improbable qu'il puisse y avoir divergence dans les normes nationales et vraisemblablement le CENELCOM n'aura pas à s'occuper du CAMAC sauf pour constater, au moins dans ce domaine, qu'il n'y a pas de problème.

II

SOFTWARE FÜR CAMAC DV ANWENDUNGEN

SOFTWARE FOR CAMAC COMPUTER APPLICATIONS

SOFTWARE DANS LES SYSTÈMES CAMAC AVEC CALCULATEUR

Präsident - Chairman - Président:

I.C. PYLE, University of York, Heslington, England

INTRODUCTION TO CAMAC SOFTWARE

I. N. Hooton
A. E. R. E. , Harwell, England

ABSTRACT

The ESONE- and NIM-CAMAC Software working groups have selected the basic facilities of CAMAC Software which should be provided for the general user. These have been embodied in the semantic definitions of the CAMAC Intermediate Languages (IML). The facilities of IML are described and discussed as a basis for Software standardisation.

ZUSAMMENFASSUNG

Die ESONE- und NIM-CAMAC Software Arbeitsgruppen haben die grundlegenden Eigenschaften von Programmen ausgewählt, welche in Zusammenhang mit CAMAC dem Anwender generell zur Verfügung stehen sollten. Die Eigenschaften wurden bei der Festlegung der Semantik Definitionen der "CAMAC Intermediate Language" (IML) berücksichtigt. Die Möglichkeiten von IML werden beschrieben und die Verwendung von IML als Grundlage für eine Software Standardisierung wird erörtert.

RESUME

Les groupes de travail software ESONE, software NIM-CAMAC ont sélectionné les caractéristiques du software CAMAC qui doivent être fournies à l'utilisateur, en général. Ces caractéristiques ont été incorporées dans les définitions sémantiques du"CAMAC Intermediate Language" (IML). Définition des caractéristiques d'IML et discussion de leur utilisation comme base de normalisation du software.

1. Introduction

Camac software may be defined as the programming statements which enable the user to specify Camac activities without reference to the mechanism of the Camac-computer coupler. In this paper attention will be focussed on the facilities required rather than on the means of expressing them. The ESONE Software Working Group, in close collaboration with the NIM-CAMAC Software Working Group, has made a selection of the facilities which should be provided. This will shortly be published as the semantic definition of a language currently known as the Camac Intermediate Language, or IML. Work is in progress on a macro-expansion syntax, a translator syntax and a sub routine-call syntax which implement the semantics of IML. The publication of the semantic definition will be accompanied by, at least, the macro-expansion syntax. This version of IML has already been implemented.

The following sections give an informal description of the facilities in IML. Examples are expressed in the macro-expansion syntax.

2. The Camac Intermediate Language

IML is a low level language which provides the first level of software standardisation above the hardware specifications of Camac. It is intended to be used in conjunction with existing programming systems, giving the additional facilities required to incorporate Camac into the total computer system.

The statements in IML may be divided into two general classes, action statements and declaration statements. Action statements are executed at run time and make use of symbolic names that have been previously declared.

2.1 Declarations

IML permits local and global declarations. A distinction is made between entities declared in the current program which may be used elsewhere (Export) and those declared elsewhere which are referred to in the current program (Import).

2.1.1 Device declarations

The most important entities in Camac are devices and demands. Devices are accessed at a Camac address, that is at a subaddress (A) of a station-number (N) in a crate (C) on a branch (B). Devices may be referenced immediately, indirectly through a pointer, or as an element of a hardware array. The indirect reference allows the run time program to change the Camac address. The hardware arrays allowed in IML are:-

i) a random distribution of devices (given-array).

ii) a regular distribution of devices (calculated-array).

iii) an array accessed in the address scan mode of EUR4100e (Q-scan array).

A typical local device declaration in IML macro-expansion syntax is:-

```
LOCD    TTYPE,H,1,2,3,0
```

This declares the devise name TTYPE (teletype) as an immediate reference (H) to the device at Camac address branch 1, crate 2, station 3 and subaddress 0.

2.1.2 Demand declarations

Camac demands have two aspects. The first is the source of the L-signal in the module. According to the hardware specifications this may be either at a subaddress or at a bit position in a group 2 register. The Camac function codes required to operate with these two cases are different. By providing two forms of declaration IML is able to have identical action statements for accesses to demands of either form.

The second aspect of a demand is its access to the computer system. IML allows the user to declare the 'Graded-L number' associated with a demand.

Examples of the two forms of local L-declarations are

```
LOCL    NEXT,GL17,SUB,1,2,3,4
```

and

```
LOCL    DONE,GL15,BIT,8,1,3,4
```

NEXT is the demand controlled at sub-address 4 of the teletype interface module and associated with graded-L 17. DONE is a demand accessed at bit position 8 in the L-register in the module at Camac address branch 1, crate 3, station 4. It is associated with graded-L 15.

2.1.3 Memory reference declarations

Memory references used in IML action statements must have been previously declared. IML requires the word length (number of bits needed to hold the data) to be declared. Memory references must conform to the host programming system. IML allows for constants, variables, arrays and buffers. An example of a local variable declaration is :-

```
LOCM    LOCN,V,24
```

2.2 Action Statements

Action statements are divided into single-action, block-transfer, multiple-action, module status and system statements.

2.2.1 Single-action statements

The IML single-action statement is the fundamental mechanism for calling for Camac activity. It defines a single operation in the Camac hardware and can access any single Camac address with any of the Camac function codes.

If a data transfer is involved a memory reference is also required. In IML macro-expansion syntax typical statements are

```
SA  READ,TTYPE,LOCN
SA  FØ,TTYPE,LOCN
SA  ENABLE,TTYPE
```

The first two statements are identical (READ being equivalent to function code zero). The data in the device TTYPE is transferred to the memory reference LOCN. The third statement enables the device TTYPE.

Any action to be taken on the Q Response to this command must be defined in a subsequent statement. For ease of programming a version of the single action statement with a subsequent jump on Q (SJQ) or jump on not Q (SJNQ) has been included in IML.

```
SJNQ  READ,TTYPE,LOCN,AGAIN
```

If Q=0 is received when this instruction is executed the program branches to the label AGAIN. If this statement were itself labelled AGAIN this would implement the "repeat mode" of EUR4100e.

2.2.2 Block-transfer statements

In the above examples a word has been read from a teletype interface. The "repeat mode" allows the operation to be repeated until a successful transfer is achieved. However a common requirement is for successive characters to be read into an area of memory each time a demand is received from the teletype interface module. When a preset number of characters, or carriage return, is received the total transaction is terminated and the collected information is processed. IML provides an asynchronous block transfer statement for this type of situation.

```
BLQ  READ,TTYPE,BUFF,REP72,NEXT,CH2
```

- BLQ is a block transfer in which each transfer is initiated by an L-signal from the module via the graded-L structure. It is terminated by Q=0 from the module or by the 'repeat count' running out.
- READ is the type of transfer. All read or write functions are permitted.
- TTYPE is a single Camac device which gives Q=0 on the last valid transfer. (e.g. when Carriage Return is received from the teletype).
- BUFF is a reference to a buffer in computer memory. The implementation depends on the programming system, but in IML successive accesses are directed to consecutive locations.
- REP72 is a reference to a repeat count, which specifies the maximum number of transfers permitted. It also references a tally variable, which indicates the actual number of transfers achieved, and a status variable, which indicates the reason for terminating the transfer.
- NEXT is the name of the demand which initiates each transfer.
- CH2 is a name known to the local programming system. It specifies whether the transfer is to be performed by software or hardware, and if hardware the channel to be used.

The individual operations that make up the transfer could be programmed by the user. However they are a consistent set that should be optimised for the particular installation by the system programmer. He will also know the alternative instructions required to set up a hardware data channel if that is called for.

IML provides a similar statement (BSQ) for those users who require the fastest possible response to each demand. The initiating L-signal for each transfer is conveyed to the controller by a special line instead of by a Graded-L operation. This reduces the two Camac hardware cycles per transfer required by BLQ operations to one Camac cycle per transfer.

These two statements provide efficient, simple and compatible means of performing asynchronous block transfers (read or write). The operations may be interleaved with each other, with transfers of the same type to other modules, or with single-action operations.

A synchronous module is one that produces or accepts data at the maximum speed of the system. L-signal initiation of each transfer is therefore not required. The IML statement (BQ) for synchronous block transfer is as for BLQ but without the demand name parameter.

2.2.3 Multiple-action statements

Multiple-actions are essentially a series of single-actions performed on a sequence of Camac addresses using the same Camac function code. In read and write operations data is transferred to a sequence of locations in memory on a one-to-one basis. As with block-transfer statements a repeat count, tally variable, status variable and channel identifier are required. The basic form (MA) allows all Camac function codes to be used with given and calculated Camac arrays.

A special statement (MNQ) has been introduced into IML. This is a multiple test which terminates when Q=1 is received. It is used to find the first module in a sequence that has its L-signal or status set.

The address-scan mode of EUR4100e is supported by a Multiple Address Scan (MAD) statement. This may only be used with Q-scan arrays.

2.2.4 Module status statements

In order to avoid confusion with the two methods of accessing L-signal a set of statements for accessing L-signals and status have been included in IML.

```
ENL  NEXT
```

This statement enables the L-signal

declared as NEXT. The actual Camac command (F code etc.) used depends on the declaration. Similar statements may be used to disable, clear, test lam present and test lam status.

2.2.5 System statements

System statements operate on crate controllers and on system controllers. For example they initialise, clear, control the inhibit signal, and control demands at the crate and system level. Simple mnemonics are used for the features defined for crate controllers type A in EUR4600e.

3. Discussion

IML provides the basic facilities required by the user for programming Camac. The macro-expansion syntax and the subroutine-call syntax allow these facilities to be built up in a consistent manner as the need arises.

Single-action statements are the first requirement, these may be augmented by status and system statements if required as an aid to programming. The asynchronous block transfer statement BLQ greatly reduces the load on the user programmer and makes the use of data channels simple and efficient. The remaining block transfer and multiple action statements are of less general interest but may prove indispensible in particular applications.

4. Conclusions

The non-expert Camac programmer requires simple, easy to understand statements that control Camac activity. IML gives a carefully selected range of facilities described explicitly by its semantics. These have been implemented and found useful. The semantics should therefore be examined, not only by potential users of IML, but also by designers of other forms of Camac software as a stage on the route to software standardisation.

5. Acknowledgements

The interpretation of IML given here is the personal opinion of the author. Thanks are due to the untiring efforts of the working groups in selecting, defining and implementing the facilities of IML.

DISCUSSION

Q - F. Schutte

1. In October 1972 during the ESONE General Assembly you presented a preliminary version of a CAMAC language definition (IML; report SWG 16/72). We have implemented this language in our PDP-9-CAMAC system and are curious to know how many others have performed an implementation of the IML and have they raised specific problems on this subject.
2. From your lecture I noticed changes in the definition of IML. Is it possible to review the major alterations?

A - I. Hooton

1) The current version of IML has been implemented by members of the Software Working Group of the ESONE Committee during its development and some of its features have been changed as a result of the problems encountered. IML has been implemented in macro-expansion form on the PDP-10, PDP-11 and PDP-15. It may well have been implemented on other machines. As is general in CAMAC, as soon as any information is released many people make use of it without necessarily informing the working groups.
2) Essentially IML now consists of 3 single action, 3 block transfer and 3 multiple action statement types. These are so defined that a suitable subset may be implemented in the knowledge that future extensions will be compatible.

Q - G. van Heusden, F. Schutte

Is there a new report available about IML, replacing the report SWG 16/72?

A - I. Hooton

A report on IML is currently in draft form with the Software Working Group. This includes the semantic definitions and a macro expansion syntax. The USAEC NIM-CAMAC Software Working Group are anxious that this should appear as a joint publication and hope that it will also include a FORTRAN subroutine-call syntax. We are concerned that this may delay publication of an official document and may add the additional syntax at a later time.

Q - A. Voss

We have been using logical references to our modules. The program does not know the C, N values of the modules but these are determined by a table look-up at execution time. Subsequent physical relocation of the hardware can be accomodated in all programs by a utility program which patches the vector. It was not apparent to me that this type of approach would be supported by IML. Would the speakers like to comment on this?

A - I. Hooton

In IML the hardware declaration of the B, C, N, A values for a module may be indirect, that is via a pointer. This means that the values are held in data space rather than in the program and hence may be referenced and modified.
The given array declaration essentially defines a table of CAMAC addresses and individual modules are referenced by subscripts of the array. This would appear to be precisely the mechanism you require if many modules are involved.

C - H. Halling

I think it is very dangerous to change a pointer in a system; the whole goes into the infinite sink in the sky.

C - E. Rimmer

For our implemention in BASIC on the HP machine the following can be stated:
Since the crate address, station and subaddress are variables in the called statement, you may wish to write a program in a general way if you know that somebody is going to moving a module in the middle of an experiment. Then the first statement of your BASIC program can ask the user to input the present variables, present station numbers and crate addresses which can then be referred to as C, N and 0, 1, 2 etc. This is the way we allow for modules being moved in our applications.

Q - F. Colling

What is the highest speed for the synchronous module (BQ)?

A - I. Hooton

In the paper a reference to a synchronous module means one which can provide or accept data at the maximum rate of the computer system, hence the module may be assumed to be ready for each transaction. This is distinguished from an "asynchronous module" where the state of readiness is indicated by an L signal from the module which is used to trigger each individual transaction.

PROCEDURE CALLS - A PRAGMATIC APPROACH

J. Michelson, H. Halling
Kernforschungsanlage Jülich GmbH, Zentrallabor für Elektronik,
Jülich, Germany

ABSTRACT

The advantages and limitations of procedure calls as the basis for CAMAC software within high-level languages are discussed. The advantages of user familiarity and ease of implementation are stressed, but the limitations due to linkage-overhead and overall coding inefficiency are acknowledged. The US NIM CAMAC FORTRAN subroutines are discussed, and the alternative format provided by the ESONE Software Working Group - IML - is presented.

ZUSAMMENFASSUNG

Die Vorteile und Grenzen der Verwendung von Prozeduraufrufen als Grundlage für CAMAC-Software in einer höheren Programmiersprache werden diskutiert. Als Vorteil werden die Vertrautheit der Benutzer und die einfache Realisierung betont, aber auch die Grenzen infolge des Verwaltungsaufwands für den Unterprogrammanschluss und des schlechten Gesamtwirkungsgrades des erzeugten Codes werden aufgezeigt. Die US-NIM-FORTRAN-Subroutinen werden diskutiert sowie das Alternativformat vorgestellt, das aus der IML-Entwicklung der ESONE Software Working Group stammt.

RESUME

Discussion des avantages et des inconvénients des appels de procédure considérés comme base du software CAMAC dans les langages de haut niveau. L'accent est mis sur les avantages de cette méthode, habitude de l'utilisateur et facilité de mise en oeuvre bien que les limitations dûes à la surcharge de l'assembleur et à l'inefficacité d'un codage général soient reconnues. Discussion des sous-programmes CAMAC-FORTRAN proposés par le Comité NIM et présentation du projet IML étudié par le Groupe de travail software ESONE.

It is widely accepted in the computer industry that computer-programs written in machine language for particular applications are generally faster and shorter than an equivalent program written in a high level language and for a more general class of applications. However, studies have shown that programs can be written to solve a problem much faster in a high level language. This is especially true for the generation of large and complex programs. Since the costs of hardware is constantly decreasing it becomes more and more costeffective to utilize high-level languages to solve problems in spite of their lower machine efficiency.

Having accepted the use of a high level language for the solution of programming problems involving data acquisition and control using CAMAC we must now examine the methods by which such control can be integrated into high level language programs. There exist two general classes of implementation within the frame work of high level languages. The first is procedure calls the topic of this discussion, the second is the generation of inline code at the implementation level and the modification of the syntax at the language level.

There are some advantages to the use of procedure calls for implementing CAMAC software in a high level language environment. First: They provide a structure or framework on which to construct the software conventions. Second: The use of procedure calls within the high level language can be implemented using compilers which already exist. The software effort involved in generating a new compiler for a modified language syntax is quite large. It takes time and manpower which must be invested for each different type of computer and language so modified. Third: They are core efficient. The use of a single copy of a procedure for many places in a program results in low core requirement. Fourth: Because of their well defined structure they make it easier to produce machine independent programs. The communication paths between the user program and the procedures are well defined. As far as CAMAC is concerned procedure calls allow a relatively simple resolution of conflicts for usage of the CAMAC multiplexing hardware because they form effectively a single port from the user software to the software of the operating system. Fifth: The advantage of user familiarity cannot be overemphasized. High level languages allow the inexperienced user to describe his problem effectively and procedure calls beeing an integral part of the existing language structure are easier for untrained people to comprehend.

Procedure calls can be utilized in a variety of ways to implement CAMAC software. Perhaps the best known set of procedures presently available is the FORTRAN subroutine specification published by the NIM Committee software working group[3]. The basic philosophy of these subroutines is to allow the easy expansion of FORTRAN without modifying the syntax to describe the necessary hardware sequences and perform them. The passage of parameters is done using standard FORTRAN language elements. This could be viewed philosophicly as a single instruction processor or an I/O device driver where a given request is processed after the parameters are delivered. A good example of the NIM subroutine calls is the subroutine CMCRPT, which executes a CAMAC repeat mode transfer.

```
SUBROUTINE CMCRPT (F, B, C, N, AD, LN, DATA,
                   ERRORA)
COMMON/CMCCOM/NCOMP,NOMOD,NOCRT
INTEGER F,B,C,A D, DATA (2) ERRORA (4), Q
NWDS = NCOMP
IF (LN.GT.0) NWDS = 1
K= IABS (LN)
IF (LN.EQ.0) GO TO 40
L = ISIGN (1, LN)
J= 1
I= 0
1o  IF (I.GE.K) RETURN
    I=I+1
2o  CALL CMCBSC (F,B,C,N,AD,L,DATA(J),Q,ERRORA)
    IF (ERRORA (1).NE.0) GO TO 30
    IF (Q.EQ.0) GO TO 20
    J=J+NWDS
    GO TO 10
3o  ERROR A (2) = I
    RETURN
4o  ERROR A(1) = 9
    ERROR A(2) = 0
    EEOR A (3) = ERROR A (3) + 1
    RETURN
    END
```

These parameters for CAMAC repeat are the function code, the branch number, the crate number, the station number and the subaddress. A parameter for wordsize indication and number of CAMAC cycles, a parameter for the data buffer and a parameter for an error status vector. It is not intended to describe and explain the example in detail;if one is interested he should read the FORTRAN subroutine paper.

One of the difficulties of this structure is the very large number of parameters found in the calls. This is prone to programming errors and is difficult to read.

An expansion of the NIM subroutine structure is possible through the use of a two level concept where a set of procedures is used to combine and check parameters and another set of procedures is used for execution. The set of execution procedures will only allow parameters which have been combined by the declaration procedures. This can minimize the runtime overhead of processing the parameters. It also makes the resulting program less prone to programmer errors and much more readable.

The ESONE Software Working Group IML provides a good example of the use of a 2 level concept (but the IML of course can also be implemented using inline machinecode)[1) 4)].
The equivalent of the US NIM subroutine call for CAMAC repeat is a BR blockrepeat instruction implemented using a CALL BR blockrepeat.

CALL BR (F, EXT, INT, REP1)

The parameters for the call are the F code, an external reference parameter, an internal reference parameter and a repeat field parameter. These parameters have to be declared by declaration calls before use in this call.

```
CALL DCLE (EXT, H, B, C, N, A)
CALL DCLI (INT, B, STORE, 16, FINISH)
CALL DCLR (REP1, STAT, K, 1oo, TALLY)
```

All the parameters in the calls are defined in terms of the datatypes of the high-level language. The declaration of the external reference includes the parameters for the CAMAC hardware address, the internal declaration defines a buffer named STORE, storing 16 bit words and passing control to the instruction at label FINISH when the buffer is full. The parameters for the repeat declaration consist of a status variable holding status information like Q or error after the return, K specifies the constant 1oo which is the number of words to be transferred and TALLY the parameter to receive the actual number of transfers which did take place if the operation is not completed successfully.
All these declaration statements are executed in the initialization run which should be clearly distinguished from the "Real time run". The initialization run is used to evaluate, check and pack the hardware address etc. so that those time consuming operations are done before the efficient realtime run.

A comparison of the NIM subroutines to the format produced by implementing IML with procedure calls shows the advantages of the two level structure. It also shows that the well defined nature of the NIM approach is suited to explaining CAMAC actions to software people familiar with FORTRAN.

The contrasting approach of language syntax modification in inline machine code generation allows the use of instructions with execution address parameters for conditional branching which can't be done in a pure procedure call implementation. This allows sequences of hardware actions to be performed independent of the normal algorithmic part of the host language.

Another approach being pursued at the University of Oxford is a hybrid implementation using procedure calls to access blocks of inline code which were generated in assembly language by a MACRO expansion technique. This approach is possible due to the existence of language independent linkage structures - such as procedure call BEGIN and RETURN - in IML[2)].

A final approach and probably the widest used by a silent majority is the application-software-approach. In these implementations one is not concerned directly with CAMAC commands but rather provides subroutines that are application oriented. An example would be a set of subroutines for controlling steppermotors that would be used in the implementation of a triple axis neutron spectrometer. The user here is not concerned with any details of the hardware. He is not interested in programming

CAMAC with C, N, A , F commands - he is only interested with problem oriented language elements to describe his problem. A possible solution for this particular example would be CALL DRIVE (M1,-125,6) where the parameters define the motor number, and the direction and angle of the rotation.

The very great advantage of this type of CAMAC procedure call approach is that the implementation can be done by highly skilled system programmers and extremely good machine utilization can be achieved while at the same time the user may define his problem and its solution in a problem oriented language that contains elements familiar to him.

Now finally lets come to the disadvantages of procedure calls. Because of their nature they are usually slower than the equivalent logic programmed using inline code. This is due first to the required linkage and second to the passage of parameters and runtime manipulations of these parameters prior to hardware execution. In the case of the two level concept the runtime manipulations don't matter and the parameteraccess only adds one level of indirection compared to inline code which may be unavoidable for a certain class of machines.

For certain classes of procedures such as to implement various blocktransfers like repeat mode or stop mode, the overhead in calling a procedure and manipulating the parameters is not very significant. However, for single CAMAC commands causing only one CAMAC cycle the overhead becomes quite significant.

other limitation is that in a real time system the procedures themselves must be coded in a relatively sophisticated manner in order to be reentrant. If they are not made reentrant much of the advantage or core-efficiency is lost.

Some of the following papers show how to avoid all these difficulties and how to use a high level language to write efficient code. But there is a large number of users not familiar with these solutions which are sometimes too specific to justify the efforts in training the programmers to learn other languages besides the giant FORTRAN standing on his feet of clay as far as realtime applications are concerned.

(1) HALLING, H., MICHELSON, J. "Will FORTRAN Tolerate this IML?" SWG 2o/72

(2) HAGAN, P.J. "A View of IML" SWG 25/73

(3) THOMAS,R.F. "CAMAC FORTRAN Subroutines" SWG 28/73

(4) HAGAN, P.J., DAVIES, M.P. "Another view of IML" SWG 33/73

(SWG papers are internal working papers of the Software Working Group - If you would like copies contact the secretary)

DISCUSSION

Q - J. Visschers

Concerning the treatment of LAM's, I feel there is a difference between
- LAM's generated each time a word is available for reading e.g. in a FORTRAN-level initiated block transfer and
- LAM's occurring not on the initiative of the FORTRAN-program, but on the initiative of the outside world, e.g. the occurrence of an error condition in a process-control system.
It is clear to me that the first case can be handled inside a FORTRAN block-transfer procedure.
The second case requires treatment, preferably on FORTRAN level, but without making the whole system deaf to normal LAM's during the execution of the high-level error-treatment routine.
Could you comment on the implementation of this kind of LAM-treatment?

A - H. Halling

You are right. The IML language clearly distinguishes between word synchronising LAM's and LAM's triggering the real time operating system. In the second case one depends on the features of the FORTRAN environment. If it is able to perform real time actions one can use these features. If not don't use it for real time applications.

Q - J. Nehmer

H. Halling and E. Rimmer have described the PROCEDURE CALL technique as a convenient tool to add standardized CAMAC-I/O - capabilities to existing real time systems.
My questions are:
1) What are the CAMAC-configuration limitations (hardware) introduced by fixed CAMAC-I/O - instruction formats?
2) The problem of actually handling the CAMAC I/O requests is left to some existing (unknown) operating system by the methods proposed. Did you investigate possible implications arising from the delegation of unspected service to existing real time operating systems?

A - H. Halling

1) There is no limitation of CAMAC configurations by using parameter-oriented CAMAC-calls. Each address entity (branch, crate, module, sub-module) is represented by an integer or variable.
2) The LAM handling heavily depends on the operating systems that are used. If the FORTRAN (or BASIC) environment does not allow interrupts I would not use it in a real-time application.

CAMAC AND INTERACTIVE PROGRAMMING

E. M. Rimmer
Nuclear Physics Division, CERN, Genève, Switzerland

ABSTRACT

Interactive programming in a high-level language, such as BASIC, allows simple program generation, modification, and execution, typically through keyboard/printer dialogue. Addition of CAMAC operations to such a system provides excellent facilities for setting-up, debugging and operating CAMAC hardware, and for any non-time-critical CAMAC application.

ZUSAMMENFASSUNG

Die interaktive Programmierung in einer höheren Programmiersprache wie z. B. BASIC gestattet es, auf sehr einfache Weise Programme zu erzeugen, zu modifizieren und ablaufen zu lassen, im allgemeinen im Dialogverkehr über Tastatur und Drucker. Durch Zusatz von CAMAC-Operationen zu einem solchen System erhält man hervorragende Möglichkeiten für Inbetriebnahme, Fehlersuche und Bedienung von CAMAC-Anlagen sowie für nicht-zeitkritische Anwendungen von CAMAC.

RESUME

La programmation conversationnelle dans un langage de haut niveau tel que BASIC, permet normalement la mise au point, la modification ainsi que l'exécution de programmes simples, au moyen d'un dialogue clavier/imprimante. Par addition d'opérateurs CAMAC, on obtient des systèmes qui conviennent parfaitement à la mise en place, à la mise au point et à l'exploitation des éléments CAMAC ainsi qu'à toutes les applications CAMAC où le temps n'est pas un facteur décisif.

1. INTRODUCTION

CAMAC systems may be regarded as high-level interfacing hardware between a set of devices and a controller such as a minicomputer. The software interfacing between the computer and CAMAC should therefore also be at a high level to retain simplicity, especially for users who are more interested in results than in programming.

2. INTERACTIVE PROGRAMMING AND INTERPRETERS

Many applications require user intervention and user-controller dialogue. In the build-up, test, and debugging phases of a hardware/software system, efficient interaction is most desirable, and is often useful at a later stage in non-time-critical processes. For these purposes an interpretive language is ideal. BASIC is becoming a standard such language for minicomputers; FOCAL is implemented on DEC computers.

These languages are FORTRAN-like and easy to learn. They allow algebraic manipulations and some input/output, usually keyboard, printer, and paper tape ASCII character handling.

The interpreter plus handlers reside in core. The user program is stored in source code from keyboard or paper tape input. Numbered source statements are interpreted and executed sequentially at run time, execution being started and stopped by keyboard commands. Programs can be modified and re-run quickly and easily from the keyboard.

3. EXTENSION OF BASIC INTERPRETERS

We will confine our attention to the BASIC language. The need is to extend the I/O facilities to include any device, CAMAC in particular.

One method is to extend or alter the syntax of the language by modifying the interpreter itself. This can be a complicated and inflexible procedure, although it may produce efficient interpreted coding.

A much more simple method is to implement additional facilities using the CALL (or similar) statement within the language. This involves wiriting independent subroutines, in assembler code, together with a link table of entry points which is scanned by the interpreter on encountering a CALL statement. This method is gaining wide approval for its advantages of simplicity and modularity[1]. Libraries of such subroutines are already being established by some companies.

4. CAMAC IN BASIC

We will describe the addition of CAMAC operations to a Hewlett-Packard 2100 series BASIC interpreter by a group at CERN, Geneva[2]. Work on other systems can be found in the literature[3-6].

The general form of a HP BASIC CALL statement is: statement number CALL(subroutine number, parameter list) e.g. 10 CALL(7,A,B,100,X*Y). The statement number determines the execution sequence of statements within the BASIC program. The subroutine number identifies the assembly language subroutine to be entered and corresponds to a FORTRAN name. The parameter list is of optional length and parameters can be any formula. An address list of these parameters is passed to the called subroutine for transfer of values, if any, between itself and the BASIC program.

For a single-branch, multi-crate CAMAC system, we found the necessary and sufficient parameters were:

C = crate address

N = station address

A = station sub-address

F = function code

D = data word(s) transferred (if any)

Q = true or false (1 or 0) Q response returned from module

W = word count if data block transfer.

The CALL statement then has the general form CALL(1,C,N,A,F,D,Q), or CALL(10,C,N,A,F,D(1), Q,W) in the case of data block transfer where D(1) is the first element of the data array and W the word count for transfer to or from this array.

The data format, e.g. binary, BCD, etc., was specified by calling different subroutines each with the same parameter set, although an additional parameter and a single subroutine could have been used equally well.

These subroutines have been written for three HP-CAMAC interfaces: BORER 2201 Branch Driver, CERN Type 7218 Branch Driver, and a HPCC-066 single crate interface. Within hardware compatibility limits, programs in BASIC are compatible with all three systems.

Detailed FAULT diagnostics are provided similar to BASIC error messages, namely a message is printed during execution of the statement wherein the fault was detected, such as crate failure, missing flags, and so on.

Non-keyboard interrupts (look-at-me, LAM) are not serviced by the interpreter which we used, but in CAMAC they can be monitored by the "test LAM" function.

5. GENERAL COMMENTS

Interpreters are large and slow; ∿ 4K of core and ∿ 5 milliseconds per statement for HP 2100 BASIC. This CAMAC facility, together with several utility subroutines, was intended primarily for testing and debugging hardware, and as such has proved invaluable to maintenance and experimental teams alike. In applications with low data rates, with low operational rates, or in cases where the useful lifetime of the final program does not merit laborious production of "elegant" coding, extended interpreters provide extremely simple and effective software systems for CAMAC users.

6. ILLUSTRATIVE EXAMPLE

The following example illustrates the use of BASIC CAMAC calls to punch binary data blocks onto a CAMAC-controlled paper tape punch. The handshake is via a LAM; that is, the CAMAC module returns a look-at-me whenever one data word has been punched.

```
10  REM:  INITIALIZE CAMAC SYSTEM
20  CALL(4)
25  LET C = N = 1
30  LET A = 0
35  DIM D(100)
40  REM:  TURN ON PUNCH
45  CALL(1,C,N,A,26,X,Q)
50  REM:  PUNCH BLOCK
55  FOR I = 1 TO 100
60  CALL(1,C,N,A,16,D(I),Q)
65  REM:  WAIT FOR LAM, AND CLEAR IT
70  CALL(1,C,N,A,8,X,Q)
75  IF Q = 0 THEN 70
80  CALL(1,C,N,A,10,X,Q)
85  NEXT I
90  REM:  TURN OFF PUNCH
95  CALL(1,C,N,A,24,X,Q)
100 END
```

The contents of the punched data array D would obviously be established prior to entering this CAMAC coding.

REFERENCES

1) Report on NIM-CAMAC Software Working Group Meeting, July 1973, TRIUMF, Vancouver, B.C., Canada.

2) RIMMER, E.M., BASIC callable CAMAC routines, NP Group note NP-DHG, CERN, Geneva (1973).

3) MAY, F., HALLING, H., PETRECZEK, K., FOCAL overlay for CAMAC data and command handling, CAMAC Bulletin No. 1 (1971), 18.

4) ZWOLL, K., POFAHL, E., HALLING, H., PDP-8 operating system for non-time-critical CAMAC experiments, CAMAC Bulletin No. 5 (1972), 28.

5) HALLING, H., ZWOLL, K., JOHN, W., CAMAC overlay system for single-user BASIC and modification of 8-user BASIC for a PDP-11, CAMAC Bulletin No. 6 (1973), 15.

6) BALS, I., de AGOSTINO, E., An extended BASIC language for CAMAC programming, CAMAC Bulletin No. 7 (1973), 25.

IMPLEMENTING CAMAC BY COMPILERS

W. Kneis
Kernforschungszentrum Karlsruhe, Zyklotron-Laboratorium,
Karlsruhe, Germany

ABSTRACT

Demands to real-time systems as minimal execution time minimal core requirements etc. recommend the use of compilers in CAMAC-programming.

The possibility to implement a CAMAC-language by a compiler firstly is a function of the level and concept of the language.

There are languages (meta-languages) for the description of the syntax of a programming language. By using these meta-languages you are able to formulate a compiler for a special language.

ZUSAMMENFASSUNG

Anforderungen an Real-time-Programme, wie schnelle Laufzeit, kleiner Kernspeicherbedarf etc. empfehlen den Einsatz von Compilern bei der CAMAC-Programmierung.

Die Implementierungsmöglichkeit einer CAMAC Sprache durch einen Compiler wird jedoch in erster Linie vom Sprachlevel und Sprachkonzept abhängen.

Es gibt Sprachen (Meta-Sprachen) zur Beschreibung der Syntax einer (Programmier-) Sprache. Mit Hilfe dieser Metasprachen können Compiler für spezielle Sprachen formuliert werden.

RESUME

Si l'on exige que des programmes en temps réels apportent des temps de traitement courts et un besoin limité de mémoire centrale etc., il faut employer des compilateurs pour la programmation CAMAC

Mais la possibilité d'implanter un langage CAMAC par un compilateur dépendra en premier lieu du niveau et de la conception du langage.

Il y a des langages (métalangages) pour décrire la syntaxe d'un langage (de programmation). A l'aide de ces métalangages des compilateurs peuvent être formulés pour des langages spéciaux.

1. Real-time-Programmierung und CAMAC

Der Einsatz von Computern in der Real-time-Anwendung ergibt bis heute folgendes Bild:

- Einsatz einer Vielzahl verwendeter Rechnertypen
- Einsatz einer Vielzahl verschiedener Peripherie
- Eine Vielzahl spezieller Problemloesungen

Demgegenueber steht der in den letzten Jahren einsetzende Trend einer Standardisierung der real-time Prozessperipherie, wie sie insbesondere durch das CAMAC-System angestrebt wird.
Spezielle Problemloesungen sind gewiss von einem schnellen Zeitverhalten in der Ausfuehrung, einer oekonomischen Kernspeicherausnutzung, und einer schnellen Realisierungsphase gekennzeichnet. Sie sind jedoch in erheblich groesserem Masse mit dem Nachteil der speziellen Assemblersprachen verschiedener Rechnertypen, der damit verbundenen Nichttransportabilitaet, der schwierigeren Verstaendlichkeit der Programme und der mangelnden Selbstdokumentationsmoeglichkeit der Assemblersprachen behaftet.
Die weitere pragmatische Entwicklung dieser Loesungen resultiert in der Erstellung von Unterprogrammen fuer spezifische Operationen. Ein Aufruf dieser Unterprogramme aus einer hoeheren Sprache (z.B. FORTRAN) besitzt den wesentlichen Vorteil, dass normale Rechenoperationen und sonstige Standardfunktionen in dieser Sprache formuliert werden koennen, was Schreibaufwand und Fehleranfaelligkeit stark vermindert (1). Der Widerspruch dieser Loesung ist dass sie eine Standardisierung von Peripherieoperationen anstrebt, diese jedoch durch eine rechnerspezifische Problemloesung implementiert. Sie ist damit immer noch rechner- und peripherieabhaengig.
Eine Peripheriestandardisierung wie sie das CAMAC-System darstellt, ermoeglicht durch normierte Hardware-Schnittstellen eine allgemeine Definition von Peripheriefunktionen. Diese Funktionen sind nicht mehr rechnerabhängig. Die Eingliederung von CAMAC in einen Rechner resultiert in dem Problem, den in der Literatur vielzitierten "Virtuellen Controller" (VC) zu definieren (2). Der VC ist eine rechnerbedingte Kombination von Software und Hardware, die nach aussen fest definierte Schnittstellen besitzt (3). Die Schnittstelle zur CAMAC-Peripherie ist durch den CAMAC-Systemcontroller gegeben. Die Schnittstelle zu CAMAC-Peripheriefunktionen wird durch eine zur Zeit in der Entwicklung befindliche "Intermediate Language" (CAMAC IML) definiert. Innerhalb des VC koennen die zu bewaeltigenden Aufgaben zum groessten Teil sowohl hardware- als auch softwaremaessig geloest werden. Der Softwareteil des VC stellt gerade die Menge von Programmen dar, die spezifisch fuer jeden Rechnertyp bereitgestellt werden muss.
Nach Schweizer (1) gestattet IML in Verbindung mit dem VC die maschinenunabhaengige Darstellung von Peripheriefunktionen. Die sich abzeichnenden Eigenschaften von IML gestatten jedoch darueber hinausgehend die Moeglichkeit, den VC selbst in IML zu schreiben.
Forderungen, die bei einer CAMAC Implementierung unbedingt im Vordergrund stehen sollten, sind:

- schnelle und sichere Ausfuehrung von Peripheriefunktionen
- oekonomische Kernspeicherausnutzung
- leichte Programmierung
- Selbstdokumentation
- Transportabilitaet

Hierbei sind die drei letzten Punkte durch die CAMAC Sprache selbst gegeben, waehrend die beiden ersten Punkte sehr stark von der Implementierungsmethode abhaengen.

2. Uebersetzer

Voraussetzung zur Loesung eines Problems durch einen Computer ist das Vorhandensein bzw. die Erstellung eines Programmes in der Maschinensprache (Object-Sprache) des jeweiligen Rechners. Die Entwicklung hat gezeigt, daß der Trend in der Datenverarbeitung sehr schnell dazu ueberging, Programme zur Bearbeitung eines Problems in einer dem Problem angepassten, hoeheren Sprache (Source-Sprache) zu definieren und die Erstellung des eigentlichen Maschinenprogramms vom Computer selbst erzeugen zu lassen. Programme, die ein in einer hoeheren Sprache abgefasstes Programm uebersetzen, nennt man Uebersetzer.
Ausgehend von der Definition eines Programmes als einer endlichen Folge von Instruktionen wird ein Uebersetzer dadurch charakterisiert, dass er eine in der hoeheren Sprache abgefasste Instruktion in eine endliche Folge von Maschineninstruktionen uebersetzt. Unter Zuhilfenahme von verschiedenen Kriterien, wie z.B. Sprachumfang, Niveauunterschied zwischen Source- und Objectsprache und Realisierungsmethode kommt man zu einer groben Einteilung von Uebersetztern in Assembler, Compiler und Interpreter.
Der Assembler kann als ein spezieller Uebersetzer aufgefasst werden, der eine Eins-zu-eins-Beziehung zwischen Source- und Objectsprache voraussetzt. Compiler und Interpreter unterliegen dieser Einschraenkung nicht; sie unterscheiden sich untereinander im wesentlichen durch verschiedene Realisierungsmethoden. Ein Compiler uebersetzt Instruktionen der Source-Sprache in eine Object-Sprache, die von einem Computer direkt ausgefuehrt werden kann. Ist die Object-Sprache die Assembler-Sprache des Computers, so spricht man von einem Kaskaden-Compiler, da vor der Ausfuehrung noch ein Assemblerlauf notwendig ist. Uebersetzt der Compiler jedoch direkt in die Maschinensprache, so wird er als Direkt-Compiler bezeichnet. Ein Interpreter hingegen wandelt die Instruktionen der Source-Sprache in eine abstrakte interne Darstellungsform um,

die nur ueber ein eigens dafuer erstelltes Programm, dem sogenannten Simulator ausgefuehrt werden kann.
Der eigentliche Uebersetzungsvorgang laesst sich im wesentlichen in zwei Hauptaufgaben unterteilen:

- Satzaufbauanalyse (Syntaxanalyse)
- Bedeutungsanalyse (semantische Anal.)

Die Funktion eines Uebersetzers ist damit einerseits die Definition und Anwendung der Syntaxregeln der Source-Sprache und andererseits die Codegenerierung bzw. die Interpretation von Instruktionen der Source-Sprache durch Auswahl einer geeigneten Folge von Object-Instruktionen (4). Die Komplexitaet dieser Aufgabe wird in erster Linie bestimmt von:

- Eigenschaften der Syntax der Source-Sprache
- Eigenschaften der durch die Object-Sprache darstellbaren Semantik

Der bei einer Uebersetzung zu bewaeltigende Aufwand ist daher eine unmittelbare Funktion des Levelunterschiedes zwischen Source- und Object-Sprache. Daher sollte bei der Definition einer Sprache auf die enge Verknuepfung zwischen syntaktischer und semantischer Definition geachtet werden, da strukturierte Instruktionen lediglich eine Hilfe bei der Interpretation eines Satzes darstellen (4). Ein Compiler ist genau dann wohl definiert, wenn er sämtliche Syntaxregeln der Source-Sprache enthaelt und fuer jede dieser Syntaxregeln eine entsprechende Codegenerierungsregel besitzt. Eine weitere Forderung bei der Sprachuebersetzung sollte sein, dass die Syntaxregeln lokal anwendbar sind. Man spricht in diesem Fall von einer kontextfreien Grammatik (5). Contextfrei bedeutet anschaulich, dass bei der Syntaxanalyse einzelne Satzelemente (Bestandteile einer Programminstruktion) ohne Hinzuziehen des Contextes als syntaktisch richtig erkannt werden koennen. Was die Codegenerierung anbetrifft, so muss der Compilerdesigner die Semantik, die jeder Codegenerierungsregel zukommt, dadurch festlegen, dass er die explizite Folge der Instruktionen der Object-Sprache bestimmt (4).

3. CAMAC Sprachen

Eine vollstaendige Softwareintegration der CAMAC-Hardware als universale I/O-Struktur in einem Rechner fuehrt zwangsläufig auf eine eigene Sprache. Dabei wurde von der Ueberlegung ausgegangen, eine"high level" CAMAC-Sprache (CAMAC HLL) zu definieren, die weitgehend unabhaengig von dem verwendeten Rechner ist. Dies wurde dadurch erreicht, dass man als Object-Sprache der CAMAC HLL eine rechnerunabhaengige "intermediate level" CAMAC-Sprache (CAMAC IML) waehlte. CAMAC IML wird nun so definiert, dass sie den niedrigsten implementierungsunabhaengigen Level besitzt. Damit ist die CAMAC HLL rechnerunabhaengig. Vom Standpunkt der CAMAC HLL stellt CAMAC IML eine Art Assemblersprache dar, die allen Eigenschaften der CAMAC-Hardware nach (6) und (7) Rechnung traegt. Das gilt insbesondere deshalb, da man CAMAC IML als die Assemblersprache des VC auffassen kann. Vergleicht man CAMAC IML jedoch mit Assemblersprachen reeller Rechner, so besitzt sie gerade wegen ihrer Implementierungsunabhaengigkeit und Problemorientiertheit gewisse Eigenschaften hoeherer Programmiersprachen (3). Dies ist aber gerade charakteristisch fuer "intermediate level" Sprachen.
Ausgehend von dem CAMAC Hardwarekonzept das im wesentlichen aus I/O-Funktionen besteht, muss eine Sprachkonzeption fuer CAMAC zunaechst nur Sprachelemente enthalten, die fuer den I/O und saemtliche Kontrollfunktionen notwendig sind. Damit ergibt sich aufgrund der CAMAC Norm fuer die reine I/O-Kontrolle und -Ausfuehrung eine Sprache mit einer relativ geschlossenen Syntax und Semantik. Sprachen mit diesen Eigenschaften bezeichnet man oft als echt problemorientiert; sie bieten vom Standpunkt des Uebersetzungsaufwandes erhebliche Vorteile, auf die in Kapitel 5 genauer eingegangen wird.
Andererseits kann eine solche echt problemorientierte Sprache als Widerspruch zum Konzept der vollstaendigen Softwareintegration betrachtet werden, da gerade in der real-time Anwendung eine enge Verknuepfung zwischen I/O-Kontrolle und Datenmanipulation gewuenscht wird, um Timing und Scheduling durchfuehren zu koennen. Damit ergeben sich zwangsläufig zwei kontraere Sprachkonzeptionen: zum einen eine "universal" CAMAC Sprache, die z.B. alle Moeglichkeiten der Datenmanipulation etc. beinhaltet, und zum anderen das Konzept der reinen CAMAC-I/O Sprache, die sich nur auf die Kontrolle der CAMAC Hardware beschraenkt. Zum Konzept der "universal" CAMAC-Sprache ist zu bemerken, dass die Aufgaben in der real-time Anwendung ebenso breit gefaechert sind wie in der batch Anwendung. Hier sind jedoch bis heute alle Versuche zu einer einheitlichen, normierten Sprache zu kommen, fehlgeschlagen, wie z.B. bei Fortran, wo nahezu jeder Rechnerhersteller USASI-Fortran mit "Nicht-Standard-Fortran-Erweiterungen" anbietet. Aehnliche Bestrebungen auf dem real-time Anwendungssektor, wie sie in juengster Zeit durch die LTPLC/E-Gruppe mit Sprachen wie PEARL, PROCOL, CORAL etc. unternommen werden, sind gegenwaertig noch nicht vollstaendig abgeschlossen. Allerdings ist im Gegensatz zu frueheren Versuchen bemerkenswert, dass beim Sprachdesign auf das Verschmelzen der besten Eigenschaften obiger Sprachen und auf die Formulierbarkeit geeigneter Subsets fuer den Anwender (8) Wert gelegt wird.
Damit bietet sich zur Zeit die Implementierung von CAMAC durch Einbettung der CAMAC Sprachen (HLL und IML) in eine geeignete, fremde Sprache (Host-Parasite-Konzept) an, die Ausgangspunkt fuer die weiteren Betrachtungen sein soll. Dieses Konzept bietet die

Moeglichkeit,eine eingebettete Sprache in einer "fremden" Umgebung zu definieren, ohne genaue Kenntnis der formalen Syntax und Semantik dieser Umgebung. Dies ist nur moeglich, wenn beim Uebersetzungslauf eines Host-Parasite-Programmes an den entsprechenden Stellen ein "Umschalten" auf die verschiedenen Uebersetzer von Host- und Parasite-Sprache durchgefuehrt wird. Wie dieses Problem der syntaktischen Verknuepfung von Host-Parasite-Programmen geloest werden kann, wird in Kapitel 4 naeher ausgefuehrt. Die semantische Verknuepfung der Parasite-Sprache mit der Host-Sprache erfolgt ueber Symbole und Deklarationen und ist damit Sache des Anwendungsprogrammierers.

4. Sprachlevel und Compilerdesign

Eine enge Verflechtung zwischen Sprachlevel und Compilerdesign resultiert nach Knuth (9) prinzipiell darin, dass der Compiler selbst die Sprache definieren sollte. Das ergibt einerseits die Abhängigkeit der Sprache von den Compilereigenschaften und andererseits die Implementierung von Spracheigenschaften durch einen Compiler. Aufgrund des heutigen Standes der Compilertechnik sollte damit aus Gruenden der Einfachheit in der Compilererstellung von einer durch eine contextfreie Grammatik zu beschreibenden Sprache ausgegangen werden. Weiterhin sollte eine solche contextfreie Sprache keine Mehrdeutigkeitsstellen enthalten (unambigity). Diesen Punkten wurde durch die bisherige Arbeit der CAMAC-Software-Working-Group (SWG) bei der Definition der CAMAC HLL und CAMAC IML Rechnung getragen, einerseits durch die eindeutige Fixierung der CAMAC HLL auf CAMAC IML und andererseits durch die syntaktische Struktur von CAMAC IML selbst.
Unter dieser Voraussetzung ist die Implementierung von CAMAC durch Compiler nur noch von folgenden Punkten abhaengig.

- Hardware und Software Environment mit
  - Betriebssystem
  - Compiler fuer hoehere Sprachen
  - Assemblereigenschaften
- Sprachkonzept fuer CAMAC als
  - full language
  - host parasite language

Hardware und Software Environment sind zum einen insofern ausschlaggebend für die Implementierung eines Compilers fuer eine Camac Sprache weil sie Moeglichkeiten und Voraussetzungen darstellen, die sowohl fuer die Erstellung des Compilers als auch fuer die eigentliche Uebersetzung der CAMAC Sprachen von Bedeutung sind. Insbesondere ist die Moeglichkeit der Verwendung hoeherer Sprachen, wie z.B. Fortran, bei der Erstellung eines Compilers wuenschenswert. Weiterhin sollte fuer die Uebersetzung die Software Compilierungserleichterungen bieten, wie z.B. Filehandling, Link- und Load-Facilities, was im wesentlichen bei der Verwendung von hoeheren Sprachen gegeben ist. In diesem Zusammenhang wird auf die Moeglichkeit verwiesen, die Compilerentwicklung fuer eine CAMAC Sprache an einer anderen, groesseren Rechenanlage durchzufuehren, die diese Software besitzen sollte (10). Dies wird umso attraktiver, je besser die Kommunikationsmoeglichkeit zwischen Gross- und Kleinrechner ist. Im Falle einer direkten Hardwarekopplung ergeben sich damit geradezu ideale Arbeitsbedingungen fuer den Compilerdesigner und Anwendungsprogrammierer, die sich waehrend der Compilierungs- und Checkout-Phase nicht an die Einschraenkungen der oft abgemagerten Kleinrechner binden müssen.
Zum anderen stellt gerade das Betriebssystem mit seinen Moeglichkeiten der Assemblersprache, der Verwendung von Macroaufrufen und Procedure-calls, des Taskings oder Multitaskings etc. eine Basis dar, auf die bei der Generierung eines Objectcodes aus einer CAMAC Sprache zurueckgegriffen werden kann. Gerade bei der Implementierung der CAMAC IML durch einen Kaskadencompiler ist der Sprachumfang des Assemblers ausschlaggebend fuer den Uebersetzungsaufwand, da vom Assembler sehr viele Routinearbeiten, wie Aufstellen von Symboltabellen, Konstantenverarbeitung, Speicherplatzreservierung und -einteilung etc. uebernommen werden koennen. So kann z.B. mit Hilfe von Pseudoassemblerbefehlen (Deklarationen) auf sehr einfache Art und Weise die Zusammenstellung eines 24-bit CNAF-Wortes (6) geloest werden, ohne dass der Compiler selbst eine Umrechnung der Dezimaldarstellung in die maschinenabhaengige Zahldarstellung vornimmt, da dies vom Assembler erledigt wird.
Das ergibt den Widerspruch, dass einerseits durch Verwendung von hoeheren Sprachen im Compilerdesign eine - wenn auch mit gewissen Einschraenkungen - Unabhaengigkeit von dem verwendeten Rechnertyp erreicht wird, andererseits aber gerade bei der Codegenerierung die speziellen Eigenschaften des Rechners benutzt werden. Dieser Widerspruch wird genau dann aufgeloest, wenn man eine strenge Trennung in der Implementierung des Compilers und der sich anschliessenden Implementierung der zu uebersetzenden Sprache durchfuehrt. Das bedeutet, dass die Spezifikationen der Syntax und Semantik der zu uebersetzenden Sprache dem Compiler als Inputdaten angeboten werden und erst danach eine Uebersetzung der Sprache stattfinden kann. Ein Beispiel fuer eine solche vollkommene Eingabesteuerung eines Compilers wird in Kapitel 5 anhand eines Metacompilers erlaeutert.
Aus den in Kapitel 3 genannten Gruenden werden die Aspekte eines full language Konzeptes hier nicht weiter ausgefuehrt. Fuer die Automatisierung der Einbettung einer Parasite-Sprache in eine Host-Sprache lassen sich hinsichtlich der Kontrolluebergabe und Uebersetzungsmethode unterschiedliche Implementierungsmöglichkeiten anführen.

Die Kontrolluebergabe bei der Uebersetzung geschieht entweder ueber Fehlerausgaenge gleichberechtigter Prozessoren oder durch einen uebergeordneten Prozessor, der markante Syntaxunterschiede zwischen Host- und Parasite-Sprache erkennt. Die erste Moeglichkeit fuehrt damit prinzipiell auf laengere Uebersetzungszeiten gegenüber der zweiten Moeglichkeit, die von vornherein den richtigen Uebersetzer fuer ein zu uebersetzendes Sprachelement wählt. Hinsichtlich der Uebersetzungsmethode muss davon ausgegangen werden, ob der Host-Uebersetzer als Direkt- oder Kaskadenuebersetzer arbeitet. Im Falle der Direktuebersetzung muessen Host- und Parasite-Uebersetzer eng miteinander gekoppelt sein, was z.B. bedeutet, dass sie dieselbe Objektsprache besitzen, dieselben Symboltabellen benutzen etc. Das hat jedoch fuer die Implementierung erhebliche Eingriffe in den Host-Uebersetzer zur Folge. Im Falle der Kaskadenuebersetzung der Host-Sprache oder einer Einbettung der Parasite-Sprache in eine Assemblersprache besteht eine Kopplung der beiden Uebersetzer nur ueber die gemeinsame Objektsprache. Es eruebrigt sich damit ein groesserer Eingriff in den Host-Uebersetzer. Keinerlei Eingriffe in den Host-Uebersetzer ergeben sich, wenn man von einer Kontrolluebergabe durch einen uebergeordneten Prozessor und einer Einbettung der Parasite-Sprache in eine Assemblersprache oder in eine von einem Kaskadencompiler zu uebersetzenden Host-Sprache ausgeht.

5. Metacompiler

Als Basis fuer die heutige Compilertechnik koennen die grundlegenden Arbeiten von Chomsky gelten, die von einer grammatikalischen Erfassung der Sprachen ausgehen. Die Grammatik einer Sprache sollte nach Chomsky (11) "alle und nur die" Saetze einer Sprache generieren. Ausgehend davon hat man Schreibweisen (Formen) entwickelt, um die Grammatik einer Sprache darstellen zu koennen. Diese Formen sind jedoch selbst wieder Sprachen und da sie Sprachen beschreiben, bezeichnet man sie als Metasprachen. Die in der Compilertechnik heute am weitesten verbreitete Metasprache ist die Backus-Naur-Form (BNF) (12).

Die Beschreibung einer Sprache in der BNF besteht aus einzelnen hierarchisch angeordneten Syntaxregeln, die, entsprechend angewendet, Saetze der Sprache erzeugen, d.h. auch vorhandene Saetze analysieren koennen. Ein Uebersetzer ist damit lediglich ein Programm, das die Syntaxregeln (Metabeschreibung) einer Sprache anwendet und die Implementierung einer Sprache auf einem Computer besteht ebenfalls nur darin, ihre Metabeschreibung in ein ausfuehrbares Programm, z.B. einen Compiler, zu ueberfuehren.

Zur automatischen Uebersetzung von Metabeschreibungen in ausfuehrbare Programme gibt es heute eine Vielzahl automatischer Uebersetzungssysteme, sogen. Metacompiler (13). Eine spezielle Klasse innerhalb der Metacompiler bilden die selbstbeschreibenden Metacompiler, die ihre eigene Metabeschreibung uebersetzen, d.h. ihren eigenen Compiler erzeugen koennen. Notwendig hierzu ist, dass die Metabeschreibung so abgefasst ist, dass sie sich selbst beschreiben kann. Die folgende, nur von den Grundsymbolen NONTERMINAL, TERMINAL, LETTER, DIGIT und daraus abgeleiteten Begriffen

```
 '          NONTERMINAL
.STR    =  'TERMINAL ...'
.ID     =  LETTER(LETTER/DIGIT)...
```

ausgehende Metabeschreibung besitzt Selbstbeschreibungseigenschaften.

Metabeschreibung:

```
PROG ::= STMT ... ;
STMT ::= .ID '::=' EXPR ... ';' ;
EXPR ::= ELMT ('/' ELMT ) ... ;
ELMT ::= .ID/.STR/'.ID'/'.STR'/'...'/
         '(EXPR')' / EXPR ;
```

Besitzt ein selbstbeschreibendes Metacompilersystem ueberdies die Moeglichkeit der Erweiterungsfaehigkeit, wie z.B. das META-II-System von Schorre (14), so koennen damit auch Compiler fuer andere Sprachen hergestellt werden. Das Prinzip einer solchen Erweiterung kann der Abbildung I entnommen werden. Damit ergibt sich die Moeglichkeit der Implementierung einer Sprache Y durch eine vollkommene Eingabesteuerung, naemlich durch die Spezifikation der Sprache Y durch ihre Metabeschreibung My. Der aus My resultierende Compiler enthaelt "alle und nur die" Syntaxregeln der Sprache Y, d.h. er definiert die Sprache Y. Dieses Prinzip ist ohne Einschraenkung auf Sprachen mit einer geschlossenen Syntax (5) anwendbar. Da die Semantik einer CAMAC Sprache, die sich lediglich auf die I/O-Kontrolle beschraenkt, ebenfalls als geschlossen angesehen werden kann, sollte es demnach sehr leicht möglich sein, auf diese Weise einen Compiler fuer eine solche CAMAC Sprache zu erzeugen.

6. Zusammenfassung

Das Konzept einer CAMAC-I/O-Sprache wird damit einerseits den Forderungen der Real-time-Anwendung nach einer Softwareintegration der CAMAC Hardware gerecht und bietet andererseits die Möglichkeit, den Aufwand bei der Erstellung des CAMAC Uebersetzers in Grenzen zu halten. Von entscheidender Bedeutung fuer die Ausbaufaehigkeit und Flexibilitaet des Compilers ist weiterhin die Verwendung eines Metacompilers oder eines aehnlichen Systems im Compilerdesign. Nach eigenen Erfahrungen des Autors (3)(15) mit einem auf dem Meta-II-Prinzip von D.V. Schorre basierenden Metacompiler kann damit ein sukzessiver Ausbau eines CAMAC Compilers erfolgen, der urspruenglich von einem relativ kleinen Subset einer CAMAC Sprache, CAMAC IML, ausging.

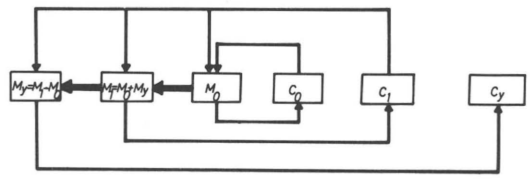


ABB I M_o - Metabeschreibung von C_o
C_o - Compiler für M_o
M_1 - Metabeschreibung von C_1
C_1 - Compiler für M_1
M_y - Beschreibung der Sprache Y
C_y - Compiler für Sprache Y

Referenzen

(1) SCHWEIZER, G., MALL, M. Das Camac-System, ein Schritt zur standardisierten Prozessperipherie. Regelungstechnische Praxis und Prozess-Rechentechnik, 6, (1973), 136.

(2) MICHELSON, J. Concepts of the VC and the IML. SWG 12/73, (1973), (unpublished).

(3) KNEIS, W. General Aspects of Translation and Implementation of the CAMAC-IML. SWG 13/73, (1973), (unpublished).

(4) CRESPI-REGHIZZI, S. et al. The Use of Grammatical Inference for Designing Programming Languages. Comm. of the ACM, 16,2 (1973),83.

(5) GINSBURG, S. The Mathematical Theory..., McGraw Hill (1966)

(6) EUR4100. Camac, a Modular Instrumentation System for Data Handling

(7) EUR4600. Camac, Organisation of Multi-Crate Systems Specifications of the Branch Highway and Camac Crate Controller Type A.

(8) PEARL - A Proposal for a Process- and Experiment Automation Realtime Language, Ges. fuer Kernforschung, KFK-PDV1, April (1973).

(9) KNUTH, D. E. Semantics of Context-Free Languages. Mathem.Systems Theory, 2,2,(1968), 127.

(10) RUSSEL, R. D. Intermediate-Level Programming Languages for On-Line Data Acquisition and Control. Computer Phys. Communications, 5, (1973), 89.

(11) CHOMSKY, N. Syntactic Structures. Mounton, s'Gravenhage, (1957).

(12) BACKUS, J.W. The Syntax and Semantics of the Proposed International Algebraic Language of the Zuerich ACM-GAMM Conference. Information Processing, UNESCO Paris, (1959).

(13) FELDMANN, J., GRIES, D. Translator Writing Systems. Comm. of the ACM, 11 (1968), 77.

(14) SCHORRE, D.V. Meta II, a Syntax-Oriented Compiler Writing Language. Proc. of the 19th Natl. ACM Conference, (1964), D1.

(15) KNEIS, W. (to be published).

CAMAC FACILITIES IN THE PROGRAMMING LANGUAGE PL-11

R. D. Russell
Data Handling Division, CERN, Genève, Switzerland

ABSTRACT

This paper describes the features for programming CAMAC easily and efficiently in the programming language PL-11.

ZUSAMMENFASSUNG

Dieser Vortrag beschreibt, wie mit den Fähigkeiten der Programmiersprache PL-11 auf einfache Weise wirkungsvolle Programme für CAMAC geschrieben werden können.

RESUME

Cet article décrit les instructions permettant une programmation facile et efficace de CAMAC en langage PL-11.

Introduction

PL-11 is an intermediate-level, machine-oriented programming language for the PDP-11 computer (1). It was designed and implemented as the programming tool for the on-line mini-computers used in data acquisition at the CERN OMEGA project (2). Because all the experiment electronics are connected to the computer via CAMAC, facilities have been incorporated into PL-11 that enable the physicists to program both CAMAC and the PDP-11 in a single language. This paper, which is a condensation of an earlier paper (3), discusses just the features of the language that are related to CAMAC.

1. Design Philosophy

There are three basic requirements that must be met by any CAMAC programming system if this system is to be an easily used and understood tool for effective use of CAMAC.

1. The programmer must have the ability to give meaningful symbolic names of his choice to any CAMAC C-N-A triple. The association of a name with a triple appears exactly once in any program (in PL-11, as a declaration at the beginning of the program), and all operations with this CAMAC module are done by referring to this symbolic name. Not only does this make the program more readable, it also makes modification of a CAMAC configuration extremely simple. One has only to change a single declaration and recompile the program. The compiler will then ensure that the addresses in all operations to this module are changed properly.

2. The programmer must have the ability to give meaningful symbolic names of his choice to any CAMAC function.

3. The programmer should be able to use the same syntactic notation for CAMAC variables and functions as he uses for ordinary program variables and functions. This includes the ability to test Q responses in the same manner as normal Boolean tests are made, the ability to index through arrays of CAMAC variables in the same manner as normal arrays are indexed, etc. Such a consistent use of a high-level language syntax makes it easy for a non-professional programmer to utilize CAMAC effectively, since he is able to manipulate CAMAC data in exactly the same way as he deals with normal data.

PL-11 is designed to be syntactically as high-level a language as possible so that readable, structured programs can be written easily. At the same time, PL-11 is semantically as close to the machine architecture as necessary in order to guarantee efficiency and the ability to utilize fully all features of the hardware. Therefore, in addition to the three requirements listed above, there is a fourth requirement satisfied by the CAMAC facilities in PL-11:

4. The compiler should produce efficient, in-line object code for every CAMAC operation. There should be no subroutine-calling overhead and no required run-time system to support the CAMAC features of the language. This implies that an entire real-time operating system can be written in PL-11 without the use of any other language and without any necessary run-time environment.

2. Declaring CAMAC Variables

The purpose of CAMAC variable declarations in PL-11 is to give programmer-defined symbolic names to CAMAC registers so that all operations to CAMAC can be done by reference to these names. The syntactic form of such declarations is analogous to the PL-11 construct that enables the programmer to give a symbolic name to any machine register, memory cell, or I/O device register.

```
INTEGER PRINTBUFFER SYN MEMORY($177566),
        PRINTSTATUS SYN MEMORY($177564);
```

gives the symbolic name PRINTBUFFER to the memory cell with absolute address 177566 octal, and the name PRINTSTATUS to the cell with absolute address 177564 octal.

```
INTEGER CRTBUFFER SYN CRATE 1 STATION 2
                      SUBADDRESS 14,
        CRTSTATUS SYN CRATE 1 STATION 5
                      SUBADDRESS 0 GROUP 2;
```

gives the symbolic name CRTBUFFER to the CAMAC register in crate 1 station 2 subaddress 14 group 1 (by default), and the name CRTSTATUS to the group 2 register at subaddress 0 station 5 of crate 1. In such declarations, the compiler will accept only crate numbers in the range (1,7), station numbers in (0,31), subaddress numbers in (0,15), and group numbers in (1,2).

It is important to note that both memory and CAMAC integers are 16-bit quantities. In keeping with the philosophy of designing a fully integrated system it was decided very early that since a PDP-11 word is only 16 bits all CAMAC modules used in OMEGA would utilize only the low-order 16 bits of the 24 data bits provided in CAMAC. This decision has proven to be correct: all programming errors due to a mismatch of precision are eliminated, and no time is lost packing and unpacking PDP-11 words. Since many 16-bit CAMAC modules are commercially available, no penalty was paid in higher costs or lost data. Clearly this decision also made it easier to integrate CAMAC variables into PL-11.

3. Using CAMAC Variables

Once a CAMAC variable has been declared, it is used just like any other PL-11 variable. Assuming the declarations written above,

```
X => PRINTBUFFER;
```

moves the contents of location X into location PRINTBUFFER, which causes a character to be typed on the teletype due to the definition of PRINTBUFFER as absolute address 177566 octal (the teletype data register).

```
X => CRTBUFFER;
```

writes the contents of location X into the CAMAC register CRTBUFFER, which presumably causes the module at that CAMAC location to display this character on a CRT.

```
PRINTSTATUS => Y;
```

moves the contents of PRINTSTATUS (the teletype status register) into location Y.

```
CRTSTATUS => Y;
```

reads the contents of CAMAC register CRTSTATUS into memory location Y.

Of course it is not necessary to store these status values in memory as they can be tested directly as follows:

```
WHILE PRINTSTATUS = BUSY DO ;
```

is an empty wait loop for the teletype to become "not busy".

```
WHILE CRTSTATUS = BUSY DO;
```

is a wait loop for the CAMAC display modules to indicate "not busy" in the same manner. To test for an ordinary variable X in some range we can write:

```
IF X >= LOWLIMIT & X <= HIGHLIMIT THEN
    BEGIN  COMMENT  WITHIN RANGE; ... END;
```

If SCALER is declared as a CAMAC register, we can write a similar test:

```
IF SCALER >= LOWLIMIT & SCALER <= HIGHLIMIT
THEN BEGIN COMMENT WITHIN RANGE;...END;
```

If we wish to wait until SCALER reaches some value THRESHHOLD, we write:

```
WHILE SCALER < THRESHHOLD DO ;
```

In all of these examples, the PL-11 compiler generates for each CAMAC reference the two in-line instructions that are required by the manufacturer's CA11 interface for a CAMAC operation. The first instruction sets up the crate and function codes, the second sets up the station and subaddress codes and actually performs the transfer. Hence this method is "as efficient as possible" as well as being easy to use and understand.

4. CAMAC Arrays and Block Transfers

A block of ten CAMAC registers starting at station 1 subaddress 0 of crate 1 can be declared in PL-11 as:

```
ARRAY 10 INTEGER SCALER SYN
         CRATE 1 STATION 1 SUBADDRESS 0;
```

If we also define a compile-time constant SPACING to be the address increment between consecutive SCALER modules:

```
EQUATE SPACING SYN 2;
```

then the statement:

```
FOR R1 FROM 0 STEP SPACING UPTO  9*SPACING DO
           0 => SCALER(R1);
```

will generate code to reset to zero ten scalers in consecutive subaddresses at the same station. The same statement will reset to zero ten scalers in consecutive stations if we instead declare:

```
EQUATE SPACING SYN 32;
```

In order to accumulate the sum of these scalers in a variable SUM with the average in variable AVERAGE, we can write:

```
0 => SUM;
FOR R1 FROM 0 STEP SPACING UPTO 9*SPACING DO
             SUM + SCALER(R1);
SUM => AVERAGE/10;
```

If we wish to store the ten scaler values in a ten element array X, we can write:

```
0 => R2;
FOR R1 FROM 0 STEP SPACING UPTO  9*SPACING DO
    BEGIN
    SCALER(R1) => X(R2);   R2 + 2;
    END;
```

where we have to keep an extra index register R2 to account for the constant spacing of 2 between consecutive integers in memory.

This last example demonstrates a transfer of ten words in which each word transferred requires a CAMAC read operation. If this scaler block included a control module that allowed the entire block to be transferred with a single command to that module, this could also be done in PL-11 as follows:

```
INTEGER BLOCKSTART SYN CRATE 1 STATION 3
                       SUBADDRESS 0;

REF(X) => BLOCKSTART;
```

where REF(X) is the memory address of array X and we assume that initializing the BLOCKSTART control module with a memory address will cause the block of scalers to be transferred into memory starting at that address. A more common situation is to include in the CAMAC control module a block size register, to define how many words to transfer. Then the block transfer is written in PL-11 as:

```
REF(X) => BLOCKSTART;    10 => BLOCKSIZE;
```

5. Other CAMAC Functions

The preceding sections demonstrated the CAMAC read, write, and reset functions in PL-11 and showed that they could be written with the same syntactic form as normal variable operations in PL-11. There are, however, 32 possible CAMAC functions, and any or all of these can be given symbolic names by PL-11 declarations such as the following;

```
CAMAC FUNCTION ENABLE(26), DISABLE(24),
               TESTLAM(8), CLEARLAM(10);
```

where the symbolic name is chosen by the programmer, and the function number must be in the range (0,31). These functions can be applied to any CAMAC variable using the normal function notation:

```
        ENABLE(SCALER);
        DISABLE(CRTSTATUS);
        CLEARLAM(SCALER(R1+8));
```

For example, if we wish to read a block of scalers into an array X such that each scaler is read only when it responds to a TESTLAM function, we can write in PL-11:

```
    0 => R2;
    FOR R1 FROM 0 STEP SPACING UPTO 9*SPACING DO
            BEGIN
WAIT:       TESTLAM( SCALER(R1));
            IF NOT Q THEN GOTO WAIT;
            SCALER(R1) => X(R2);
            R2 + 2;
            END;
```

where we have used the name Q which is a pre-declared PL-11 name for a bit in memory that is set or reset by the CAMAC CA11 interface according to the Q-response of a CAMAC operation.

Interrupts caused by CAMAC modules can be handled in PL-11 by declarations such as:

```
CAMAC INTERRUPT 20 PROCEDURE SCALEROVERFLOW:,
     BEGIN . . . END;
```

This declaration is just an extension to CAMAC of the interrupt handling mechanism present in the PL-11 language. It causes the compiler to set up transfer vectors by which the operating system can call this procedure whenever an interrupt is caused by LAM bit 20. The interrupt number must be in the range (1,24), and all interrupt enabling, disabling, clearing, and resolution of multiple LAM sources can and must be programmed explicitly with PL-11 functions.

6. Conclusion

The CAMAC features integrated into PL-11 are efficient, easy to use, and easy to understand. They make the full range of CAMAC functions available to the programmer in a simple symbolic notation. This has been accomplished by a slight extension of an existing language to include CAMAC variables, functions, and interrupt procedures. Clearly this technique can be applied to other languages, such as FORTRAN, ALGOL, and especially PL/I since its structure allows an easy extension by the addition of a CAMAC attribute. The PL-11 compiler is able to generate highly efficient in-line code for each CAMAC operation due to the simple design of the CA11 interface between the PDP-11 and CAMAC. For a more complex interface a compiler would probably have to generate calls to system subroutines, but the notation used by the programmer would remain unchanged (this is the technique used for FORTRAN floating-point operations on many small computers that lack floating-point hardware). It is suggested that such an extension of a widely used language, such as ALGOL, FORTRAN, or PL/I, along these lines would be a highly desirable way to provide a universal tool for programming CAMAC.

References

1. RUSSELL, Robert D. 'Preliminary Specifications for PL-11: A Programming Language for the DEC PDP-11 Computer', OMEGA Project Development Note SW-29 (Nov. 1971).

2. RUSSELL, Robert D. 'The OMEGA Project: A Case History in the Design and Implementation of an On-line Data Acquisition System', Proceedings of the 1972 CERN Computing and Data Processing School CERN 72-21 (Dec. 1972) PP 275-340.

3. RUSSELL, Robert D. 'Easy and Efficient CAMAC Programming in PL-11', Paper presented at the Third European Seminar on Real-time Programming (May 1973).

POLYP, A POWERFUL HOST-LANGUAGE FOR CAMAC

G. Musstopf
SCS Scientific Control Systems GmbH, Hamburg, Germany

ABSTRACT

The application of CAMAC hardware requires not only the extension of operating systems but also the existence of suitable languages and compilers. This leads to the demand for host-languages for CAMAC.
The past has shown that not all existing languages and compilers are suitable as host-languages. The reason for this is to be found in the structure of these languages.

ZUSAMMENFASSUNG

Der Einsatz von CAMAC-Hardware fordert nicht nur die Erweiterung des Betriebssystems, sondern auch die Existenz geeigneter Sprachen und Compiler. Daraus ergibt sich die Suche nach einer Gastsprache für CAMAC.
Die Vergangenheit hat gezeigt,dass sich nicht alle existierenden Sprachen und Compiler als Gastsprachen eignen. Der Grund dafür ist in der Struktur der betreffenden Sprachen zu finden.

RESUME

La mise en exploitation de l'équipement CAMAC exige non seulement l'élargissement dy système moniteur mais aussi l'emploi de langage et de compilateur adéquats. Cela mène à la recherche d'un langage "substitut" assurant pleinement l'efficacité d'action du complexe CAMAC.
Les expériences passées ont montré que la plupart des langages et compilateurs existants ne se prêtaient pas efficacement comme langage "substitut". La raison de cette faiblesse réside dans la structure même des langages utilisés.

1. Situation

Due to a remarkable increase in efficiency connected with a considerable decrease in cost more and more new areas of application are being disclosed especially in the field of computerized process control. At the same time the number of concerned with the operation of automatic data processing systems is growing. These tendencies constitute a significant reason for the development of new special languages. The variety of hardware offered on one hand, and the large number of compilers needed for application oriented languages on the other, point out the necessity of searching for new methods.

When CAMAC hardware is used in a real time installation the operating system must be extended. Consequently the programming language, such as FORTRAN, must also be extended and the compiler correspondingly modified. This work can in general only be undertaken by system specialists. The compiler modifications must be maintained together with the system software.

Considering the large number of available real time computers for process control it would be uneconomical to develop compilers for special CAMAC programming languages such as IML and CAMAC-language (1). A solution to this problem is to use host languages such as CORAL66 (2) or POLYP (3). In other words: the design of language and compiler provides for an extendable interface to the CAMAC functions.

2. Requirements to be met by a software-production system

A software production system is understood to comprise both hardware (the processor) and software; i.e., language, compiler, operating systems, and testing aids essential for the development, testing, and the maintenance of programs. This does not necessarily imply that the software production system, or the target processor for which the development takes place be operated on the same hardware. In the field of computerized process control the design of a software production system requires that the phases system test and program maintenance be assigned a high degree of importance.

2.1 Requirements to be met by a programming language

Requirements which are repeatedly listed are:

A programming language should be
- machine independent,
- application oriented,
- self documenting
- easy to learn.

For the fields of application mentioned initially it is furthermore expected that at least at critical program points the efficiency of the generable object program, with respect to its storage and timing behaviour, largely matches that of a good, hand-coded assembler program.

The above mentioned requirements may be summarized in one sentence:
The programming language must liberate the user from having to use "EDP-slang" and allow him to concentrate on the formulation of the problem.

2.2 Requirements to be met by a Compiler

Usually a compiler is to fulfill the following requirements:
- choice optimizations,
- sophisticated error detection and analyses
- implementation of a language subset which is as large as possible and which contains no elements not belonging to the defined language fullset,
- availability of test aids for the run-time-system.

An analysis of compilers for small and medium sized systems indicates that many limitations to the language and reductions in comfort must be accepted for an implementation on relatively small EDP systems. However, the installation of so-called cross compilers on larger machines often leads to a remarkable extension of the language volume and to more comfort.

2.3 Test aids

The majority of testing aids should be provided by the compiler. On the one hand, these are the routines and lists required for syntactic and (as far as possible) semantic error checking. On the other hand, much more expenditure is required for the run-time test, e.g., testing must be possible predominantly on source level through the provision of symbolic storage dumps or comparable means. Furthermore test sequence controls and data generators are necessary for testing of program modules. In other words, a simulation of the environment of the module to be tested must be possible. Where required, it must be possible to operate parts of these test aids dynamically. Hence, applicable control statements must be contained in the source program. It should be possible to modify and extend these test aids in order to adapt them to a specific application. This, however, leads to the question as to which elements necessary for testing problems should be incorporated into an application oriented language.

3. Extendable programming languages

As recent years have shown, the quest for a unique universal language appears to be neither reasonable nor has it been successful. The reason for this may be found in the large number of EDP-applications and for demand for application oriented languages. The production of a large number of compilers with appropriate comfort, however, is economically impossible with those productions methods currently in use.

A way out of these difficulties is offered by compiler generators, and by the installation of extendable programming languages. In this context, a compiler generator is understood as a program system, which from the syntax and semantics of a programming language, and the description of an EDP machine and an operating system, is able to generate a compiler for the specified language and hardware. From the presently known methods no satisfying solution to this problem can be expected.

Extendable programming languages are, in this context, understood as languages to which extensions are possible without modification to the corresponding compiler. Within this study only those programming languages are further examined which are extendable with respect to their semantics. According to the experience obtained until now this limitation is considered fully sufficient for the purpose of the above mentioned aims.

3.1 Nucleus of a language

The analysis of an application oriented programming language reveals, from the viewpoint of the compiler, a division into two largely independent sections:

- language elements relevant to the compiler (all elements influencing the decoding within the compiler),

- language elements irrelevant to the compiler (all elements which can be implemented into the compiler simply by extending internal lists).

The nucleus of a language consists of the compiler relevant language elements, e.g. data definitions, arithmetic and logical expressions, value assignments and subroutine calls. The interfaces for compiler-irrelevant elements must also be defined in the nucleus of the language. In this paper the language nucleus can be considered as an operating system independent host-language.

Typical elements irrelevant to the compiler are calls to numeric functions, to the operating system, to input/output routines, and to task management. Whether a closed or an open subroutine call, or a so-called system call is initiated, or a parametrized instruction sequence (macro) is inserted, is of no concern. CAMAC-driver calls are typical compiler-irrelevant language elements.

The comparison of the two classes of language elements clearly indicates which elements must be defined in the nucleus of the language to obtain the above mentioned interfaces. The most simple solution may be found by defining several classes of subroutines. These classes have an identical syntax; they differ only with respect to their incorporation into the compiler. Each class is given its own calling sequence.

The classification of language elements, however, has another important consequence: all language elements that contain operating system calls belong to the class of elements which are irrelevant to the compiler.

Consequently, the compiler and the nucleus of the language are independent of the operating system (as far as the object programs generated for a specific OS are concerned). The influence of the operating system may thus be excluded from further considerations.

3.2 Language extensions by routines

The extension of a language by the incorporation of further routines identified by standardized names is commonly used by hardware manufacturers and, less frequently, is also practiced by some users. Of course, this requires the user to check whether call or parameter conventions are violated.

The identifier list of a compiler which usually is preset with entries for certain standard routines, may be supplemented with any new subroutine entries in order to adapt it to extensions of the operating system, e.g. additional I/O drivers. From the user's viewpoint, however, the documentation of most of the compilers available to him is not sufficient

3.3 Extensions by semantic elements

A comparison between programs of the same application areas which, for example, are written in FORTRAN, demonstrates that parts of expressions, statements, or application groups often repeat themselves almost identically. These parts of a program can be parametrized and coded as new language elements.

Such new language elements are precisely those language elements specific to this application. This assertion immediately becomes plausible when for purposes of analysis programs from automatic test systems, or information systems, are examined.

Language extensions as described are not without severe restrictions, possible for languages and compilers available to the user.

4. Realization

The language and compiler features described in 3. have already been partially taken into consideration for languages like PL/I, or CORAL66.

For the purposes mentioned above this has completely been realized for the POLYP programming system. The significant constituent parts are:

a) a so-called problem language (like PL/I or ALGOL6o) with an extension containing elements of low level languages,

and

b) a so-called control language which has been designed for the modification of source programs (generalized macro language; fig. 1).

4.1 Language features

In the POLYP programming system, only the nucleus of the language in the sense of 3.1 is defined. All parts of the operating system or existing routines to be called as subroutines, may be integrated as normal, standard, I/O, or system procedures (see 4.2).

The language extensions described in 3.3 may be realised by means of the control language. Those program sequences which are repeatedly used in the specific area of application are resident on a library. As a consequence the source program written by the user comprises both elements of the control language as well as of the problem language. The control language is decoded and executed in the same way as a macro generator by a first stage, interpretatively operating compiler. The result is a modified source program which only contains elements of the problem language. The following second stage, generatively operating compiler, produces an object program. In the following linkage-editor-run the connections between the routines already available are established (see 3.2).

It is of significance to the practical application of such methods, that especially for applications in the field of computerized processcontrol also lower language elements are contained in the problem language, e.g. for data storage positioning or address computation and indirect addressing. Therefore, the insertion of assembler instructions need not be resorted to for critical program sections.

4.2 Compiler feature

It is hardly sufficient that a compiler for the required language and the selected hardware be merely available; it must also offer a reasonable comfort. This, however, does not apply to most of today's low cost small and medium-scale real-time processors. Thus, there is an increasing tendency towards the installation of cross compilers. Large-scale EDP-systems are used for the development of programs. Compilers contain a general part (central compiler) for translation of source programs into a machine-independent intermediate language. Finally a code generator produces an assembler program, or object code (fig. 2). It is of significance that such cross compilers are not restricted to connecting large-scale systems and real-time processors of one single hardware manufacturer only. A compiler must have a routine (see 3.2) that modifies the identifier list initialization in the compiler. Furthermore, each user can define his own private routines simply by providing appropriate control cards.

As mentioned earlier, a modification of the source program is to be performed prior to the compilation. The library required for this pre-run is able to adapt those program sections containing low level language elements (see 4.1). Such macros are an adequate means of fully exploiting the hardware used.

4.3 Test systems

The availability of elaborate testing aids-is of major importance to assure the punctual availability of the whole system. Test and error analysis must be possible at source program level. When a program error leads to an interruption the cause of the error and a symbolic storage dump at source level (source identifier name, converted values, etc. instead of hexadecimal addresses and storage contents) should be made available to the user. It should also be possible to call such a symbolic storage dump as a subroutine during a normal program run. The realization of these test aids in the POLYP programming system revealed that few amendments to the compiler were required. The test system is implementend by the aid of the control language. These testing aids may, at any time, be extended by the user in accordance with his specific appications.

Furthermore, by the method described above, the transfer of test work from the target proccesor to largescale systems is supplemented. A simulation of the target processor's hardware on the large scale system is not demanded because it is possible to generate executable modules through the installation of corresponding code generators. Nevertheless, a simulation of the module's environment remains necessary in order to perform module tests. These modules are also programmed in the same language. When testing on another processor it may, however, in some cases be necessary to simulate those features required for the proper functioning of the other operating system. As far as possible this is achieved through the usage of interface routines thus removing the necessity of completely copying the operating system nucleus.

5. Present state

The POLYP programming system has been available since the beginning of 1973. Installations have, up to now, taken place in the fields of laboratory automation recording of measurements data, operations planning, production control, and compiler and operating systems development. Compiler for the control languages and central compilers for the problem languages are available for the IBM/36o/37o, and for the Siemens 4oo4 systems. Code generators for IBM/36o and /37o, Siemens 4oo4, DEC PDP11, and TR86 are available. Further code generators are under development.

Experience gathered up to now shows a change of attitude. This does not apply to adaptation to the new language but to the new method.

The difficulties may be compared with those occuring when changing from basic assemblers to macro-assemblers. In general the specific language extensions should be definded prior to the programming phase.

The described method allows for the gaining of experience on special languages without risk to the respective projects.

An evaluation of such experience should also serve the determination of "future-proof" standards for these special areas.

References:

(1) ESONE COMMITTEE SOFTWARE WORKING GROUP. Proposal for a CAMAC Language, CAMAC Bulletin, 5 (1972),3

(2) Inter-establishment committee for computer applications, Official Definition of CORAL66 (197o)

(3) Musstopf, G. Das Programmiersystem POLYP, Angewandte Informatik, 14 (1972), 441


ORGANISATION
EDV
ANWENDUNGEN
Statistik
Verkehrssteuerung
Buchungssystem
Numerik
Fertigungssteuerung
Laborautomation
Management Informationssystem
Forschung
Rechnungswesen
Prozeßlenkung
Marktforschung
Auskunftssystem
Lageroptimierung
Command + Control
Unternehmensspiele
Ausbildungssimulator
Statik
ANWENDUNGSPROGRAMME
LOPAC
MACRAD
IMS
Datenbanken
DAZU
UCOPY
MOSCOR
MINCOS
BOMP
ISI
CLASS
LISA
COSY
DAVID
UMPIRE
ATCAP
FLUSI
ALADIN
BOMM
TROPHY
PICS
BASIS
KWIC
KWOC
CICS
MMG
COSY
SESAM
MAUSI
MASTER
PRISMA
DAKS
ASMUS
BTAM
QTAM
MARS
ABEL
RESPOND
VSPX
BETRIEBS-SYSTEME
CP V
EROS
RSX
SCOPE
EXEC8
BSV
OS
TSO
STAR
BS3
RESY
BESY
PBS
DOS
OVERSY
SYMON
BPS
HARDWARE
ARGUS
R407
HP 3000
910
PDP 11
MITRA 15
PDP 10
GPA 73
EPL
Σ 530
1108
UYK 15
NAKED MINI
TR 440
H 4400
6600
/370
S 330
/7
4004
x8
ZEBRA
TR 86
1401
SERVICE
LASCIATE OGNI SPERANZA, OH VOI CH'ENTRATE
SPRACHEN COMPILER
MULI
MADAM
EXAPT
BCPL
RTL
PL/I
CORAL
ALGOL 60
COBOL
FORTRAN
PL 360
PS 440
ASSEMBLER
NORMIERTE PROGRAMMIERSPRACHEN
ENTSCHEIDUNGS-TABELLEN
FORTET
VORELLE
DETAB
TABTRAN
TABLEMASTER
ETAB
DECTAB
DESIGNER
PROGRAMMIERER

Compiler Structure

®SCS

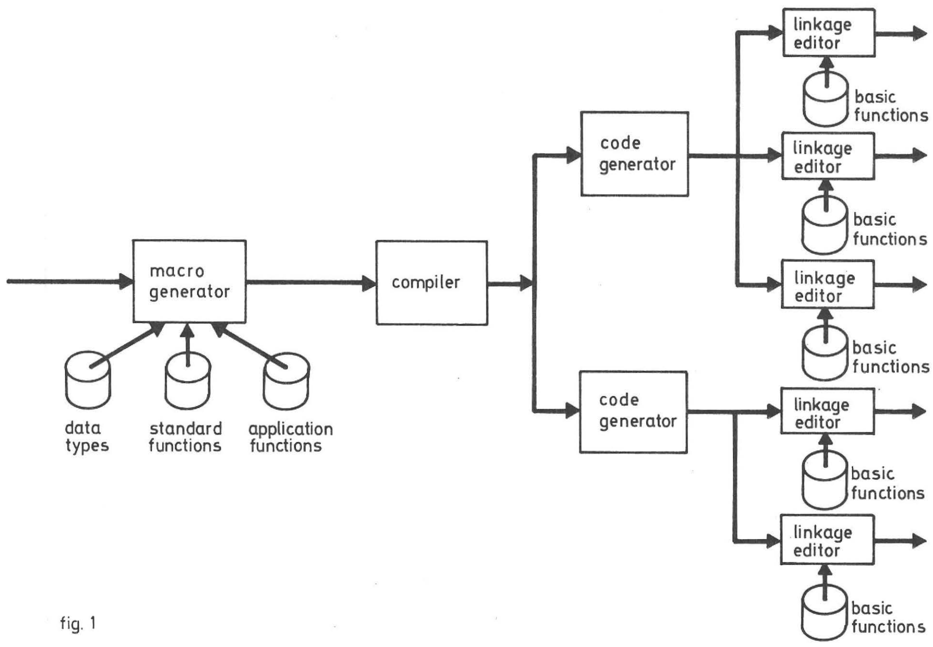


fig. 1

Macro Generator

®SCS

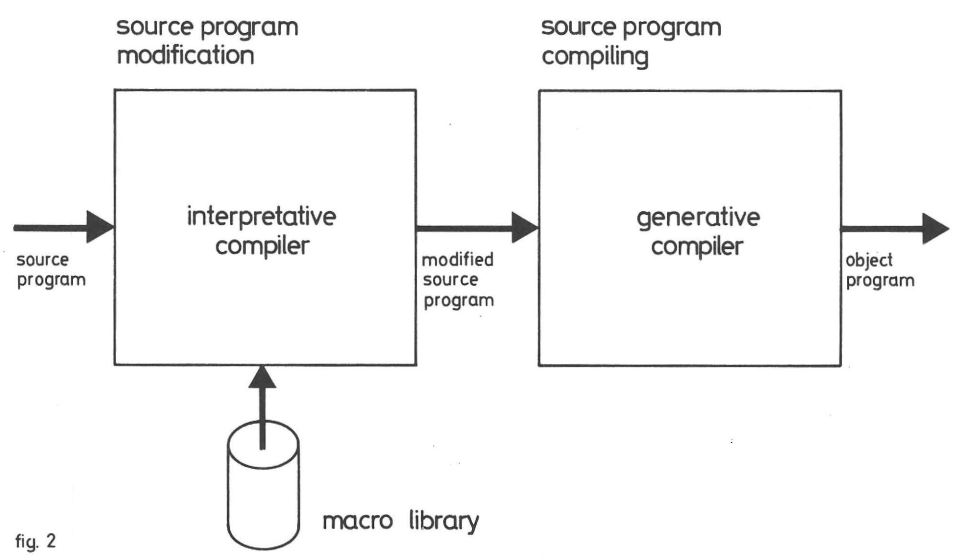


fig. 2

CAMAC AND PEARL - COMMENTS ON STANDARDIZATION IN LABORATORY AUTOMATION AND PROCESS CONTROL

V. H. Haase
UNICOMP, Blankenloch, Germany

ABSTRACT

PEARL is a programming language which shall replace assembly languages and is proposed as a software standard for real-time computer applications. PEARL has a modular program structure, powerful task- and I/O-management capabilities and allows to describe hardware configurations. It can efficiently be used to program CAMAC equipment.
Both hardware- and software-standards (CAMAC and PEARL) are necessary (cost reduction especially in complex applications).

ZUSAMMENFASSUNG

PEARL ist eine Programmiersprache, die die Assemblersprachen ablösen soll und als Norm für Realzeitanwendungen von Computern vorgeschlagen wird. PEARL-Programme sind modular aufgebaut, haben ein leistungsfähiges Task- und E/A-Management und gestatten auch die Beschreibung der Hardware. PEARL kann effektiv zur Programmierung von CAMAC eingesetzt werden.
Hardware- und Software-Standards (CAMAC und PEARL) sind notwending (besonders in komplexen Anwendungsfällen können die Kosten stark gesenkt werden).

RESUME

PEARL est un langage de programmation destiné à remplacer les langages assembleurs; il est proposé comme norme de software pour l'utilisation des ordinateurs en temps réel. Le programme PEARL a une structure modulaire, il possède de grandes possibilités de gestion des tâches et des entrées/sorties; il permet en outre de définir les configurations du hardware. PEARL peut être utilisé avec succès pour la programmation de l'équipement CAMAC.
Les normes hardware et software (CAMAC et PEARL) sont toutes les deux indispensables (réduction des coûts, spécialement dans les cas complexes).

1. The development of PEARL.

1.1 Introduction

Due to similar reasons that lead to the creation of CAMAC, i.e. cost reduction in process control by means of standardization, in several countries activities were started to develop a programming language for universal use in process control and experiment automation. This paper is a report on the language PEARL (developped by a German working group), its highlights, its relation to process control hardware (exspecially to CAMAC), and the attemps to forward it as a programming language internationally standardized.

1.2 The history of PEARL

Our developments of a real-time-language for process and experiment automation were started in 1969 by a group of people who had spent long time coding real-time problems in assembly language. The effort was sponsored by the "Studiengruppe Nuklearelektronik" the German ESONE member. At present time approximately 30 organisations (universities, hardware- and software-suppliers, computer users) are members of the PEARL working group, which itself is represented in a number of national and international professional organisations (e.g. the Purdue Workshop on Industrial Computer Languages). The group's output were two papers, the PEARL-"concept" /1/ published in 1970 and the PEARL-"proposal" /2/ including a syntax definition, published in 1973. The work of the group is coordinated by Peter Elzer and his PEARL secretariat at the university of Erlangen, and by Dr.L.Frevert, representing the German ministery of technology, which sponsors the PEARL development.

1.3 PEARL'S present state

Concurrently with the (tentative) final definition of PEARL implementations of PEARL-programming systems (compilers and operating systems) have been started at several places (computer manufacturers, universities ,system houses) for 5 different computers. The first running system is supposed to be available next year.
It is regarded necessary to have working PEARL-software as soon as possible and not to spend long time in discussions on standards and perhaps still better features before (all this can be done in parallel). We want to have running systems so that we can win experience out of practical work and achieve improvements, which afterwards will be regarded when a standard for a procedural language for process control is set up.

1.4 Correlation PEARL - CAMAC

CAMAC was designed without having in mind a special programming language (CAMAC language considerations were started when the basic hardware principles were already defined), and PEARL is not bound to a certain instrumentation interface hardware, it was developped to cover all conceivable designs and configurations of hardware. One of the most important features of PEARL is its ability to describe hardware configurations. The programmer has the possibility to write his application program without regarding the used hardware (using logically defined identifiers for his instrumentation). These definitions which are collected in the "system division" of the PEARL-program are used by compiler and operating system to interprete the "non-standard" I/O-statements addressing CAMAC or non-CAMAC process-control-hardware, in the optimal way.
Therefore the use of PEARL is independent of the use of CAMAC. But the software design of the PEARL I/O-system is easier and the execution of process-I/O-statements may be more efficient if a PEARL system can rely on CAMAC as process-interface hardware.

1.5 Efficiency vs. standards

There are both hardware- and software people who do not like to use both CAMAC and PEARL like concepts, because high-level and standardization are regarded to make systems inefficient. CAMAC is meant to be a comparatively complex interface to process instrumentation when there are only few I/O-devices and a straightforward way of transferring data. PEARL needs perhaps a computer configuration equipped with larger storage to hold its features designed for special real-time use.
Two arguments against this opinion: The first is that "higher level" reduces manpower. This counts exspecially in the software discussion. A program written in a procedural language costs less time for coding, testing and documentation than the same program written in assembly language, also in process control (where software costs at present are at least comparable to hardware costs). The second argument reflects standardization. As far as CAMAC is concerned: interfacing n different I/O devices to m different computers by n X m different methods was the state of the art before; the mere fact that a standard is <u>used</u> (it may even be a badly designed one) promotes cost reduction. As far as PEARL is concerned: count the number of programming languages an automation programmer has to learn now. If there is <u>one</u> language which is more or less useful for real-time programming and compilers are available for most of the commonly used process control computers, it will have good chances to become standard and make all other solutions obsolete.
To be fair, one argument against, resp. one criterium for standards should be taken into account: standards have to be flexible and easy to use. This is - at present time - not fully met by both CAMAC and PEARL, exspecially when seen from the great number of minicomputer applications. Substandards (simplified versions) and more flexibility towards extensions are needed.

1.6 Correlation between PEARL and CAMAC - languages.

This may be a delicate question in the context of this session. Both PEARL and the proposed CAMAC-language /3/ claim to be the solution for programming process control problems where CAMAC is used (from the PEARL point of view this solution is

not restricted to CAMAC). Both definition papers say a lot about transportability of programs, easier program writing and even easier translator writing. My comment: PEARL is designed to cover all problems which conceivably can appear in programming real time systems; a "danger" may be that it is too complex and not handy enough for smaller problems, but in no account it should be a language with missing features. - The proposed CAMAC language by definition is an uncomplete language which is to be complemented by any (?) conventional programming language. The statements describing the CAMAC-configuration and the symbolic functions standing for CAMAC-F-codes are a sufficient CAMAC extension of e.g. PL/1, they are not in the case FORTRAN is the host language. The real-time-statements in the CAMAC-language (e.g. LINK, DONOW) which may be necessary in a FORTRAN-host program, are not necessary or even conflicting with a host language on the PL/1 level. The scope of a pure CAMAC language does not seem well defined.
The way to take one defined host language and to add CAMAC - statements, probably is a very good short term approach. - Seen as developments for long term use a view on implementation is necessary, too. One "complete" real-time language needs newly designed and implemented compilers and operating systems, of course; The CAMAC-language approach leads to extensive modifications of the used host systems (due to the necessary real-time-features) which may be a greater amount of work in the end.

1.7 General comments on data-processing standards

Data-processing standards are far more the concern of businessmen than of computer scientists, i.e. they should reduce costs. The computer scientists' task is to see that the standards are the best possible from a professional point of view. Still they should not forget that users should work with new and better ideas as soon as possible - therefore not the most sophisticated but the shortest path which is useable is the best one. This is exspecially true for software, for programming languages. PEARL, the most important features of which are presented in the following paragraph, certainly is not the best approach in every detail, but it is a good one and will be available soon. That there is a real chance that PEARL or a PEARL-type language will become a standard could also be a reason for the real-time-CAMAC user to consider it carefully.

2. The basic features of PEARL.

2.1 The scope of PEARL

PEARL is a compilable middle-level language for process and experiment automation purposes. A middle-level language was chosen to be as machine-independent (therefore no low-level or multi-level language) and as problem-independent, i.e. open for any automation task, as possible (therefore no special "problem oriented language"). Flexibility, value for documentation, easy handling and easy learnability were regarded most important. To cover a wide range in the field of automation problems the language had to be sufficiently rich, so that it is reasonable to define appropriate subsets (a way which is followed by the current implementations). To avoid reinvention of useful elements of conventional languages as many elements as possible out of the algorithmic and the standard I/O-part of PL/1 /4/ have been integrated into PEARL.

2.2 Structure of PEARL - programs

In PEARL a program is built up out of one or several independently compilable entities called program modules. Thus corresponding to the specific automation problem a systematic construction of the program system is possible. To make the description of automation algorithms independent of the hardware configuration, PEARL makes a distinction between system description and problem description.
A module contains a "system division" or a "problem division" or both. In the system division identifiers are declared for terminal devices, by the use of these symbolic names automation algorithms can be represented device independent in the problem division.
Communication between PEARL modules is done via global objects which may be identified by additional library names; so modules are connected by controllable sets of names which also can be used for operator communication.

Example:
```
MODULE;
SYSTEM;   /*system division, compiled
.             separately */
.         /*hardware description */
MODEND;
MODULE;
PROBLEM;
DECLARE K INT GLOBAL LIB1;
DECLARE L INT GLOBAL LIB2;
/*first problem solving program,
   hardware independent */
MODEND;
MODULE;
PROBLEM;
DCL K INT GLOBAL LIB1;
DCL L INT GLOBAL LIBXYZ;
/*second problem program, compiled
   independently; the two Ks are iden-
   tical, the Ls different */
MODEND;
```

2.3 Real-time features and input/output

A language to be used for process automation must necessarily contain elements for the description of dynamic processes, and of data-transfers between non-standard I/O-devices.
PEARL contains a number of elements to control parallel execution of tasks, and to schedule them dependent on spontaneous or periodic events. Synchronisation features are as well provided as scheduling of system resources according priorities.
Task must be declared statically - this prescription gives the compiler the chance of generating effective code for task-management and storage-management operations - and are activated dynamically. There are also the functions "suspend" and "continue", "delay", "terminate", "schedule" and "pre-

vent". Scheduling means the execution of any other task operation on a defined point of time, within fixed time intervals or triggered by the occurence of an interrupt.

```
Examples: TASK T;        /*declaration */
          DECLARE P PROCEDURE = BEGIN

                                END;
          .
          ACTIVATE T: CALL P; /*activation */

          AT 8:0:0 ALL 2 HRS UNTIL 20:0:0
          ACTIVATE T: CALL P; /* schedule */

          ON INTERRUPT1 SUSPEND T;
          /* interrupt-scheduling; suspendation */

          DELAY T DURING 1 SEC; /* suspenda-
          tion for a specified amount of time */

          TERMINATE T;
          PREVENT T; /* prevent means cancellati-
          on of all schedules arranged for task T */
```

Synchronization of tasks is done via semaphores /5/ and lists of semaphores. Thus the states of a number of tasks can be used to trigger the execution within another task. Aggregates of semaphores can be handled as a whole; these "bolt-variables" are used to organize e.g. queuing problems.

```
Example:  REQUEST SEMA1;
          /* the task "stops" at this point until any
          other tasks executes the statement :    */
          RELEASE SEMA1;
```

The dynamic description of input and output is done via the task-mechanisms, too: I/O-statements are imbedded in tasks. To handle any process input/output one statement is sufficient:

```
          MOVE source TO sink; it may be followed
by:       GAUGE gauge-procedure.
```

Source and sink are either identifiers standing for adresses in working store or standing for devices (as declared in the system division of the program). Binary data can be transported to and from any terminal device in the system. The gauge procedure controlles conversion or scaling of data to be performed together with the transfer by a (central or peripheral) processing unit.

```
Example:  DCL SUBBACK PROC . . . /* procedure
          to subtract background */
          DCL ADC DEVICE; DCL DATA BIN;
          .
          MOVE ADC TO DATA GAUGE SUBBACK;
          /* variable"data"will receive the correc-
          ted value */
```

A further feature of PEARL are I/O-statements to handle graphic devices, e.g. displays, plotters and lightpens; graphic formats are adjusted to the requirements of the experimentator and process operator.

2.4 System division

The PEARL-programmer describes the whole hardware configuration of the process-control system and declares symbolic names for hardware elements used in the application program by means of the system division. There are several advantages of this concept.

a) the configuration of the system is documented (easily readable) in the program itself.
b) it is possible to generate an optimal operating system for the given problem and configuration.
c) the complete description of hardware-software interrelation makes superfluous a "job control language" specific for each computer.
d) the problem-programmer need not care about the path from data-source to data-sink; changing the hardware connections does not change the application program.

The PEARL system division describes the hardware as a network of connections between devices. Thus both tree-structures (e.g. CAMAC) and network-structures (e.g. in a multiprocessor computer system) can be handled.

```
Examples: COMP777 <-> CHN;
          APPARATUS: CHN *0 <-> BRANCH(1);
          BRANCH(1) * 2 <-> DX : CRATE(2);
          PROCESSX(1:5) : <-> DX :*(2:10/2)*0;
          /* comp777, chn, branch and crate
          are implementation defined names, they
          stand for device types. apparatus, dx
          and processx are identifiers chosen by
          the programmer and used in his problem
          division. */
          /* the first branch is connected to con-
          nection zero of the channel; the array
          of devices called processx is connected
          via subadresses zero of adresses 2,4,6,
          8 and 10 (2:10/2) of crate 2 - called
          dx - of branch one - called apparatus.
          all connections are bidirectional */
```

These system description and hardware-naming features are not limited to "strictly ordered" hardware as CAMAC. They are applicable to any interface logic, even for several different types within one system. Implementation in a pure CAMAC environment will certainly be easier.

3. Conclusions.

3.1 Final general comments

PEARL has been presented in this paper as an approach to make programming real-time computer applications cheaper and easier - exspecially when CAMAC is used, too. Three features should be stressed once again, being very important:
a) PEARL covers the full range of automation problems, both hardware-dependent and -independent algorithms can be expressed.
b) PEARL is a middle-level language; a program written in PEARL is transportable from one computer to another (compilers assumed to be available).
c) PEARL or a PEARL-like language is to become a long term standard for process-automation programming; therefore CAMAC and PEARL should be regarded together.

3.2 CAMAC - and PEARL - limitations

One comment on both CAMAC and PEARL should not be conceiled. They are expensive when you start with. The comfort in use is paid with comparatively complex systems even if the automation problem to be solved is not complex. The reason may be a fact that often appears in developments done by joint groups, compromizes are made by taking all the personal ideas of the members into the definition (even if they are contradicting). The "product" then is more complex than necessary. I do hope that the experience in the practical use of CAMAC and during the implementation phase of PEARL will lead to substandards which are cheaper and easier to use.

3.3 Recommendation

My last comments should not be misleading. The use of CAMAC is to be recommended in the majority of automation problems now. But using a good interface hardware is not enough. CAMAC-use has to be backed by the use of a proper automation programming language, a language that is as computer and problem and even CAMAC-independent as possible (you will never get rid of a piece of non CAMAC-hardware in your system). There are two ways to achieve this object: a short term one by adding CAMAC-features to a programming language which is well tried in real-time problems; a long term one by using a well designed universal real-time-language, e.g. PEARL.

References:

/1/ J. Brandes, S. Eichentopf, P. Elzer, L. Frevert, V. Haase, H. Mittendorf, G. Müller, P. Rieder: PEARL, The Concept of a Process- and Experiment-oriented Programming Language; elektronische datenverarbeitung, 10(1970), 429-442

/2/ K. Timmesfeld, B. Schürlein, P. Rieder, K. Pfeiffer, G. Müller, K. Kreuter, P. Holleczek, V. Haase, L. Frevert, P. Elzer, S. Eichentopf, B. Eichenauer, J. Brandes: PEARL, A Proposal for a Process- and Experiment Automation Real-time Language; Gesellschaft f. Kernforschung mbH, Karlsruhe, PDV-Bericht KFK-PDV 1, April 1973

/3/ ESONE Committee Software Working Group: Proposal for a CAMAC Language; CAMAC Bulletin, No. 5, Nov. 1972

/4/ PL/1 Language specifications, IBM C28/65 - 71

/5/ E. W. Dijkstra: Co-operating sequential processes; published in "programming languages", edited by F. Genuys, Academic Press, London 1968

DISCUSSION

Q - H. Woda

In an earlier talk today it has already been mentioned that the success of CAMAC will be dependent on the development of a flexible programming language for process control, and related standardization. PEARL appears to be a successfull attempt in this direction. However its requirements exceed the capabilities of small computers. Can you comment on attempts to adapt PEARL for small computers (subsets) and are there other solutions for this class of computers, that may be in a state more readily available than PEARL.

Q - A. Pätzold

PEARL was developed as a programming language for process control and experiment automation.
CAMAC is one of the applications for PEARL.
Is it planned to have a minimized version of PEARL especially for CAMAC?
If it is planned, when will it be available?

A - V. Haase

The current implementations of PEARL programming systems (PDP-11, AEG-6050, Siemens 305 and 330) do not cover full PEARL; the chosen subset will be able to handle CAMAC-applications.
For smaller machines (e.g. Mincal, Mulby, Unicomp) a more restricted subset called 'MINIPEARL' is being defined by a working group coordinated by the Werkzeugmaschinenlabor, RWTH Aachen. The date of availability of MINIPEARL will probably not be before 1976. CAMAC-'compatibility' is regarded as being very important in this development.
My opinion is, that before that time compound or multilevel languages based on e.g. IML and BASIC are the best way to program CAMAC (as shown by the developments at CERN, KFA Jülich, SCS-POLYP, UNICOMP-Process BASIC and others).

CASIC : A CAMAC EXTENDED BASIC LANGUAGE

J. M. Servent
Schlumberger Instruments et Systèmes, Centre d'Instrumentation Nucléaire, Bagneux, France

ABSTRACT

Designed for use with CAMAC in conjunction with a PDP 11 family computer, the CASIC is a programming language which provides all the standard statements of the BASIC, plus a set of statements for CAMAC purposes.
It is a conversational language allowing the user to execute any CAMAC function and to handle the LAM's interrupts. By generating, internally, a binary program in computer language, the execution speed is improved by a factor 10, compared to a purely interpretor program.

ZUSAMMENFASSUNG

CASIC ist eine für den Einsatz von CAMAC in Verbindung mit einem Rechner der PDP 11 Familie entwickelte Programmiersprache, die alle Standardanweisungen von BASIC sowie eine Zusammenstellung von Anweisungen für CAMAC-Zwecke umfasst.
CASIC ist eine Sprache für Dialogverkehr, mit deren Hilfe der Benutzer alle CAMAC-Funktionen ausführen und die Anforderungen (von LAM Quellen) verarbeiten kann.
Durch internes Generieren eines Binärprogramms in Maschinensprache wird die Ausführungsgeschwindigkeit gegenüber einem Interpretierprogramm um den Faktor 10 verbessert.

RESUME

Le CASIC est un langage de programmation qui offre toutes les instructions classiques du BASIC plus un jeu d'instructions spécialement destinées au CAMAC. Il a été conçu pour les systèmes CAMAC avec calculateur de la famille PDP 11.
Le CASIC est un langage conversationnel permettant à l'utilisateur d'exécuter toute fonction CAMAC et de traiter les appels. De plus, la génération interne d'un programme binaire en langage machine accroit d'un facteur 10 environ la vitesse d'exécution par rapport à un programme en interpréteur.

1 - INTRODUCTION

The CIN-SCHLUMBERGER software group has developped a CAMAC extended BASIC language, the CASIC,to give the user a valuable tool to solve some practical problems of using CAMAC with a minicomputer. CASIC has been designed to work with a PDP 11 family computer through our branch interface ICP 11 or our crate interface JCC 11. Due to the fact that the CASIC program is not an interpretor but a macroinstructions generator, the execution speed is improved by a factor of about 10, compared to the conventional BASIC itself. Another interesting feature of CASIC is its ability to handle LAM's interrupts ; this can open a wider field of application by allowing the user not only to write his tests or debugging programs when checking his system, but also to write more powerful programs to handle his experiment.

2 - DESCRIPTION OF CASIC

Besides the statements available in BASIC itself, the CASIC offers the following set of statements dedicated to CAMAC use.

2.1 - Naming statement

Giving a name to a CAMAC register can be done by writing :

LET < name > = STA < Camac address >

where : < name > can be 1 to 3 significant alphanumeric characters which must be preceded by the % character.

: < Camac address > = < b,c,n,a >
with b giving the branch number
c giving the crate
n giving the station
a giving the subaddress

b,c,n, and a can be any constant, or expression such as ;3,K, F+1, N-4, A↑2 etc ... provided they are equal to a convenient integer.

For example :

```
...
110 LET %1    = STA (1,2,18,0)

120 LET %S8   = STA (2,4,N+3,A+2)

130 LET %REG  = STA (B+1,C+1,P,M)
```

where N,A,B,C,P and M are variables previously defined in the program.

2.2 - CAMAC operation statements

The general form of a CAMAC operation statement is :

LET <dest> = CAM (<source>,<function code>)

2.2.1 - READ operation

in this case :

- <dest> is a variable which is given the value of the result of reading
- <source> is the CAMAC register to be read
- <function code> is :
  . either " READ " or " Ø " for F(Ø)
  . or n for F(n) with n = 1 to 7

Example :

```
10 LET R1 = CAM (%S8, READ )

20 LET Y  = CAM (STA (1,2,12,3),2)
```

2.2.2 - WRITE operation

in this case :

- <dest> is the CAMAC register to be written in
- <source> is any constant, variable or expression of the program
- <function code> is :
  . either " WRITE " or " 16 " for F(16)
  . or n for F(n) with n = 17 to 23

Example :

```
10 LET %REG = CAM (2↑6, WRITE)

20 LET %12  = CAM (CAM(%REG, READ),16)
```

Statement 10 will write 64 in %REG CAMAC register and statement 20 will write in %12 CAMAC register the content of %REG.

2.2.3 - Non READ - WRITE operations

in this case, there is no destination, the general statement is simplified to :

LET CAM (<source>,<function code>)

where <source> is the CAMAC register
and <function code>is n =8 to 15 or 24 to 31

Example :

```
10 LET CAM (STA(B,C,30,10),26)

20 LET CAM (%REG,8)
```

Statement 10 will enable Crate C in branch B. Statement 20 will test LAM in %REG.

2.2.4 - Conditional branching on X or Q response

After a CAMAC operation statement has been executed, two special registers %XCAM and %QCAM are set according to X and Q responses and branch statements can be used such as :

```
IF %XCAM = 1 THEN  GOTO <other program
                         address>
               or  PRINT
               or  <other statement>

IF %XCAM × %QCAM = 0 THEN  LET A =A+1
```

2.3 - LAM's handling

Having been designed to be interruptible, CASIC provides the two following statements for LAM's handling :

2.3.1 - ON LAM <CAMAC station> THEN GOTO x x x

This statement will " activate " the subroutine defined to process the LAM emitted by <CAMAC station>. If this LAM occurs before the program has reached this statement, it will not be processed till that moment. So, this statement must be properly placed in the program.

NOTA : The subroutine must be terminated by a RTI statement.

Example :

```
...
...
 40  ON LAM %REG THEN GOTO 100
...
...
...
100  LET  K = CAM (%REG,2)

110  PRINT  " Z = " ↄ↑LOG(K)

120  RTI  %REG
```

2.3.2 - WAIT <CAMAC register>

This statement can be used to know if the LAM from <CAMAC register> has been received and processed and to wait for it if it has not.
It is possible to write

```
WAIT  %REG, %S8 , %1
```

3 - SHORT PROGRAM EXAMPLE :

```
10  LET N,A,S = 0

20  FOR N = 1 to 20  STEP 2

30  FOR A = 1 to 15

40  LET %SCA = STA(1,2,N,A)

50  LET  R = CAM(%SCA,READ)

60  IF %XCAM × %QCAM ≠ 0 THEN PRINT "SCALER"
                          N "," A "=" R

70 LET S = S + R × %XCAM × %QCAM

80 NEXT A

90 NEXT N

100 PRINT "TOTAL =" S
```

This program prints the contents of a group of scalers in scanning mode with Q and X tests and the sum of all the contents.

SOME ASPECTS OF A COMPILER WRITING SYSTEM AND THE IMPLEMENTATION OF THE CAMAC LANGUAGE

K. H. Degenhardt, U. Marx-Rehbein, W. Woletz
Hahn-Meitner-Institut für Kernforschung GmbH, Berlin, Germany

ABSTRACT

Recent work at the Hahn-Meitner-Institut to automate the writing of compilers for programming languages is presented. The first application of the Compiler Writing System (CWS) will be the implementation of a compiler for the CAMAC language.

ZUSAMMENFASSUNG

Neuere Arbeiten am Hahn-Meitner-Institut zur Automatisierung des Schreibens eines Compilers für eine Programmiersprache werden vorgestellt. Die erste Anwendung des Compilergenerators (Compiler-Writing System, CWS) wird ein Compiler für die CAMAC-Sprache sein.

RESUME

Présentation des travaux récents effectués au "Hahn-Meitner-Institut" en vue d'automatiser l'écriture des compilateurs utilisés pour les langages de programmation. La première application du système d'écriture de compilateur "Compiler Writing System (CWS)" sera la mise en oeuvre d'un compilateur du langage CAMAC.

INTRODUCTION

The purpose of a compiler writing system (CWS) is to simplify the implementation of compilers. The main components of a CWS are:

- a language for describing the syntax of a programming language,
- a semantic language, i.e. a language in which sematic routines are written.

The term compiler-compiler (CC) is often used because a CWS compiles other compilers.

A CWS[1] works as follows:

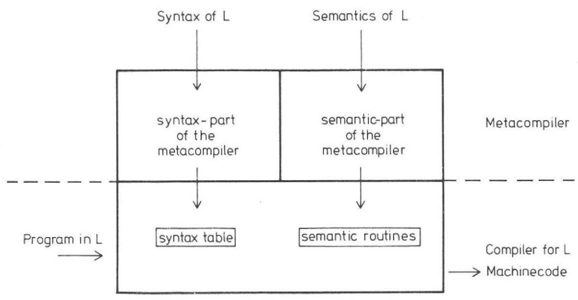


Fig. 1 A compiler writing system

Suppose a compiler for some language L, e.g. the CAMAC language[2] is required. The input to the CWS consists of the formal syntax of L written in a metalanguage and a set of semantic routines (semantic metalanguage). The CWS compiles the syntax into a table and the semantics of L into routines in some language (e.g. FORTRAN). The syntax table is used by a standard parsing routine to perform the recognition and parsing of programs written in L. The information contained in the semantic routines describes the meaning of statements in L. Semantic routines are associated with syntactic constructs. These routines are executed when syntactic constructs are recognized. Generally a compiler for some language L generated as described above will be less efficient in memory and compile time than a compiler written in assembly language. The reasons for using a CWS for the construction of compilers are similar to those for using high-level languages for programming.

AUTOMATICALLY CONSTRUCTED RECOGNIZER

Weak precedence

Generally the notation (metalanguage) used for describing the syntax of programming languages is the Backus-Naur-FORM (BNF) or some extension of it. The syntax of a language L can be defined by a set of rules (productions) written in BNF. The symbols (reserved words, delimiters, identifiers, etc.) used in these rules form the alphabet of L.

There are three types of precedence relations ⋖ , ⋗ and ≐ between the symbols of a syntax. A language L is a precedence language if at most one relation holds between any pair of symbols and if no two productions of the syntax have identical right parts. A precedence recognizer uses these relations to decide whether to scan a new symbol or to do a reduction and, if so, which reduction. Practically, most programming languages are not precedence languages. Problems arise because very often more than one relation holds between two symbols. Identical right sides are also familiar in programming languages. These conflicts are due to the fact that a precedence parser uses very little context in making decisions (it only uses two adjacent symbols). In order to decrease the number of conflicts, the concept of precedence has been generalized to that of weak precedence.[4] There are only two weak precedence relations ≐ and ⋗; and the decision which reduction should be performed depends on pattern matching. The concept of weak precedence does not solve all problems, because most programming languages are not weak precedence languages.

Production language

The weak precedence relations can be derived automatically from a weak precedence syntax. Usually the relations are stored in matrix form or in some kind of list. A weak precedence parser using a matrix is expensive in core memory, if it uses a list it is expensive in time. Ichbiah and Morse have developed an algorithm for automatic translation from a weak precedence matrix to an "almost optimal" program[5,6] written in production language (PL). By this process, which is performed at metacompile-time, the information contained in the matrix could be stored in a very compact manner. The production language is a metalanguage for writing recognizers. A recognizer written in PL consists of a set of productions of the form

$$L1: S_3S_2S_1 \rightarrow S_2',S_1' \quad * \; L2 \quad (1).$$

The productions specify actions to be performed on symbols of a stack. Production (1) reads as follows: compare S_1, S_2 and S_3 with the top of the stack. If there is a match, the matched symbols S_1, S_2, S_3, in the stack are replaced by the symbols S_1', S_2', If they do not match the next production in the

PL-program is tried. "*" indicates that the next input symbol is scanned and pushed onto the stack. After successful execution of (1) the production labelled L2 is executed. In practice the PL-program is transformed into tables on which a standard parser acts.

The syntax part of a metacompiler

As already mentioned, most programming languages (e.g. ALGOL 60, CAMAC language) are not weak precedence languages because the weak precedence concept uses very little context.
The symbols and their relations are used without regard to the environment of the symbols. We have developed a method of changing a given syntax in order to solve these problems. An important feature of this method is that it does not change the structure of a syntax. The syntax part of our CWS is a program written in FORTRAN and IBM 360 Assembly. It accepts a syntax written in BNF, transforms it, constructs the weak precedence matrix and translates the matrix into a PL-program which is finally transformed into tables. The standard parser which acts on these tables is rather simple and can easily be written in some language convient for the available computer.

The syntax tables generated for the CAMAC language need 5K bytes (one pass compilation without segmentation). The tables are generated in 5 min on a SIEMENS 4004-55. The program has been successfully tested with other languages, e.g. ALGOL 60.

SEMANTICS

As mentioned, one of the components of a compiler-compiler is a semantic metalanguage which describes the meaning of the source language statements (e. g. CAMAC language).

Each PL-Statement in the syntax language may include a command which is a jump to a semantic statement. This semantic statement will be executed each time that production (PL statement) is matched.[7]

The actions to be performed by a semantic statement are the following:

- Checking the semantic correctness (e.g. whether an identifier is a CAMAC c-name or not)
- Generating code (In the case of the CAMAC language IML-statements (Macros) are generated)
- Changing the state of the translator (e.g. if a CAMAC equivalence statement is matched)
- Working on information in the symbol table.

The semantic part of the CWS is a program written in FORTRAN and IBM 360-Assembly. It accepts semantics written in the meta-language and translates them into FORTRAN routines.

This meta-language looks like a bastard FORTRAN. It has the facility to indicate at which syntactical symbol the the semantic action or code-generation should be performed. The access to all the informations contained in the symbol-table is described by the notation[8]:

POINTER.ATTRIBUTE

Where POINTER points to a symbol table entry and ATTRIBUTE is the name of a column in the symbol table.

```
Syntax rule:

<c-name-statement-body>::=<identifier>(<size>)=<address-set>G2(<mode>)

associated sematic actions:

action.<identifier>::=IF(<identifier>.DECLARATION .EQ. DECLARED) GOTO <<ERROR(N)>>
                    ::=<identifier>. DECLARATION = DECLARED
                    ::=<identifier>. INTERNAL =<c-name-identifier>. INTERNAL
action.<size>       ::=<identifier>. DIMENSION=<size>. VALUE
                     .
                     .
                     .
```

Fig. 2: Some semantic actions

In general, there may be some semantic attributes associated with a particular syntax element. To store the semantic attributes we use a semantic stack, which works in parallel with the syntax stack. Each element of the semantic stack points to a symbol table entry.

REFERENCES

1. Feldman, J., Gries, D., Translator Writing System. Comm. ACM, 11, (1968) p. 77
2. CAMAC Bulletin No. 5, Supplement, (1972)
3. Wirth, N., Weber, H., EULER - a generalisation of ALGOL, and its formal definition. Comm. ACM, 9 (1966), p. 13, p. 89
4. Ichbiah, J.D., Morse, S.P., A technique for generating almost optimal Floyd-Evans productions for precedence grammars. Comm. ACM, 13, 1970, 501
5. Floyd, R.W. A descriptive language for symbol manipulation. J.ACM, 8 (1961) p. 579
6. Evans, A. An ALGOL 60 compiler. Annual Rev. in Automatic Programming, 4 (1964) p. 87
7. Feldmann, J.A. A formal semantics for computer languages and its application in a compiler-compiler. Comm. ACM, 9 (1966) p. 3
8. Gries, D. Compiler Construction for digital Computers. (1971)

GENERAL DISCUSSION

Q - R. Hagelberg

We heard a number of contributions on CAMAC software. They have shown a lot of possibilities of variation depending on the basic language used as well as on the CAMAC - computer coupler involved. I have the feeling that it would help if one standardizes not only the computer languages but also the CAMAC - computer coupler. What is the status on such a CAMAC - Interface standardization?

A - I. Hooton

Standardization in CAMAC is at all times a matter of mutual agreement. The semantics of IML give the facilities which, in the opinion of the Software Working Group (SWG), are required by the general user. This therefore contributes an input to an agreement on controller facilities. The SWG would support moves towards standardization, for example by a recommendation on the general facilities from which selection could be made.

C - E. Rimmer

At CERN an attempt towards a standard interface has been made by reducing the computer-dependent portion of the interface to one or two plug-in cards.
P. Ponting should further comment on this.

C - P. Ponting

At CERN, to maximize standardization in interfaces while having several types of computers, the approach was to assume CAMAC to be the centre and the computers to be peripherals to the hardware. In this way common interface facilities can be extended much further towards the computer while retaining standard hardware.

C - P. Elzer

Concerning the correlations between CAMAC as a hardware standard and the forthcoming standards for real time software I would like to make a comment.
CAMAC is at this moment, and may well be for some time, "the" manufacturer independent hardware interface standard.
As V. Haase mentioned in his paper, there is a world wide effort towards a standard programming language for general real time applications. A crystallisation-point of this work was the "Purdue Workshop on Standardization of Industrial Computer Languages", but in the meantime it has grown into a really world wide effort (e.g. sponsored by IFIP). The most active committee of this workshop is the Long-Term Procedural Language Committee (LTPL-C). LTPL-C has three regional subgroups: LTPL-A, LTPL-J and LTPL-E. The most active one is LTPL-E (as in CAMAC). It has approximately 30 active members and 20 observers. To give an impression, at the last meeting in London there were representatives of GEC, RRE, BISRA, Bradford Univ. and ICI from U.K., CAP, CERCI and Univ. d'Orsay from France, BBC, SIEMENS, AEG-Telefunken, ESG, Erlangen Univ. and IDAS from Germany and the Hungarian Academy of Science. The work of LTPL-C is based on the principle of "synthesis of best features" and was started by a comparison of the "candidate languages". These are, at this moment in time: ALGOL 68, CORAL 66, PAS/I, PEARL, PL/I, PROCOL, and the CAMAC software. Agreed tasking and algorithmic features can be expected in 1974.
The question is how to find the optimal way of cooperation between the hardware standard and the forthcoming software standard.

C - I. Pyle

Let me summarize comparative details of software for CAMAC-computer applications in so far as the nine papers of session II are concerned ˣ), by the table below. The column codes of the table have the following meaning:

1 Paper number in session II
2 CAMAC facilities provided (in broad outline)
  a Symbolic device names
  b Single actions
  c Multiple actions (several CAMAC actions resulting from one statement; B = restriction to actions on a single CAMAC module, i.e., block-transfer)
  d Demand handling (interruption of program in response to LAM - any system with single actions can test LAM)
3 Environment for software described
  a Host language or non-CAMAC basis for software described
  b Style of incorporation of CAMAC features
    macro = additional instructions in-line
    subr = additional instructions as subroutines
    comp = extensions to a language requiring a special compiler
    meta = extensions to a language handler by use of a meta compiler
  c Computer concerned
4 Date of availability

Table

1	2				3			4
	a	b	c	d	a	b	c	
II-1	√	√	√	√	PAL 11	macro	PDP-11	Nov.73
	√	√	√	√	PAL 15	macro	PDP-15	Nov.73
	√	√	√	√	PAL 11	subr	PDP-11	May 72
	√	√	√	√	DOS	comp	PDP-11	Aug.73
	√	√	-	-	MACRO 10	macro	PDP-10	Apr. 72
	√	√	-	-	ALGOL 60	subr	PDP-10	Oct.73
	√	√	-	-	FORTRAN	subr	PDP-10	Oct.73
II-2	√	√	B	-	FORTRAN	subr	PDP-11	Sept.72
II-3	√	√	B	-	BASIC	subr	HP2100	Jan.72
II-4	√	√	√	√	META II	meta	IBM360	May 74
II-5	√	√	-	√	PL-11	comp	PDP-11	Oct.73
II-6					POLYP	meta	IBM360 Siemens 4004 PDP-11 TR86	Jan. 73
II-7	√	√		√	PEARL	comp		74/75
II-8	√	√	√	√	BASIC	comp	PDP-11	Dec.73
II-9	√	√	√	√	FORTRAN	meta	IBM360 Siemens 4004	

ˣ) many other implementations have also been made

III

AUTOMATION IN LABORATORIEN

AUTOMATION OF LABORATORY INSTRUMENTATION

AUTOMATISATION DE L'INSTRUMENTATION DE LABORATOIRE

Präsident - Chairman - Président:

P. GALLICE, CEA Saclay, France

AUTOMATION OF LABORATORY INSTRUMENTS - A REVIEW

J. Farren, R. K. Webster
Applied Chemistry Division, A. E. R. E., Harwell, England

ABSTRACT

The range of instruments used in chemical, clinical and solid state laboratories is reviewed, and their interfacing requirements grouped under high speed input, low speed analogue and digital input, and control functions. An outline is given of the current status and trends for various automation schemes.

ZUSAMMENFASSUNG

Es wird eine Übersicht über die in chemischen, klinischen und festkörperphysikalischen Laboratorien benutzten Instrumente gegeben und die Interface-Erfordernisse werden jeweils für schnelle Eingabe, für langsame Analog- und Digital-Eingabe sowie für die Steuerfunktionen behandelt. Es folgt ein Überblick über den derzeitigen Stand und die verschiedenen Tendenzen auf dem Gebiet der Labor-Automation.

RESUME

On passe en revue les classes d'instruments utilisés dans les laboratoires de chimie, de physique du solide, et les laboratoires médicaux. Les interfaces requises par ces instruments sont classées en : entrée à grande vitesse, entrées analogique et digitale à faible vitesse, fonctions de commande. On donne un aperçu de la situation actuelle et des tendances de divers projets d'automatisation.

The trends in laboratories towards faster and more complex instrumentation, and a growing emphasis on results and applications, have set a high demand for automation. This is needed to extract the maximum information from instruments with improved speed, accuracy and reproducibility, and to permit measurements which are not feasible by the slower manual methods; to increase sample throughput, and to permit better quality control procedures; and to release the time of qualified staff to consider application problems rather than difficulties concerned with the operation of complex equipment. The automation of instrumentation has therefore become very diverse and sophisticated in recent years, especially with the advent of new interfaces, hardware, and extensive applications and system software. This paper provides a brief review of laboratory instrumentation, lists some basic system parameters, classifies instruments into a few basic groups based on input/output requirements, outlines some interfacing approaches now in use or under development, and summarises general computer requirements.

Laboratory instrumentation

Direct multi-channel recording of nuclear pulses, for example for gamma-spectrometry and its various applications in activation analysis, and multi-scaling options as in Mössbauer spectrometry, are well-known to most CAMAC users and are not reviewed here. It was the direct availability of output data in digital form which expedited such applications, but most instrumentation provides a time-varying analogue signal, conventionally as a chart recorder output.

In nuclear magnetic resonance, changes in an rf signal level, corresponding to the absorption of energy during the transitions of protons and other magnetic nuclei between different energy states, are recorded while varying an applied magnetic field - different proton absorption lines correspond to different chemical states in the sample, and so permit investigation of chemical structure. Signal-noise ratios are inevitably limited, and sensitivity is enhanced by digitising the signals for multi-scaling: repetitive addition then provides a statistical improvement in sensitivity. Subsequent data handling may involve digital smoothing or resolution enhancing procedures, background subtraction, and possibly comparison with a theoretical spectrum computed by simulation. Instrumentally, the homogeneity of the magnetic field must be controlled to within about $1/10^8$. Higher sensitivity is achieved in pulsed NMR, but at the cost of higher data rates. Short (1-10 μs) high power pulses are applied at a fixed magnetic field, and the free induction decay signal (up to 32k data points) recorded using signal averaging techniques. Fourier transform of the decay signal then provides the normal absorption spectrum, but with greatly enhanced sensitivity (by a factor of 10-20). The technique is particularly valuable for "weak" nuclei such as carbon-13 and phosphorus-31.

Similar signal averaging methods are also applied in electron spin resonance. Automation permits rapid signal integration, and correction for the base-line slope which is introduced by the magnet field modulation used for the phase-sensitive detection system. The variation of transition metal spectra for crystals with orientation in the magnetic field can be studied more efficiently with automatic control over the angular position of a rotating electro-magnet.

For infra-red and Raman spectroscopy, digital accumulation again provides an improved signal-noise ratio, and also permits background subtraction, data normalisation, and conversion of transmission data to optical densities. Fourier transform techniques are applied to infra-red interferometers. As with many of the techniques discussed, digital storage of infra-red spectra suits archiving for subsequent retrieval and matching with standard library spectra.

High resolution mass spectrometry is a well established technique for organic structural studies. However, spectra must be recorded at high speed over a wide mass range at high resolution. Following digital recording, the data points are processed to provide the "arrival time", corresponding to the centroid of each mass peak, and its intensity. Exact masses are then calculated from these arrival times, spectra smoothed, backgrounds subtracted as necessary, and an elemental formula calculated to correspond to each mass.

A gas chromatograph separates the components in a mixture by partition between a flowing vapour phase and a suitable stationary phase - retention times are governed by absorption equilibria, and the GLC output signal can therefore be used for the quantitative analysis of mixtures with known components. Over 100,000 gas chromatographs are in use in laboratories, and for quality control in chemical, petroleum and pharmaceutical plants. Automatic data reduction can cover calculation of peak areas with corrections for base-line drift and peak overlap, and normalisation of retention times when using internal standards.

The combination of gas chromatography with mass spectrometry provides a powerful tool where the mass spectrum of each component is recorded as it is eluted from the chromatograph column. However, a chromatograph peak may last for only 1-10 seconds at an intensity suitable for the mass spectrometric analysis. The requirement for fast mass analysis of many peaks from the chromatograph therefore sets a two dimensional automation problem.

Automation in clinical laboratories is typically aimed at high throughputs with "simple" rather than complex instruments, for example in using a range of automatic chemical analysers for the determination of various blood components, typically by the addition of reagents followed by spectrophotometry. A recent development is the "fast analyser" which is effectively a spectrophotometer which can examine about 16 samples or standards in cuvettes mounted on the periphery of a rotor. At 500 rpm samples and standards are examined at frequent intervals to provide high speed chemical analyses, e.g. for glucose, and particularly for enzymes where results are obtained from the rate of change of absorbance. Automation involves data recording for all parallel channels, conversion of transmissions to absorbances, etc. Liquid chromatography is a fast developing technique; with a u.v. detector a single analysis can provide a large number of separated components, e.g. over 140 from a urine sample.

Automation problems are equally prevalent in the inorganic/materials areas. As with spectrophotometry, atomic absorption and flame photometric instruments suit automation for long sample runs, and so require simple recording of transmission, signal processing, and particularly control over the changing of sample solutions. Spark source mass spectrometry can measure nearly all elemental components for survey analyses. The automation problems are as appropriate to low speed mass spectrometry, but elemental concentrations can cover an unusually wide range, from ppb to the % level. Data analysis is similar to that noted for high resolution organic mass spectrometry, but requires corrections for interference from multiply charged ions, etc. The data for both mass and optical spectrographs can be recorded on photographic plates and then analysed automatically with digitally controlled microdensitometers. For quantitative analysis data processing involves construction of the photographic emulsion response curves.

The various X-ray techniques set control as well as data acquisition problems. Thus an X-ray fluorescence analyser will require positioning of the detector to the correct angle of reflection for each element, as well as sample changing. Data reduction may require corrections for inter-element absorption and enhancement effects. The electron microprobe analyser extends the process in also requiring control over movements of the specimen stage to allow information to be obtained over a relatively large area. Diffractometers require control over the crystal orientation. Similar fields of automation include the control of equipment for measuring powder surface areas and recording absorption/desorption isotherms, and the automation of creep strain equipment in tensile testing laboratories.

"Laboratory instrumentation" thus covers many instruments, and extends over analytical, inorganic and organic chemistry, biochemistry, clinical, metallurgy and engineering laboratories. The instruments have differing requirements, but these can be grouped broadly into: data acquisition; data reduction; instrument control; and the preparation of reports. The extent to which these are covered will depend upon the automation scheme: special-purpose electronic units will probably cover only the data acquisition phase, but at the other extreme a time-shared computer scheme could well support sufficient peripherals such as a line printer and a plotter to cover quite extensive report preparation. The remainder of this paper concentrates primarily on the data handling and control functions, as these are the more appropriate areas for considering the use of CAMAC units.

Basic system parameters and interfacing requirements

In designing automation systems various factors must be taken into account. These include: (a) the signal dynamic range, (b) the sample rate, (c) resolution, (d) the amount of data produced during a typical experiment, (e) any essential real-time data processing, and (f) instrument control. Almost all of these factors are interdependent. In the case of a dedicated computer/instrument system, the operator is often limited by the hardware interface, except in cases where complex real-time decisions have to be taken. Another dimension is added to the problem if several instruments are simultaneously controlled by a single computer. In this case the real-time control of experiments and the simultaneous processing of acquired data in a background mode require careful analysis of instrumental demands and can often lead to the imposition of some operational limitations, for example on the total acceptable data rates, or the system response to 'master' instruments which interrupt the computer.

The historical development of CAMAC placed considerable emphasis on its use in digital input/output environments, for example in fields such as nuclear physics, communications and accelerator control. However, many laboratory instruments do not produce the digital data generated by gamma-cameras, multi-channel analysers etc. but depend on the computer accepting analogue input signals, and being capable of generating analogue output control signals. Until the last few years the CAMAC analogue field was seriously deficient compared with the state of development of digital units, but more and more sophisticated, high speed and high resolution modules are now being produced.

Table I shows a selected list of laboratory instruments, and indicates the type of input and output functions required by each. The data rates divide conveniently into a high speed region (above about 100 Hz) and a low speed region for which 100 Hz and below is adequate. Digital and analogue control is required by many instruments for stopping and starting data flow, spectrum scanning, multiplexing and for amplifier range changing. In general, instrument control is rarely carried out at rates greater than 100 Hz especially where the decisions are based on the incoming data. The requirements of several of these instruments are discussed in order to demonstrate the way in which automation of analogue signal instruments is progressing.

The high resolution mass spectrometer and pulsed nuclear magnetic resonance spectrometer are typical of the high data rate instruments. Sample rates of upwards of 50 kHz are required. This can be achieved only by sample and hold, analogue to digital converters, often with less than 14 bits resolution. Data may be stored directly by using direct memory access (pulsed NMR), or subjected to simple peak detection software in real time (HRMS). This latter data processing is necessary to reduce the very large quantities of data (about 10^6 words) which are generated in a typical experiment.

At the other extreme of the data acquisition scale lies gas-liquid chromatography and a wide range of quite slow analytical and testing instruments. Whereas the data rate requirement is less onerous, the dynamic range of the signal may be very high. Typical modern GLCs have amplifiers capable of handling signals over a range of 10^6 or more. Two methods of digitising signals of this type are currently used. Firstly, a conventional sample and hold ADC is coupled to some form of variable gain amplifier system. The gain may be changed by multiplexing one of several

Input/output requirements of various laboratory instruments

Instrument or Technique	High Speed Input	Low Speed Analogue Input	Low Speed Digital Input	Digital Control
Mass Spectrometry (High Resolution Organic)	++	++*		
Mass Spectrometry (Low Resolution with GLC)	++	++*		++
Mass Spectrometry (Inorganic)		++	++	++
Nuclear Magnetic Resonance (Pulsed or Continuous)	++	++		++
Gas Chromatography		++*		++
Fourier Transform I.R. Interferometers		++		++
Microwave Spectrometry		++		++
Raman Spectrometry		++		
Mössbauer	++			
X-ray Diffraction (single crystal or powder)			++	++
IR, UV and visible Spectrophotometers		++		++
Emission Spectrophotometers		++		
Atomic Absorption		++		
Spectrofluorimeters		++		
Nuclear Measurements and Multichannel analysis			++	
Activation Analysis and Equivalent Methods			++	
X-ray Fluorescence			++	++
Electron and Ion Microprobes			++	++
Electron Spin Resonance		++		++
Photographic Plate Readers		++	++	
Auger Spectrometry			++	++
=Automated Chemistry (Path.Lab. etc.)		++		
=Spectropolarimetry		++	++	
=Analytical Centrifuge	++	++		
=Polarography		++		
=Ion Specific Electrodes etc.		++		
=Liquid Chromatography		++		++
=Stop-flow Kinetics	++			
=Tensile-Testing		++	++	++
=Ultrasonics etc.		++	++	

*Alternative system may include a hardwired pre-processor
=Instruments which may not individually justify a computer, but could usefully be automated if one is available in the laboratory

Table I

fixed gain amplifiers; the decision on the gain is determined by a hardware or software comparison of the last data point with preset reference levels. Secondly, for sample rates of about 1 Hz, voltage or current to frequency converters combined with suitable counters can be used. The sample rate can be increased without loss of dynamic range if an amplifier range changer is added to this system.

The choice of which system to use depends on several factors. The ADC system is more economical where many GLCs are being used simultaneously but is expensive for a limited number. It also suffers from a lower signal-noise ratio resulting from using the sample and hold method. For optimum performance, the VFC system requires a separate converter at each instrument and is therefore costly for many instruments. Also, in some cases the linearity of VFCs can be a problem where quantitative results are required over a wide dynamic range. However, the integrating property of these converters provides a good signal-noise ratio.

Between the two extremes of data rate lie a range of instruments operating at sample rates of about 0.5-5 kHz: low resolution mass spectrometers, continuous wave nuclear magnetic resonance etc. Instrumental design improvements in this area have gradually introduced another feature of laboratory automation - the need for high precision and stable analogue control. Magnetic field and radio frequency scanning can be controlled by the computer, and the length of the scan step shift varied according to decisions taken within the processor. Rapid 14-16 bit DACs can generate mass spectra by switching the spectrometer from one mass to the next, and so avoid wasting time in scanning the whole spectrum. For NMR, spectrum enhancing techniques can be optimised by using a computer driven DAC to scan the spectrum and identify a suitable standard peak. The spectrum being enhanced is then scanned using the location of the standard as a reference offset. In this way instrumental drift is corrected as each spectrum is recorded. Techniques of this kind have become possible only because of the development of wide range, stable DACs. Previous systems relied on using electro-mechanical start and stop reference points, or uncontrolled auto-scanning of the instrument. The former always presents problems of imprecision in measurement of length or time, and the latter requires either internal or external calibration of the data. In both cases a considerable part of the data collected may be redundant.

A development which is steadily increasing in importance in instrument automation is the use of data compressing or preprocessing interfaces. In many cases it is not possible to compress data before it is processed by a computer. Thus the example of pulsed NMR has already been noted where a large volume of raw high speed data (up to 32K words) must be stored; direct memory access provides a convenient method of acquiring this type of data and involves a minimum processor overhead. However, in fast scanning mass spectrometry and in GLC, interfaces are now available which can reduce the incoming raw data to the significant information such as peak centroid time, area or peak height. By using a relatively small memory for buffering this reduced data, it has been possible to output at teletype speeds the significant information from high and low resolution mass spectrometers which normally acquire raw data at 20-50 kHz. A peak-picking interface of this type, designed by Dr. Carrick, has considerable potential in environments where a dedicated computer is not available and where off-line processing of paper tape is convenient. This type of interface has also considerable potential in time-shared computer systems where processor time is at a premium.

For GLC, integrators are available which resolve the raw data into peak areas and retention times. Modern units can perform quite complex analysis of overlapping peaks using a variety of area reallocation algorithms. As with the peak-picking mass spectrometer interface, the load on an associated processor is kept to a minimum when chromatographs are interfaced through such an integrator.

Laboratory-instrument computer systems

The essential requirements for a general laboratory instrument/computer system can now be summarised. The requirements may be relaxed for some dedicated systems, where instrument complexity may justify devoting a small computer to a single instrument or where the real-time demands made on the computer are incompatible with sharing its processor capacity with other instruments.

For hardware, an input channel which satisfies high data rate instruments (5 - 100 kHz) is required; for rates at the lower end of this range it may be advantageous to have a multiplexed input to allow a small number of fast instruments to input data at the same time. A multiplexed low speed input interface is needed which accepts data in either digital or analogue form at rates up to about 100 Hz from many laboratory instruments. Finally, hardware and software are required to allow the operator to introduce a variety of instrument control methods.

For comprehensive software the requirements can be itemised as follows: (i) A multi-programming, multi-user executive which is capable of driving the instrument-computer interfaces and the more common peripherals (disk, visual display unit, line printer and graph plotter). It should support a high level language and be able to compile, link and run programs which have been written and debugged on-line in that language, or in assembler. (ii) An operating system which represents a convenient interface between the computer and the operators, and allows them to set up data acquisition and subsequent data processing tasks without any detailed knowledge of the computer-instrument interface, or the interface scheduling or program scheduling sections which form part of the executive/ operating system. It should support simultaneous operation of the instruments, data processing, and program development. (iii) A series of general software data processing routines, e.g. plot, smooth, convolute, integrate, differentiate. (iv) A range of instrument data handling programs for specific user requirements.

Conclusions

It is difficult to be certain in identifying general trends in instrument automation and interfacing. A variety of interfacing changes are taking place. Instruments are continuously becoming more sophisticated and require more advanced interfaces to make full use of them. Electronic components are also becoming more elaborate and able to perform quite complex data reduction and instrument control functions with a minimum of computer intervention. At the same time processing power and memory units are getting cheaper so that micro or mini computers which act as intelligent interfaces may take the place of data reducing hardware. If there is a general trend it is towards the incorporation of very small computers in the larger, expensive instruments. The computers will provide a minimum of processing power and instrument control but will reduce the load on any central midi computer to which many such instrument systems are connected. For simpler, cheaper instruments the trend will probably be towards a standardised computer oriented input/output format which can be interfaced through flexible hardware to a single computer. Under these conditions the current approach involving the use of several free-standing computers, each dedicated to a single instrument, might be replaced by a trend towards the use of a single large computer for laboratory rather than instrument automation.

DISCUSSION

Q - P. Groll

Sie beschrieben in Ihrem Beitrag chemisch-physikalische Messeinrichtungen für Gas-chromatographie, Massenspektrometrie, Elektronenspinresonanzmessungen u. s. w. Diese Geräte sind heute bereits von vielen Herstellern auf Wunsch mit einem Rechner ausgestattet. Wo liegen die Vorteile für den Anwender, wenn er CAMAC verwendet, einmal in ökonomischer Hinsicht und zum anderen in Bezug auf mögliche Zeitersparnisse?

A - R. Webster

This is a very broad question which ranges over the various types of computer systems as well as over CAMAC. Your question refers to instruments for which computer systems are already available from instrument manufacturers. I think that if such a system exactly matches all requirements in your laboratory, both now and with future development, then the easiest solution is to buy it. However, there are many cases where alternative approaches are worth considering. For example the instrument company's computer system will often service only one instrument or one type of instrument, and this may cause difficulties in a laboratory which wants to automate several different instrument types supplied normally by different manufacturers. There is then a gap between the different instruments which is well-bridged by CAMAC when developing an overall system. On the two specific points:

- On economics, CAMAC can save user costs in developing and implementing a system. On overall costs a time-shared system may cost no more than say two or three dedicated systems and may then provide more powerful peripheral equipment for sharing between the various instruments. The user can then also avoid the need to become familiar with two or more computers and languages.
- On time it is obviously quicker to buy an off-the-self unit, and many users underestimate the time needed to develop and commission a software system. However, if a user is going to implement his own system, the use of CAMAC should certainly minimize the time required.

THE CAMAC SYSTEM AT THE PHYSICS DEPARTMENT OF THE TECHNICAL UNIVERSITY VIENNA

D. Hammer
2. Institut für Experimentalphysik, Technische Hochschule Wien,
Wien, Austria

ABSTRACT

The actual CAMAC system and its typical applications are described. Particular attention is given to the discussion of the different decisions that preceded the specification of the final system and which are fairly general with respect to laboratory automation. The problems associated with CAMAC software are summarized.

ZUSAMMENFASSUNG

Ein zur Verfügung stehendes CAMAC-System und seine typischen Anwendungen werden beschrieben. Insbesondere werden die verschiedenen Entscheidungen erörtert, die der Festlegung des endgültigen Systems vorausgegangen sind und die in bezug auf eine Labor-Automation weitgehend allgemeingültig sind. Die mit der CAMAC-Software verbundenen Probleme werden zusammenfassend dargestellt.

RESUME

Description du système CAMAC utilisé actuellement et de ses applications les plus caractéristiques. L'accent est mis sur la discussion de diverses décisions ayant précédé la spécification du système définitif et présentant un intérêt général pour l'automatisation en laboratoire. Résumé des problèmes associés à la programmation du CAMAC.

1. Introduction

Within the physics department of our university there are four institutes working on a rather wide spread field of general experimental physics. The nuclear physics department is excluded because it is located on a different site and has its own computing and CAMAC facilities.

The main working fields are solid state physics, cryo physics, material science and surface physics including ion beam studies. There are many different experiments mainly concerned with various types of spectroscopy like X-ray, Mössbauer, nuclear magnetic resonance, photon, LEED, Auger, photoelectron and mass spectroscopy.

Three of these four institutes joined in a program for increasing the efficiency of their experiments by automating them with the aid of a computer. This was done mainly for the following reasons:

- On line data acquisition and evaluation.
- 24 hour operation of long time experiments.
- Automatic supervision and control of experiments
- Partial replacement of conventional measurement hardware by software.

This also gives a rough idea about the line we will proceed, working now on the first stage of this program.

2. How can one connect an experiment to a computer and why did we choose a CAMAC system?

By discussing this we had to start from the following facts:

- There are many different users respectively experiments which want to work independently.
- The system has to be very flexible because an experiment usually is in a state of continuous change.
- The system has to be divided into two parts. One "process" part which does the data acquisition, data reduction and the control of the experiment. One "batch" part which does the thorough data analysis with respect to physics. The latter job has to be done on a large system and is not considered here.
- The experiments are generally very slow with data rates ranging from about 10 to 10^{-2} in- and outputs per second.

There are, of course, also fast experiments like Nuclear Spectroscopy, Pulse Magnetometers and so on. But these experiments have to be buffered by an external processor, transient recorder or memory anyway. Then the data can be read from this buffer by DMA (Direct Memory Access) transfer.

The first solution we discussed was the connection of the experiments to the analog and digital interface of an already existing IBM 1800 processor. Such a system is established within the electric engineering department of our university.

The implications would have been hundreds of up to 1 km long transmission cables from the experiments to the remote interface. Since it is very hard to transmit small analog signals over these distances this idea immediately leads to a second ADC or amplifier-buffer interface at the experiment itself. Thus this clumsy, slow and expensive solution was rejected.

The second idea that arose was to have a network of small processors dedicated to the different experiments. The single processor would have only 4 to 8k words of memory, very simple peripherials like teletypes and the interface to the experiment. A larger computer is then used to supervise all the small processors which do the data acquisition, data reduction and control of the experiments.

In discussing the interface problem it became obvious that some type of standardization is almost mandatory. Otherwise we would have to face very disappointing implications like:

- Dozens of different interfaces which are tailored to the experiments in a rather unflexible manner.
- Dozens of physicists doing just interfacing instead of physics.
- Neither the hard nor the software is sharable. Thus the distribution of knowledge becomes very difficult.

It was at this time when we heard about the CAMAC (EUR 4100) system. We suddenly realized that this is not only a very fast and flexible computer interface but also a very cost effective one as compared to the many man hours of work necessary for the above mentioned solution.

Although a network of small processors connected to a host computer is probably the only solution for a fast system with high data rates it has several disadvantages for our special purpose:

- The system is very expensive because of the many computers and interprocessor links that are involved.
- The utilization of the small processor is rather bad for most of the experiments.
- If one connects more than one experiment to a processor a lot of organization software has to be written. This software probably has to be changed every time an experiment changes.

At this point of decision the CAMAC multicrate system (EUR 4600) gave us the possibility to connect many experiments respectively crates in a standardized and flexible manner together and to a central processor.

Also other solutions, as suggested by EUR 4600, have been considered. For instance the extension of a computer bus to the different crates using special crate controllers. Although this system has several advantages with respect to direct module addressing interrupt and data handling, it is rather qualified for smaller systems. For many crates one needs too many peripherial adresses and the long computer bus is very susceptible to any perturbations which are then disabling the whole system.Also if another computer should be used in the future it is for more than about three crates per branch cheaper to change the branch interface than to change all special crate controllers.

Thus a rather large common computing facility was chosen and the different laboratories were connected by CAMAC Branch Highways. This, of course, puts severe requirements on the operating system. It now becomes a multi user system that has to perform real time operations while giving full protection to the individual user.

3. How does our CAMAC system actually look like?

In order to have a uniform responsibility the whole system consisting of the CAMAC hardware, a Digital Equipment Corporation PDP-11/45 with a RSX-11D Real Time Executive System and the associated CAMAC software has been ordered from one company (Schlumberger Instruments and Systems).

At the moment, according to the three institutes involved, we have three CAMAC Branch Highways. Provisions are taken to allow easy extension to more branches.

These three Branch Highways extend to up to 250 m from the central processor since the different laboratories are far apart. This seemed to be a problem at the beginning of the project because the CAMAC Branch Highway itself can be only extended to about 30 m. Meanwhile differential branch drivers/receivers for distances up to 350 m have been developed. Nevertheless, when the Serial Branch Highway becomes available this may be a cheaper solution for low data rate, long distance applications as we have.

The efficicncy of a CAMAC system is to a large extent determined by the hard- and software interface to the computer. As far as the hardware is concerned the Schlumberger ICP-11 CAMAC PDP-11 interface has the following main features:

- Automatic Graded LAM processing. If a Branch Demand arrives the hardware automatically reads the graded LAM Pattern of the Branch and jumps to the service routine of the GL with the highest priority.
- The module address N and subaddress A is automatically decoded from the computer bus address lines. Thus for all CAMAC operations within the same Branch a maximum of two computer instructions is necessary. One to load the Crate Number C and the Function Code F and one to send or receive the data with the appropriate address (N.A).
- For non read/write operations like Test LAM the hardware provides the possibility to send Q as data bit 15 (Sign) immediately to the computer. This allows for easy and fast branching on the result of the operation.
- DMA transfers of data blocks up to 256 words are possible both in the single address and in the address scan mode. Thereby one has the choice to send a request to the computer either for each block of words or for every single word transferred. The latter operation mode allows other DMA transfers like disk operations to interfere with the CAMAC DMA transfer.

Unfortunately, the software interface between the operating system of the computer and the CAMAC equipment is not yet operating although it is part of the over all system. This is mainly due to the delivery delays and the complexity of the RSX-11D executive.

The central processor, as already mentioned is a DEC PDP-11/45 with 56k words of core memory, two disks, two small tape drives, a high speed papertape reader/punch and a floating point processor. The memory segmentation unit provides the necessary address space extension from 16 to 18 bit and the hardware facilities for the user protection by the RSX-11D Real Time Executive system.

This processor is mainly destined for process work and data compression whereas the thorough data analysis is done on a large time sharing system. A 2400 Baud asynchronous serial data link provides the necessary connection via modems.

For program development the foreground/background capabilities of the RSX-11D executive are used. Programming is performed on an interactive video display terminal.

Beside the operator console which is a DEC Writer at the time each branch has one associated remote teletype for control of the experiments and message output. More Teletypes can easily be connected either to the computer itself or to the CAMAC system by Teletype driver modules.

Diagrams and associated text are plotted individually for every experiment on slow X-Y plotters driven by CAMAC modules which include also the necessary pattern and vector generators.

4. Summary

Although the hardware is working properly the most difficulties we are facing now are arising from software. Beside the software interface to the CAMAC system we are mainly concerned about the CAMAC software.

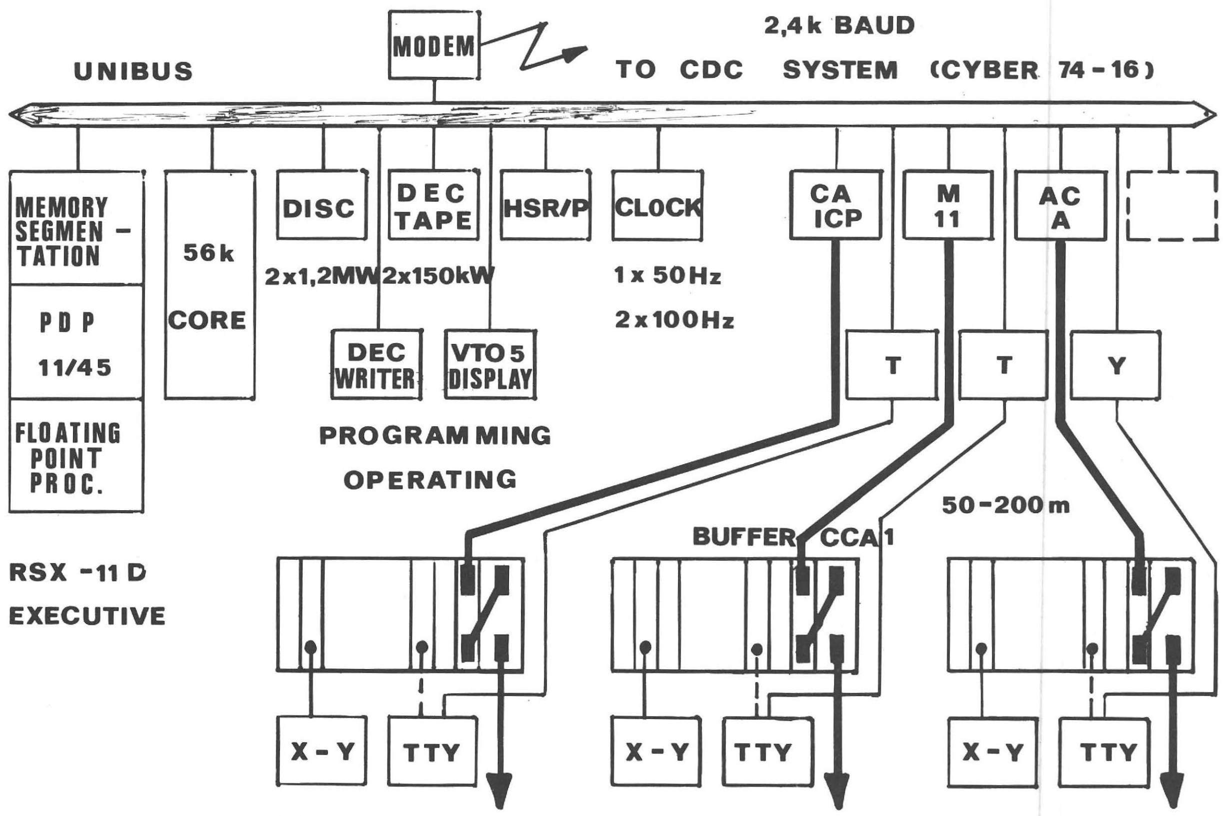
Many colleagues would like to utilize the advantages of the CAMAC system and computerize their experiments. As experimental physicists they usually have no problems with the CAMAC hardware. But they are afraid of learning assembler language, operating system directives and CAMAC command structure. Therefore, the development of CAMAC software becomes a crucial point not only for us but also for the wide spread acceptance of the CAMAC system.

Up to the time where CAMAC and IML language become available we intend to use two development stages. The first stage comprises the use of Macro Assembler statements for assigning symbolic names to CAMAC modules, initializing the CAMAC system, and performing CAMAC operations.

The second stage most probably will comprise the use of FORTRAN subroutine calls for these purposes. This will be much more convenient since almost all related persons are familiar with the FORTRAN language.

Acknowledgment

The author would like to thank Dipl.Ing.G.Janzek, Dr.H.Paul, Dipl.Ing.J.Pammer, Dr.T.Tortschanoff, and all his colleagues for their collaboration in this project and their invaluable help.



CAMAC SYSTEM Physics Departement TH VIENNA

DISCUSSION

Q - C. Stephens

We have similar slow data rates in astronomical problems and have found that interactive language extensions are easy to implement and are easily used by physicists in an equivalent situation. How many man-years of software development will be involved in implementing CAMAC operations and CAMAC drivers for teletypes and xy-plotters?

A - D. Hammer

As far as the driver for the CAMAC Branch Interface is concerned our experience is that about one man-year of software work is necessary to implement it. This figure also compromises the development of the appropriate system MACROS to work on the CAMAC system.
All Teletypes are connected directly to the processor bus during the first stage of the project. For this case the RSX-11D executive system provides the necessary multi-teletype driver. Later on, additional terminals might be connected via the CAMAC system. The necessary software can then to a large extent rely on the CAMAC driver and teletype driver.
The plotters will essentially be treated like any other instrument connected to the CAMAC system, using the CAMAC driver. Here the problem is to provide in the system library the utility routines needed by the users. The software work that is necessary depends on the sophistication of these utilities, and the extent to which existing programs can be used. Therefore, no exact figures can be given.

THE CERN SPS-MULTICOMPUTER AND CAMAC REAL-TIME CONTROL SYSTEM

M. C. Crowley-Milling, J. Alataber, M. Collins, T. Hymann,
R. Rausch, G. Schering, R. Wilhelm
CERN, Genève, Switzerland

ABSTRACT

A new concept of CAMAC, allowing critical real-time operation and using a modular CAMAC multicontroller approach, will be applied to the SPS multicomputer control system. Programmed I/O operation, interleaved DMA, fast interrupt logic and autonomous operation in the same CAMAC crate will be used. A typical application of this concept to a high speed high precision analogue data acquisition system is given. CAMAC will be used to interface general purpose multiplexers which drive very large quantities of accelerator control equipment.

ZUSAMMENFASSUNG

Eine neue CAMAC-Konzeption, die kritischen Echtzeitbetrieb gestattet und mit einer CAMAC Mehrfachsteuerung in modularer Bauweise arbeitet, ist für das aus mehreren Digitalrechnern bestehende SPS Steuersystem vorgesehen. Für programmierten Ein- und Ausgabe-Betrieb, für den überlappten direkten Speicherzugriff, für schnelle Unterbrechungsverfahren und für autonomen Betrieb wird ein und derselbe CAMAC-Überrahmen benutzt. Eine typische Anwendung dieser Konzeption für ein schnelles Datenerfassungssystem hoher Präzision wird beschrieben. CAMAC dient zum Anschluss von universellen Multiplexern, die eine grosse Anzahl von Beschleuniger-Steuergeräten betreiben.

RESUME

Une nouvelle conception du CAMAC, permettant le fonctionnement en temps réel et utilisant une approche multi-contrôleurs modulaires, sera employée pour le système de contrôle à 25 ordinateurs du SPS. Les opérations e/s programmées, l'accès direct multiple à la mémoire, les interruptions rapides ainsi que le fonctionnement autonome sont réalisés dans un même châssis CAMAC. Une application type de cette conception est décrite pour un système d'acquisition de données par multiplexage analogique rapide et à haute précision. CAMAC est employé pour interfacer des multiplexeurs d'usage général dont le but est de commander de très larges quantités d'équipements de contrôle de l'accélérateur.

INTRODUCTION

The SPS is a large new accelerator under construction by the European Organization for Nuclear Research (CERN) near Geneva. The accelerator consists essentially of a 2,2 km diameter ring around which protons are guided by some 1000 magnets while being accelerated by radio-frequency electric fields. Protons will be injected from the existing synchrotron at 10 GeV, accelerated at 400 GeV, then ejected for use in physics experiments. This process is cyclic with a repetition time from about 4 to 12 seconds. Although most of the accelerator equipment is contained in subterranean tunnels the power supplies control equipment and computers are housed in 6 separate equipment buildings on the surface, distributed round the ring.

This paper is concerned with the control system for the 400 GeV SPS Accelerator and mainly describes the CAMAC to computer interface concept for the real time control of the machine. The first part deals with some of the reasons for the choice of a multiple mini-computer configuration instead of a big powerful computer and points out the particularities of the equipment layout. The second part presents the CAMAC crate interface and its ability of autonomous operation fully interleaved with the computer input/output in order to satisfy the real time requirements.

1. THE SPS COMPUTER SYSTEM

Because of the depth of the ring tunnel, access for cables, etc., will only be possible at six points, where the equipment buildings will be. Most of the control requirements arise in these buildings, but there will also be apparatus requiring supervision and control at the Substation, at the Central Heating Building, at the Pumping Station and in the two Experimental Areas. The Main Control Building will be near the Auxiliary Equipment Building No. 3 Fig. 1.

The distance between the Main Control Building and most of the Auxiliary Buildings is considerably more than a kilometre. Thus it is essential to process and concentrate some of the data in the Auxiliary Buildings. This means that it is necessary to integrate computers into the system from the start, and it will have to be accepted that over-all control from the central control room will be impossible without the computer system.

Nowadays the mini-computer can provide considerable computing power at low cost, this makes it reasonable to use a number of separate satellite computers, each carrying out a few simple tasks with comparatively simple programs and thereby reducing the load on the central computer.

For the SPS this decentralization will be taken to its logical conclusion. Most of the simple operations, especially the real time tasks which are linked to the machine cycle, will be carried out in the satellite computers. Then most of the real time operations that the central computer system has to handle will be in the time scale of the machine operator. Some parts of the system are being designed to allow "stand alone" operation, for testing and maintenance, under the control of the satellite computer (ref. 1).

There are parts of the SPS such as the radio-frequency and beam-extraction system, which are geographically discrete and can be controlled by a single computer each, but there are others, such as the vacuum system, spread out over more than 7 km of tunnels, where it is more convenient to divide the system into sectors, determined by the number of access points into the ring. There will be a small computer at each of the Auxiliary Buildings, to deal with the things that are common to all sectors, such as vacuum, beam monitoring, power supply, water cooling, etc., and others where there is a local concentration of a system, such as beam transfer and injection at building No. 1, extraction and beam line at No. 2, radio-frequency at No. 3 , extraction and beam line at No. 6.

All these computers will be connected to the central control area by means of individual high speed data links. For the central control considerable advantages result from using a number of separate, smaller computers with a suitable software system. Each central computer will perform a few simple tasks, and does not need a complicated operating system. The system can be built up progressively, and the operation or modification of one part does not react directly on another part. The same type of computer will be used both for the various parts of the central complex and for the satellites, with obvious savings in hardware and software.

The whole layout is shown in Fig. 2. The heart of the system is the Message Transfer System. This acts as a telephone switchboard, routing messages from any one computer to any other, according to message header, but otherwise appears transparent (ref. 2). The computers shown as blocks below the message switching device are connected to it through data links varying in length from a hundred metres to nearly four kilometres. These will use serial links with a capacity of 30,000 words/sec.

The computers shown as blocks above the message switching device will all be in the central control area. There will be three main control consoles in the control room, and each serviced by a separate computer. The console computers will form the main interface between the operators and the system.

In addition to the console computers, there will be a number of special purpose computers. The first of these is the display computer. Each console will have its own simple alphanumeric display, but more sophisticated displays, including graphics and colours, will be provided for the whole system by this special computer, which will convert the display information into TV raster form for display on TV monitors.

All alarm messages will be routed to another of the special purpose computers, where they will be analysed to find out what is the cause and the effect of any malfunction, and to display only the most significant to the operators. The service computer will provide facilities for logs and records with output by printer or on magnetic tape.

Last but not least the library computer will provide facilities for file management and storage of programs and data for use by the other computers.

The computer system consists of 25 NORD-10 computers (NORSK DATA ELEKTRONIKK). The interface between the process computers and the apparatus to be controlled will be done by CAMAC associated in most cases with General Purpose Multiplexers described in ref. 3 and shown in fig. 3. At the central control the operator consoles will be connected directly through CAMAC to the computer.

2. THE OPERATION OF CAMAC WITHIN THE SPS CONTROL SYSTEM

The environment of the SPS control system presents a new challenge for CAMAC in which the multicomputer operating system must be interfaced extensively to the real time control of external equipment. CAMAC will provide a standard interface between each NORD-10 computer and external equipment which is both economical and flexible (ref. 4 and fig. 4).

Besides being used as an interface for the central control consoles and the General Purpose Multiplexer equipment already mentioned, CAMAC will be used during the important commissioning period for rapid connection of local displays and other standard peripherals when each computer is operating in a "stand alone" mode. In addition CAMAC will be used as a standard means of controlling high speed data acquisition and control equipment where real time response in the microsecond range is demanded. Consequently local control computers need not be dedicated to such high speed equipment. In this way the processing power of each computer may be conserved to meet the demands of the multicomputer operating system especially at times of "high stress".

3. REAL TIME CONTROL OF EXTERNAL EQUIPMENT

An assessment of the demands for processor time at each computer shows the need for an efficient real time operating system. The real time executive program SINTRAN written by the computer manufacturer (ref. 5) is currently being adopted to meet SPS control system requirements. The SINTRAN monitor is able to schedule and activate programs for controlling external equipment at all priority levels.

At the application program level where speed of execution is not important external equipment may be controlled by programs written in the interpretive language NODAL. The advantages of an interpretive programming language for accelerator control have been widely discussed (ref. 6). Program development is greatly eased as it is possible for engineers and physicists without a large software experience to develop programs which are written directly into the computer and executed immediately.

At a higher level of priority external equipment will be controlled by CAMAC DRIVER programs written in machine code. These DRIVERS which can include many CAMAC commands may be activated by SINTRAN every 20 ms if necessary. Each DRIVER may be linked to LAM request from a CAMAC module. During the intervening period the DRIVER is put into a waiting state. When the external action has been completed a LAM demand is raised. Servicing of this interrupt level sets a flag which when sensed by the SINTRAN monitor reactivates the DRIVER for the next control sequence.

At a still higher level of priority the CAMAC DRIVER may be linked to the interrupt service routine directly. In this way SINTRAN gives control to the interrupt service program which operates repetitively at a priority level above SINTRAN. When the sequence is complete control is returned to SINTRAN which then removes the program from the active queue. Using this technique the computer delays in program control of external equipment are reduced to a minimum. However, the CAMAC driver must be kept short (typically 50 - 100 μs) or overall system response times will deteriorate.

In cases where several pieces of external equipment require real time response of less than 100 μs at the same computer then a certain degree of autonomous control will be provided in each CAMAC crate. In such a system SINTRAN will give control to a hardware micro-program at the specified CAMAC crate. This program can provide dedicated servicing of LAM demands, typically every 10 μs, by repeated execution of a limited string of CAMAC commands. When a sequence is complete a further LAM demand will return control to SINTRAN. In this way the CAMAC system can provide simple multi processor control facilities under the general supervision of the local computer.

For repetitive data transfer from a single module or scanning of module arrays a direct memory access facility is provided. In this way data can be passed between external equipment and computer memory via a CAMAC module upon receipt of an external timing signal.

It can be seen that a wide range of facilities for real time control are provided. Using the SPS CAMAC system each piece of equipment can be commissioned using the simple NODAL language. When this has been successfully completed then the correct real time solution may be chosen or even changed later to improve response time without hardware redesign. There are thus considerable advantages to using the standard system software and CAMAC hardware and a considerable design load is removed from the designers of machine control equipment.

4. CAMAC TO COMPUTER INTERFACE

The various functions of the crate controller have been designed to meet the requirements of the SPS control system as well as the basic requirement for CAMAC crate controllers defined by ESONE specification EUR4100.

A typical SPS CAMAC system will contain from 1 to 6 CAMAC crates connected to the external input/output bus (IOB) of the NORD-10 computer. As a result the full capabilities of the computer IOB including program, interrupt and DMA control are available at each CAMAC crate controller. To satisfy all requirements the

CAMAC control system provides the following functions at each CAMAC crate.

1. A PROGRAMMED INPUT/OUTPUT CONTROLLER (PIO) for normal CAMAC commands.
2. AN INTERRUPT CONTROLLER (INT) which generates an interrupt vector address corresponding to the highest priority LAM demand.
3. A DIRECT MEMORY ACCESS CONTROLLER (DMA) for repetitive data transfer in input or output mode.
4. AN AUTONOMOUS FUNCTION CONTROLLER (AFC) for microprogrammed operation of CAMAC modules within the same crate.

In order that all of these CAMAC control functions may be fully interleaved on a cycle basis both at the level of each CAMAC dataway and also generally on the IOB it has been necessary to design each control function independently with separate data , control and status registers wherever necessary. The common use of the IOB for PIO, INT and DMA functions is determined by priorities at the CPU. At each CAMAC crate request for generation of a dataway cycle by AFC and DMA or PIO controllers are organised by the CC NORD-10 LOGIC.

It is clear that a crate control system in which all types of control are interleaved within a single crate is quite complex and expensive. The overhead is judged to be too high in crates where DMA and AFC operations are not required. Consequently a flexible modular multi-controller system has been designed which can be initially built up or later expanded according to the specific requirements of each CAMAC crate.

The PIO and INT functions, which are always necessary, are provided by a double width crate controller (CC-NORD 10) occupying stations 24 and 25 of each CAMAC crate. The DMA and AFC functions are provided by separate single width control modules which may be connected only where necessary as shown in Fig. 3. Interconnection between the various modules of the control system is made by a front panel LINK BUS.

Besides the PIO and INT functions the CC-NORD 10 is responsible for :-

a) interleaving the various requests for use of the CAMAC dataway (GRANT LOGIC)
b) generating the CAMAC dataway cycle
c) interfacing the CAMAC control system to the external IOB of the NORD 10 computer.

Considerable controller simplicity has resulted in restricting CAMAC data transfers to 16 bit words. In those cases where 32 bit precision is required two successive 16 bit transfers are made without additional software overheads, whilst the dialogue between the CAMAC module and external equipment is completed only on the last transfer.

5. PROGRAMMED INPUT/OUTPUT CONTROLLER (PIO)

Each CAMAC crate controller is addressed as a separate peripheral attached to the NORD-10 external IOB and contains three 16 bit registers : DATA register, NAF register, CONTROL and STATUS register. Each of these registers may be over written by program control and read back for diagnostic purposes without executing CAMAC commands. The NAF register uses 14 bits, (5N, 4A 5F) to specify the CAMAC command. The most significant two bits may be set if Q and X are expected. A high priority interrupt is sent if either Q or X are not present at S1 during the dataway cycle.

The IOB device and register address follows the same format used for addressing standard NORD-10 peripherals. Special IOB addresses are used for generating dataway Z and C cycles under program control.

In order that DMA and AFC demands may be handled it is necessary for the respective controllers to present their demands together with required module address N to PIO controller which will then grant dataway execution time to each of the four demands on either a cyclic or priority basis or a mixture of both. As none of these operations will use the address, data or control and status register involved in any other CAMAC control operation it is possible for all operations to proceed concurrently on the interleaved basis. Instantaneous requests for PIO, INT or DMA cycles are organised by the computer CPU. In the absence of DMA or INT request the PIO controller needs access to the CAMAC dataway only during the IOB instruction in which the dataway cycle is generated. If the AFC has control of the dataway at that instant, the IOB instruction must delay its execution for up to 2 μs until the AFC cycle is complete.

6. INTERRUPT CONTROLLER (INT)

The interrupt control section of the crate controller contains facilities for handling up to 23 LAM sources from normal CAMAC stations. These 23 sources are mixed and patched within limits to form up to 16 graded LAM(GL) signals within each crate. Each GL may be individually masked from a mask register (CMR). The enabled GL signals are "OR"ed to form a crate demand signal which may be attached by program to one of three NORD-10 interrupt levels. The highest priority GL is coded, added to the 4 bit crate address and the resulting 8 bit interrupt address is sent to the computer.

The interrupt system uses four distinct interrupt levels, the highest being reserved for CAMAC Q and X error traps and fast real time operation. This error facility provides high security CAMAC operation without wasting time checking Q and X by program. The advantage of CAMAC GL demand handling within 10μs becomes obvious in large systems with a sequential hardware wired priority chain. The hardware priority in such a system has then relatively little importance and the real processor service priorities involving perhaps several CAMAC operations and some data analysis is determined by the software operating system. The mask register associated with INT control may be loaded or read directly using special IOB instructions with selective bit set and bit clear facilities.

7. DIRECT MEMORY ACCESS CONTROLLER (DMA)

The DMA controller is a single width control module which is capable of repetitive input or output data transfer from a single module in interleaved DMA mode or block mode. In addition an address scan mode with station and subaddress auto-increment is provided. Linking of two DMA controllers permits various combinations of complex data transfer. One possibility is to acquire data from random CAMAC module addresses. In this case one table in core collects data whilst another supplies the CAMAC address.

Each DMA controller contains the following registers which are normally set by program before the channel is enabled.

1. Control and Status register
2. Memory address register (address of data)
3. NAF register (CAMAC command to be executed)
4. Word count register (12 bits)

Each register can be read back under program control. The module contains a LAM source, which is enabled at the start, and set at DMA end of the specified transfer to inform the computer of DMA completion. In block mode, data transfer rates up to 1 MHz may be achieved with the CAMAC dataway cycle overlapping the computer memory cycle. In this case a permanent request is raised to the PIO grant logic.

Because a single DMA cycle of a memory and then a dataway cycle or vice versa a CHANNEL BUSY LINE informs other controllers so as not to send further requests until the current DMA two part cycle is finished. An implemented diagnostic tool is the ability to initiate a single DMA request by programmed CAMAC command (F 25). In this way diagnostic software can work step by step through a DMA transfer waiting at each step for the next value to appear in core and checking that the DMA registers have been incremented.

8. AUTONOMOUS CONTROLLER (AFC)

The addition of the autonomous function permits the expansion of the CAMAC system to meet the parallel real time demands of several pieces of external equipment whilst the computer is engaged in more important system activities. Basically the AFC permits the execution of a simple repetitive sequence of up to 16 CAMAC NAF instruction words either for control of specific modules or for data transmission between modules within one CAMAC crate. In this respect a 16 bit accumulator will be provided for temporary storage of CAMAC data between successive read and write autonomous dataway cycles. Initially each instruction word of the autonomous instruction sequence is loaded by subaddress into a 16 word memory within the AFC under PIO control. If necessary this subaddress facility may be later used to modify a single instruction under PIO or AFC control. In this way the autonomous sequence may be made to branch as a result of some external action.

The sequence of autonomous NAF instructions is addressed by a cycling 4 bit program counter. After each autonomous dataway cycle the program counter is incremented by one for the next instruction. The sequential nature of the program counter may be modified according to the two most significant bits of the 16 bit instruction word. If bit 16 is set then the current NAF instruction is repeatedly executed until a Q response is obtained. If bit 15 is set then the program address counter is reset to zero after executing the current NAF instruction. This facility permits recycling of autonomous programs shorter than 16 words. At the same time the "sequence loop" counter, initially loaded under PIO control with the required number of autonomous program loops, is decremented by one. When the loop counter reaches zero a LAM request is generated which informs the computer that the current autonomous sequence has been completed. Further autonomous cycles are inhibited until the LAM source is serviced by the controlling program.

The exact timing of a request for each autonomous cycle can be determined by an internal clock or by a train of regular or irregular timing signals as required by external equipment.

9. CONCLUSION

A typical application of a CAMAC real time control equipment is shown in Fig. 4. A high speed analog multiplexer (up to 128 channels) is controlled by an AFC either in sequential scan mode or in random address mode. The ADC results are directly loaded through DMA into core memory or into a CAMAC Buffer Memory under AFC control. This is a typical example where real time acquisition is done whilst the computer deals with task organization, data processing or message transfer to or from the SPS operator.

(1) CROWLEY-MILLING, M.C. - The Control System for the SPS. CERN Report Lab II-CO/72-1.

(2) ALTABER, J., HYMAN, T., RAUSCH, R. - Proposal for the SPS Message Transfer System. CERN Report Lab II-CO/CC/73-13.

(3) RAUSCH, R., SAINSON, J.M., WILHELM, R. - General Purpose Multiplex Equipment for Analogue and Digital Acquisition and Digital Command for the SPS Computer Control System. CERN Report Lab II-CO/73-2.

(4) COLLINS, M., RAUSCH, R. - A Multicontroller Approach to CAMAC for the SPS Control System. CERN Report Lab II-CO/73-1.

(5) A/S NORSK-DATA ELEKTRONIKK - SINTRAN Users Guide. February 1974.

(6) HYMAN, T., SHERING, G. - Preliminary Programming Information for the SPS Control System. CERN Report Lab II/Int/CC/73-5.

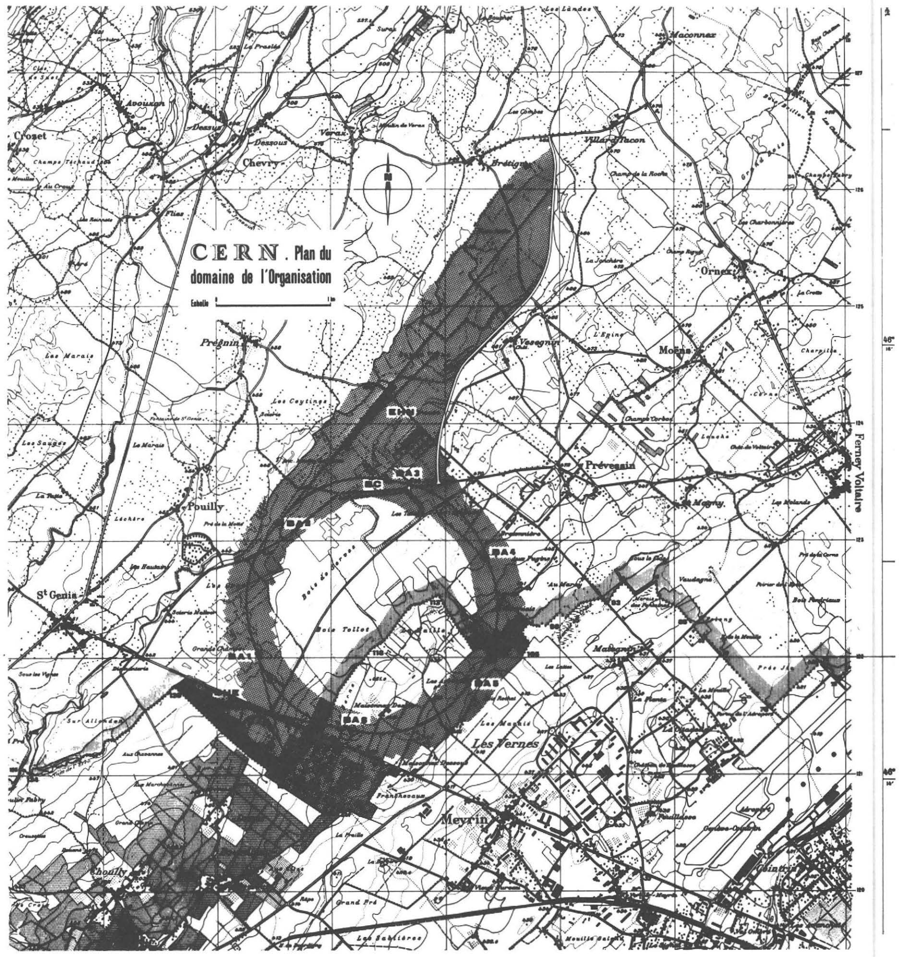

CERN. Plan du domaine de l'Organisation
Meyrin
Les Vernes
Prévessin
Ferney Voltaire

Fig. 1

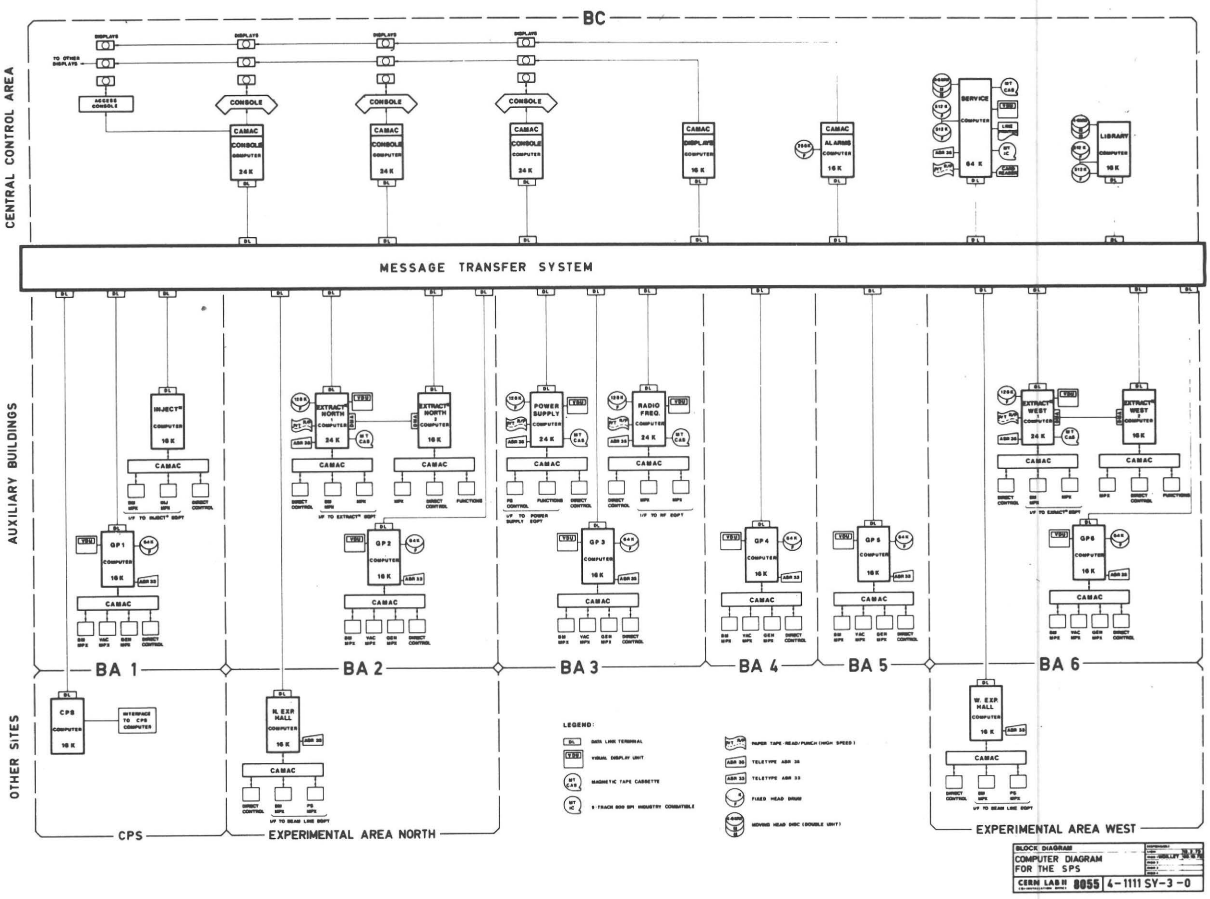

BC
CENTRAL CONTROL AREA
CONSOLE
CAMAC
MESSAGE TRANSFER SYSTEM
AUXILIARY BUILDINGS
OTHER SITES
BA 1
BA 2
BA 3
BA 4
BA 5
BA 6
CPS
EXPERIMENTAL AREA NORTH
EXPERIMENTAL AREA WEST
LEGEND:
BLOCK DIAGRAM
COMPUTER DIAGRAM FOR THE SPS
CERN LAB II 8055 4-1111 SY-3-0

FIG. 2

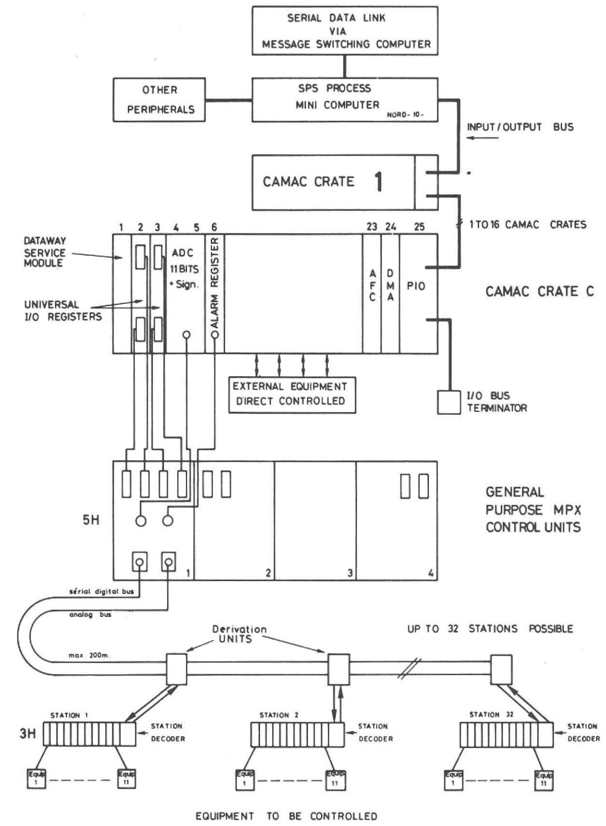

SERIAL DATA LINK
VIA
MESSAGE SWITCHING COMPUTER
OTHER
PERIPHERALS
SPS PROCESS
MINI COMPUTER
INPUT / OUTPUT BUS
CAMAC CRATE 1
1 TO 16 CAMAC CRATES
DATAWAY
SERVICE
MODULE
UNIVERSAL
I/O REGISTERS
ADC
11 BITS
+ Sign.
ALARM REGISTER
AFC
DMA
PIO
CAMAC CRATE C
EXTERNAL EQUIPMENT
DIRECT CONTROLLED
I/O BUS
TERMINATOR
GENERAL
PURPOSE MPX
CONTROL UNITS
5H
Derivation
UNITS
UP TO 32 STATIONS POSSIBLE
3H
STATION
DECODER
EQUIPMENT TO BE CONTROLLED

FIG. 3

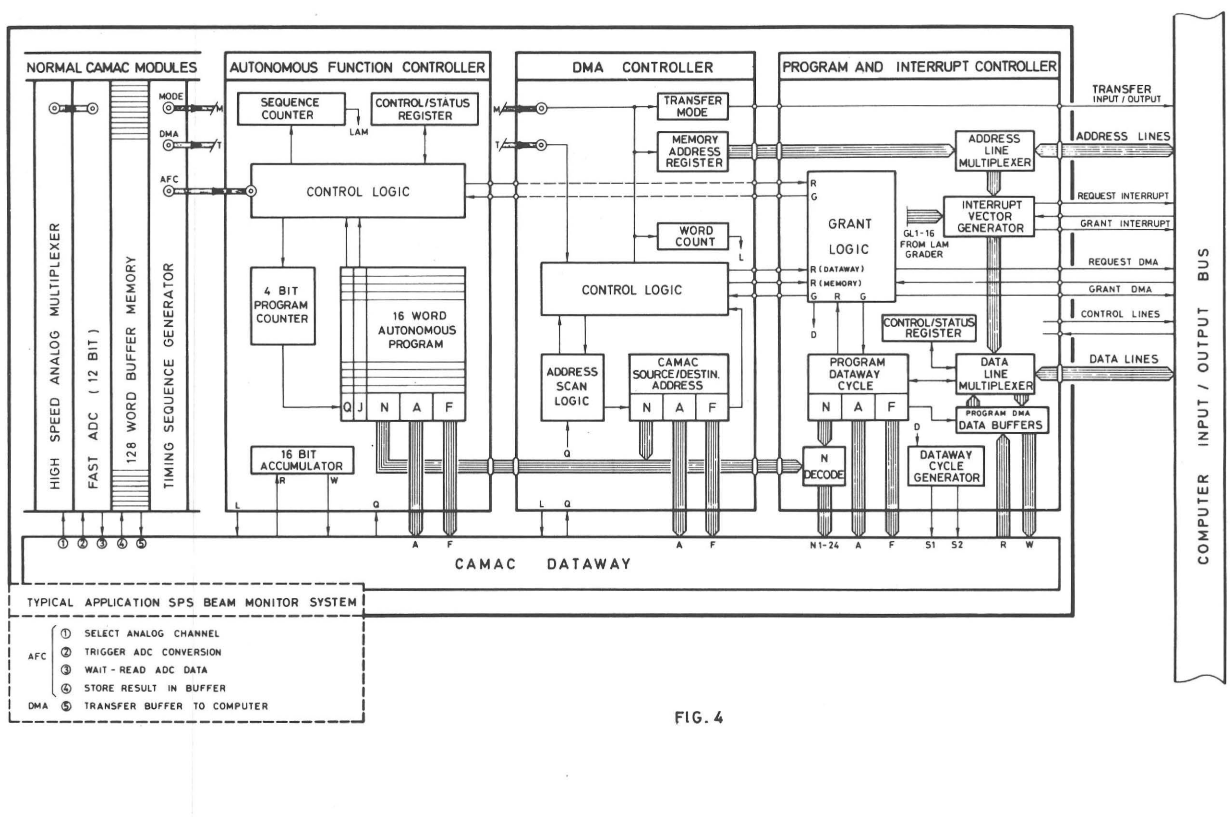

NORMAL CAMAC MODULES
AUTONOMOUS FUNCTION CONTROLLER
DMA CONTROLLER
PROGRAM AND INTERRUPT CONTROLLER
HIGH SPEED ANALOG MULTIPLEXER
FAST ADC (12 BIT)
128 WORD BUFFER MEMORY
TIMING SEQUENCE GENERATOR
SEQUENCE
COUNTER
CONTROL/STATUS
REGISTER
CONTROL LOGIC
4 BIT
PROGRAM
COUNTER
16 WORD
AUTONOMOUS
PROGRAM
16 BIT
ACCUMULATOR
TRANSFER
MODE
MEMORY
ADDRESS
REGISTER
WORD
COUNT
ADDRESS
SCAN
LOGIC
CAMAC
SOURCE/DESTIN.
ADDRESS
GRANT
LOGIC
ADDRESS
LINE
MULTIPLEXER
INTERRUPT
VECTOR
GENERATOR
CONTROL/STATUS
REGISTER
PROGRAM
DATAWAY
CYCLE
DATA
LINE
MULTIPLEXER
DATA BUFFERS
DATAWAY
CYCLE
GENERATOR
TRANSFER
INPUT / OUTPUT
ADDRESS LINES
REQUEST INTERRUPT
GRANT INTERRUPT
REQUEST DMA
GRANT DMA
CONTROL LINES
DATA LINES
CAMAC DATAWAY
COMPUTER INPUT / OUTPUT BUS
TYPICAL APPLICATION SPS BEAM MONITOR SYSTEM
① SELECT ANALOG CHANNEL
② TRIGGER ADC CONVERSION
③ WAIT - READ ADC DATA
④ STORE RESULT IN BUFFER
⑤ TRANSFER BUFFER TO COMPUTER

FIG. 4

DISCUSSION

Q - G. Christaller

Why is CAMAC used in this huge CERN SPS system only at one level of the whole system and not e.g. for the multiplexers or the message transfer system? We understood that CAMAC is the more efficient the bigger the system.

A - R. Rausch

At the message transfer level the store and forward principle is used, 50 DMA's have to be serviced and each channel speed is 30 K words/sec. Interleaved mode is thus mandatory to cope with simultaneous message transfer. In this mode the CAMAC DMA's speed is 500 K words/sec which means only 10 K words/sec per channel at times of high stress. This speed required normal interfaces directly on the I/0 bus. CAMAC might well be used in a message transfer system where either the number of computers is small or the data link speed is lower.
An alternative CAMAC system could work in direct connection between source and destination computers rather than in store-and-forward. The message switching computer would then act as a message routing device under CAMAC control.
At the General Purpose Multiplex level, due to the large number of equipments to be controlled, the interface has to be optimized (3000 modules have to be manufactured). The design optimization for modules, crates and power supplies permits appreciable savings. The modules contain full isolation from the controlled equipments by opto-couplers or reed relays. The front connector must be of high reliability. These features represent a significant contribution to the module cost. We might in the future use the CAMAC serial crate controller for building-to-building connections.

Q - L. Besse

It seems to me the SPS - Control system is one of the biggest CAMAC applications. Typical figures are:
1) Distribution over kilometers
2) Many identical modules identified just by proper addressing
3) Multi-access by more than one computer
4) Very high reliability required (cost of downtime is a portion of the annual budget for the same time)
Now the questions:
Do you have a dedicated hardware diagnostic program for quick troubleshooting if
- one data bit is wrong (i.e. analogue MPX addressed wrongly!)
- the addressing is mixed up

What are your reliability figures for MTBF, ...
Which kind of back-up features are included...

A - R. Rausch

The problem of the reliability is one of the basic problems for the SPS-computer system. We provide therefore in the whole system as much checking capability as possible.
In the message transfer system between computers, for example, we have the possibility of running systematically scheduled programs for checking all the links periodically even if they are not used at a certain moment in time. Checking is performed with small loops, bigger or complete loops in full duplex mode; obviously this implies a certain amount of software.
For CAMAC reliability, first of all new modules received from manufacturers are checked completely and the equipment is tested fully before installation. For problems occurring during operation we have foreseen, for example, certain forms of self checking loops. So we have designed a dataway service module placed always in the first position in a crate. If something is going wrong we try the interface again completely under program control by addressing the service module, which can operate all the functions and subaddress addressing, to detect if for instance one line is broken. The same applies for the multiplexer which is one level below.
You are right to say that, one might have address bits corrupted for example by noise, and so address a wrong station. For this we have provided a minimum of security. So, at least a parity check is performed on the addressing of each command before an action is taken and if a parity error occurs the command is refused. In case of a two-bit error a wrong module will be addressed.
A basic principle is followed for the overall SPS system to ensure for example that a computer can't destroy major equipment like vacuum chambers or magnets because a wrong operation is performed. For that purpose a hardware interlock system is foreseen which takes all the conditioning into account. So, even if the computer is saying wrongly 'switch off', such a command can't be performed.

A CENTRAL DATA ACQUISITION AND PROCESSING SYSTEM

E. G. Kingham, R.E. Martin
Central Electricity Research Laboratories, Leatherhead, England

ABSTRACT

The paper describes a Central Data Acquisition and Processing System whereby data from rigs throughout a large laboratory site can be collected, processed in real time and the results printed out at the rig.
The system comprises a computer interfaced via CAMAC modules to rigs which may be up to several hundred metres distant. The individual data points are selected under computer control by multiplexers located near the rigs.
The paper describes the hardware and software considerations, refers to the cost savings over alternative methods and to future development.

ZUSAMMENFASSUNG

Es wird ein zentrales Datenerfassungs- und Datenverarbeitungssystem beschrieben, mit dem Daten von allen Experiment-Einrichtungen eines grossen Laborkomplexes aufgenommen und in Echtzeit verarbeitet werden können; Resultate werden am Experiment-Ort ausgedruckt.
Das System umfasst einen digitalen Rechner, der über CAMAC-Einheiten an Experiment-Einrichtungen angeschlossen ist, welche mehrere hundert Meter entfernt sein können. Der Rechner steuert die Auswahl einzelner Datenerfassungsstellen mit Hilfe von Multiplexern, welche sich in der Nähe der Experiment-Einrichtungen befinden.
Der Bericht erörtert Hardware-und Software-Aspekte, behandelt die Möglichkeiten der Kostenverringerung im Vergleich zu anderen Konzepten für Datenverarbeitungssysteme und berücksichtigt künftige Entwicklungstendenzen.

RESUME

Le présent exposé décrit un système central d'acquisition et de traitement de données qui permet de recueillir les données fournies par des équipements situés d'un bout à l'autre d'un laboratoire de grandes dimensions, de les traiter en temps réel et d'imprimer les résultats au niveau de ces équipements.
Le système est composé d'un ordinateur relié par des modules CAMAC à des équipements qui peuvent être situés à plusieurs centaines de mètres de distance les uns des autres. Les différents points de saisie de données sont sélectionnés sous le contrôle d'un ordinateur à l'aide de multiplexeurs placés à proximité des équipements.
Le présent exposé donne une description du hardware et du software, signale les économies réalisées par rapport à d'autres méthodes et les développements futurs.

1. INTRODUCTION

All research work relies on measurement and because of the changes in technology permitting easier processing of data by computer, there has been a consequential increase in the demand for collection of data from experiments. The use of human effort in directly recording large data volumes is impracticable and has led to increasing use of automatic data acquisition systems both in research applications (1) and in commercial applications (2)(3). In the past these have mostly been self-contained, purpose built systems dedicated to a specific task and offering at best little more than conversion of the analog quantities to engineering units, alarm monitoring and output in a form suitable for subsequent off-line processing. Usually such dedicated data logging systems are inflexible and can also suffer from speed restrictions.

More recently dedicated systems using small computers have become available permitting the advantages of on-line data reduction and thus reducing the output data volume. Nevertheless, such systems are often under-utilised, still require to be individually specially configured for the application and in particular may each require a considerable amount of software effort. Rationalisation of the choice of computer is essential if this effort is to be minimised and this is difficult in the climate of a fast-changing technology.

Many of the above problems can be reduced by the use of a single larger time-shared system located in a convenient central area and connected by cables to the individual users locations.

The use of a larger system removes many of the constraints, such as limitations of processing ability inevitable in the smaller dedicated systems. Furthermore, the use of a larger system permits the ready employment of a high level programming language with consequent advantages.

A prime need in a laboratory environment is that of flexibility to meet rapidly changing and often unpredictable requirements in a speedy and economic manner. To this end it is desirable to employ a standardised and non-proprietary system of peripheral hardware. A decision was taken to use the international CAMAC digital data interface system (4) and to extend this to meet the analog acquisition requirements by the development of suitable additional modules.

In the Central Data Acquisition and Processing System (CDAPS), the users at various rig locations throughout the C.E.R.L. site need to connect transducers on their rigs to the system. This is effected through cheap multiplexing switches located adjacent to the rigs. The multiplexers are controlled from the computer by digital addressing so as to permit any one analog from a given transducer to be accessed by the central equipment. Such a method results in a very considerable saving in the many cables which would be required if transducer selection were effected centrally.

The channel capacity of the system is potentially very large but in practice this is determined by the extent of the processing which may be in use, the sampling frequency and the available storage facilities in the computer. The existing system can accept up to 1000 analog signal sources and multiplexers are provided for 750 channels. The CAMAC system and the cabling also permits the acquisition of digital data where this is found necessary.

A suite of software programs have been developed which organise the regular collection of data via the CAMAC and multiplexer system and its location in the computer store. This data is then output to each of the users at the required intervals via a teletypewriter.. The users may also opt to carry out further processing by the insertion of their individual programs written in a high level language (at present "CORAL")(5). A brief specification of the main system properties is given in Appendix I.

2. CDAPS

Figure 1 illustrates the CDAPS system in simplified form. The operation of the system is considered in Section 2.1 whilst the cable and termination systems are dealt with separately in Section 2.3. Details of the software are discussed in Section 5.

2.1 Operation

The analog signals are selected by unique addresses sent to the multiplexers at 0.1s. intervals. This address, which consists of a 10-bit word from a CAMAC driver, calls the appropriate analog signal way in both a central and one of several remote multiplexers. Also included are optional signal conditioning facilities and cold junction compensation.

Simultaneously a central digital voltmeter (DVM) is called upon to transfer the previous measurand and is prepared for the new sample. An 80ms integration period is provided to reduce power frequency interference. Increased scanning speeds above 10 per second can be provided at the expense of interference rejection.

2.2 Data processing

The data thus collected can either be output to a user together with identification, time etc., or stored for subsequent processing.

Because many users require data acquired at fairly slow rates time is available in which the multiprogramming facilities of the Modular One computer can be employed to give on-line processing to some users. These users programs are run together at the same time as the background logging routines.

2.3 External connections

Conventional 10 twisted-pair, screened over-all telephone cables are used in a radial configuration with distances up to 500m.

These carry outgoing multiplexer addresses and processed data. A separate cable carries the incoming analog signal. These cables are all terminated at a central cubicle with patching facilities and numerous junction points are provided throughout their length to which multiplexers and teletypewriters can be connected at experiments in individual laboratories. Up to 16 multiplexers can be connected at any one location.

2.4 Multiplexers and analog system

The system uses specially developed low level signal reed relay multiplexers which are now commercially available. They are digitally addressed from a CAMAC teletypewriter driver and effect 4-pole switching in blocks of 16 switches. The multiplexers are designed for use at the rig areas so as to minimise cabling and therefore incorporate an optional cold junction compensation facility which can be called by program.

The choice of multiplexers, cabling and digital voltmeter permits analog signals in the range 0-100mV to be resolved $\pm 10\,\mu$V at a rate of at least 10 per second. 50 Hz series and common mode interference can be rejected by 70 dB and 140 dB respectively.

There are alternative means permitting higher speed signal sampling should this later be required. This may use several fast analog-to-digital convertors or remote CAMAC crates using digital transmission to CDAPS.

2.5 Output system

Digital data is output to each of the rig area on teletypewriters. This data may be in the form of a simple log, or include alarm information and processing aimed at data reduction. The option also exists for this data to be output by punched paper tape for further subsequent off-line processing. The users teletypewriter keyboards can provide for control of particular programs. The flexibility of CAMAC permits other output devices to be used as required.

3. CHOICE OF COMPUTER

A number of machines were considered and the best choice satisfying the requirements appeared to be the Modular One manufactured by Computer Technology Ltd.

The prime requirements for such a system were:-

1. Modularity
2. 16 bits or more
3. Cycle time better than 1 μs
4. Supporting software availability
5. CAMAC controller availability
6. High level language
7. Adequate commercial viability
8. Core store 8K extendable to 32K minimum
9. Multiaccess/multiprogramming feature
10. Direct memory access.

In the event item 10, which was to have been satisfied through the CAMAC controller, is not available.

The present configuration of the machine comprises 32K words of core store plus two executive teletypewriters, a fast paper tape reader and the CAMAC controller.

Principally for program development, use is also made of a line printer. The latter, together with fast paper tape punches, is interfaced via CAMAC.

The modularity of the system has proved an advantage so as to permit easy enhancement and reconfiguration of the system which will later include a disc backing store, second processor and FORTRAN compiler.

The advantages to the non-specialist user or programmer of writing in a high level language such as CORAL are manifold and the penalty of some extra storage requirements was considered well worthwhile. An additional benefit is gained by thus improving the security of the system against unintentional maloperation by the user.

The storage requirements of the CORAL compiler finally supplied as a part of the original system are such as to require compilation to be effected in an off-line manner with the present system configuration, or separately on a Myriad computer. The later enhancement by the additions referred to above will permit the on-line compilation of both CORAL and FORTRAN programs within the Modular One system.

4. COUPLING TO COMPUTER

An essential item is the means of coupling the CAMAC dataway to the computer. Whilst CAMAC controllers exist for a wide variety of commonly used computers it was found that this was not at the time the case for Modular One.

Pending the supply by the computer manufacturers of a CAMAC controller operating on a "parallel" data transfer principle (to achieve high speed) a simpler "serial" controller (working at a limited speed) was evolved by C.E.R.L. This device consists of a teletype driver interface within Modular One and a corresponding CAMAC teletype driver module in the CAMAC crate, each being speeded up by a factor of about 100 times so as to enable serial transfer of 8-bit teletype words at a speed of 1000 character/s.

The "parallel" controller now available enables much higher speeds of transfer (of 16-bit words) to be achieved. It consists of a "2 width" CAMAC module (CTL type 751-1) connected to the processor driver unit (CTL type 1.750) by a 5m length of cable. If expansion is required into a second CAMAC crate in order to accommodate the full number of multiplexer and teleprinter CAMAC driver modules then this will probably be achieved by the use of CAMAC modules type 7033 and 7034. The parallel coupler to Modular One regrettably does not possess a Direct Memory Access (DMA) feature and thus the ready transfer of discrete words or blocks of data direct to store cannot be achieved. The omission of this facility will make the acquisition of data at fast rates more difficult although the development of an add-on option giving DMA is possible.

5. SOFTWARE

5.1 General considerations

A computer operating system consists of several successive levels of program (6). At the highest priority level exists that program which handles the computer peripherals and controls the running of other programs. At the lowest priority level exist the programs of the ultimate user which are probably written in a high level language such as FORTRAN or CORAL.

In the CDAPS system we have attempted to define and separate several levels of program as follows:-

(a) Executive which controls the running of programs and peripherals, and in this case enables many programs to be run concurrently in a multiprogramming manner.

(b) CAMAC Driver which controls the operation of CAMAC peripherals through the Executive.

(c) The Scanning Program which controls the continuous running of the logging system using both the Executive directly and the CAMAC Driver.

(d) Service Programs which handle the data obtained in routine logging operations by the Scanning Program and output messages through the CAMAC Driver to the customers' output devices. This category also includes separate programs used to initiate the logging service and to modify the Scanning Program while it is running.

(e) Customer Programs which are at the lowest level of priority and can in principle be written in Assembler Language or CORAL. At present for reasons of security of operation of the system only CORAL is allowed, although FORTRAN will be available in the future (see Section 7).

5.2 The Executive

The system uses the Computer Technology Ltd. "E2" executive.

5.3 CAMAC Driver

The software has been developed for a 1000 character/s serial data link pending delivery of the fast coupler type 751-1. This software receives requests appropriately arranged from programs requiring to use CAMAC devices and informs the user or service programs on completion of the function. The system is illustrated in Fig. 2. A revised driving program in CORAL has been developed for the type 751-1 CAMAC controller and this permits much faster parallel transfers.

5.4 The Scanning Programs (CDP)

These programs are fundamental to the logging system and have been under continuous development since the beginning of the project.

In essence the "Sample List" defines the order of scanning of the input points which can be completely random, while the "Reference Table" includes all data needed for performing the logging operation. The latter comprises an entry of fifteen words for each input point.

The data obtained, together with the time when it was obtained, is stored in the "Reference Table", as are alarm limits, sampling interval and any other required scanning parameters.

The program is activated by the internal real time clock of the computer every 0.1 second and it initiates a series of successive operations if the information in the "Reference Table" indicates that they are required. These operations, which are illustrated in Fig. 3, include addressing a multiplexer, reading the digital voltmeter, testing for alarm conditions or running a customer's own program. Some limited diagnostic information regarding the operation of the system is also made available as required.

5.5 Service programs

5.5.1 ALLOG

This is a program which accepts data, which is placed in a mutual buffer store by the scanning program,and outputs it to the customer's teletype or other output device.

The present version is fairly simple and consists of eight individual sub-programs which can be used to output as required to any CAMAC device. The message queue into which CDP puts data and from which ALLOG takes its data holds a maximum of two messages for each device. Whether this method of queueing and outputting is optimum for our purposes will be reconsidered in the light of future operational experience.

At present a standard format of output is used which gives all the relevant information regarding the data being output for a particular channel. This takes six seconds to output on a teletypewriter and so, when all such data is required to be printed out, the scanning rates for consecutive points are severely limited. If the incoming message rate happens to exceed the speed at which messages can be output (1 message per 6 seconds) then messages are lost. A more recent program called LOGAL can also drive up to 16 CAMAC devices. Alternative output formats will also be provided in the future.

5.5.2 ADMIN

ADMIN is the program which is used for changing the parameters which CDP uses in scanning. It is possible for instance, to change the alarm limits in the "Reference Table" or frequency of scanning while the system is operating.

ADMIN is integrated with CDP to the extent that each provides information for the other in predetermined stores, and the information so obtained from CDP can be output by ADMIN via any convenient device, for instance a CAMAC driven line printer or a Modular One driven user teletypewriter.

5.5.3 USERPACK

USERPACK is a collection of routines which enable ultimate users of the system to control the running of their own programs and the input and output of data from and to their own devices, that is multiplexers or teletypes. In addition to this function it also provides a safety barrier between the user and the computer system for the system security. The customer (ultimate user) of the system is permitted to write programs in CORAL to analyse his data as required (see Section 5.5.4). The USERPACK is added to the customer's program during the compilation process. Within his program he will then be able to make requests to output data from the Reference Table contained in the Scanning Program etc. An additional useful facility which has been incorporated in USERPACK is that of floating point arithmetic.

5.5.4 Customer programs

In addition to using the standard logging system, including the provision of alarm messages if the alarm levels are exceeded, a customer, subject to the availability of sufficient store, can write and have run his own program to handle his data as he requires it.

A customer's program is written in CORAL and employs the USERPACK routines described above to obtain data from the "Reference Table", to do such computation as is required and to output to his own teletypewriter, or possibly another device such as a lineprinter.

The program can be run automatically each time an input allocated to this customer is evaluated by the digital voltmeter, or as required by the customer and by using the routines provided the customer can control the period of collection of data etc. by using USERPACK to modify items in the "Reference Table".

An element of control of a user's experiment is also possible. Relays, driven through CAMAC and giving on/off control, can be energised by his program as required.

At present all customer programs are written in CORAL and compiled and translated on the Myriad computer at C.E.R.L. When further storage becomes available, together with backing store, it will be possible to use the CORAL compiler supplied by Computer Technology Ltd. New customer programs will then be compiled without resort to Myriad and the older customer programs will easily be changed to permit recompilation on Modular One.

6. COSTS

The present system hardware cost approximately £56.5 K and this includes the processor with 32,000 words of core store, the CAMAC system, 768 channels of multiplexers, 8 of the 10 user teletypewriters, 2 executive teletypewriters and all the costs of cable, 12 cable termination boxes and their installation.

A conservative estimate of the hardware costs of corresponding dedicated systems which would otherwise have been required has already reached a value of about £50K. It is difficult to give precise figures of cost advantages, however, it is clear that significant savings can be achieved by virtue of:-

(a) Better utilisation of capital equipment (individual rigs may only run very intermittently).

(b) Improved maintainability due to standardisation.

(c) Speedy configuration resulting from use of standard stock parts.

(d) Rationalisation of software effort by standardisation and commonality. (It is commonly recognised that on small systems software costs may be two to three times the hardware costs).

(e) Reduction in off-line processing costs and resultant more economic utilisation of rigs since in both cases data is processed as work proceeds and no delays in obtaining results by an off-line method (e.g. IBM 370) are involved.

(f) Saving in space at the rig area is achieved by the use of the physically small multiplexer and teletypewriter.

(g) By centralising equipment the use of more expensive and technically more advanced equipment can become possible because of the more favourable environment and cost sharing between users.

7. FUTURE DEVELOPMENT

The requirement for the processing content of the work load (as opposed to data acquisition) has grown very rapidly. This has resulted in considerable difficulties in concurrently accommodating the necessary extensive programs of a number of users. As a result a scheme has been proposed for extension of the storage capacity of a computer. This will comprise a further 40K words of semiconductor store, a second processor and a total of 4.8M words of disc backing store to be added in 1973.

The availability of such storage will permit the "on-line" use of the CORAL compiler, thus facilitating user program development independently of the Myriad computer. Additionally it will also be possible to use a new Executive "E4" thus permitting the more ready inclusion of special requirements and the choice of FORTRAN IV or CORAL 66 according to the nature of the job requirements. Consequential upon the provision of such facilities is the need to arrange for local program training courses, particularly for the users.

It is now clear that benefits can be obtained from the collection of data from locations outside the main C.E.R.L. site. To this end a CAMAC module permitting digital intercommunication at a fast rate over long distances has been planned (7).

The ability to develop programs more readily will allow special user requirements for data acquisition (e.g. fast data rate over limited periods) to be accommodated more easily than with the present configuration. Such experiments could often be accommodated either directly through a Modular One entry port, through the main CAMAC system, or through a remote CAMAC system according to the nature of the requirement.

The system, being designed to meet a general requirement, inevitably has some limitations, notably applications requiring fast scanning rates together with on-line processing could overload the computer. It must be expected that such applications will often require dedicated systems. Moreover, with the advent of cheap minicomputers, the hardware interface requirements between a complex rig and a central system may now be more cheaply met by their use. Nevertheless the use of CAMAC with such dedicated computers will effect a significant reduction in the cost of interfacing peripheral devices. The software in such cases may be expected to be considerably simpler than is used in CDAPS.

8. CONCLUSIONS

The users needs have been consistently met satisfactorily, with a considerable reduction in the cost and time required to supply more conventional systems. The users have found very considerable benefits in the real time features of the system, within the constraints in user program size which it has been necessary to impose. The constraints of size will be overcome by the system enhancement in hand.

Previous satisfactory experience in the use of the CAMAC system of digital instrumentation has continued to be substantiated by the performance of the CAMAC items, although considerable extra complexity in the computer software has been required. Nevertheless, the choice of a standard data handling system, which is also capable of use with other dedicated processors has been amply justified. The use of the CAMAC compatible multiplexers has proved satisfactory in practice and significant reductions in the cost of transducer cable have resulted. The flexibility of the system has been shown to be worthwhile in view of the ease with which rapidly changing demands can be met. Whilst the continued success of the system depends upon the provision of adequate effort for the software involvement there is no doubt that considerable economies can be achieved in comparison with the manpower required to fully sustain a multiplicity of independent dedicated systems having comparable facilities.

9. ACKNOWLEDGEMENTS

The authors acknowledge the help of the many individuals who have contributed to the evolution of CDAPS, and in particular Mr. G. Hall and Mr. S. J. Chesney who contributed material included in this paper. The work was carried out at the Central Electricity Research Laboratories and is published by permission of the Central Electricity Generating Board.

10. REFERENCES

(1) BISBY, H. "The CAMAC Interface and some applications". The Radio and Electronic Engineer, 41, No. 12, December 1971.

(2) JERVIS, I. M. W. "On-line computers in power stations". Proc. I.E.E., I.E.E. Reviews 119, No. 8R, August 1972.

(3) WELCH, B. R. "Computer systems in C.E.G.B. Nuclear Power Stations". Technical Survey No. 10, Nuclear Engineering International, Jan. 1973.

(4) KINGHAM, E. G., MARTIN, R. E. "CAMAC applications in a Research Laboratory". CAMAC bulletin, review paper to be published 1973.

(5) "H.M.S.O. official definition of CORAL 66". H.M.S.O. 1972.

(6) DIJKSTRA, E. W. "The Structure of T.H.E. multiprogramming System". Comms of the A.C.M. 11, No. 5, 341, 1968.

(7) LAVER, J. A. "A secure data link between autonomous data handling systems in CAMAC". Remote Control System Organisation, I.E.R.E. Colloquium, October 1973.

APPENDIX I

C.D.A.P.S. Specification

Maximum channel capacity	768 in units of 16
Signal input range	10 µV to 100 mV
Source impedance	1000 Ω balanced or unbalanced
Commode mode rejection	>120 dB at 50 Hz at 0.5 full scale
Series mode rejection	>40 dB at 50 Hz at 0.5 full scale
Scanning rate	10 channels/s
Output user teletypes	>8
Data output rate per teletype	10 character/s

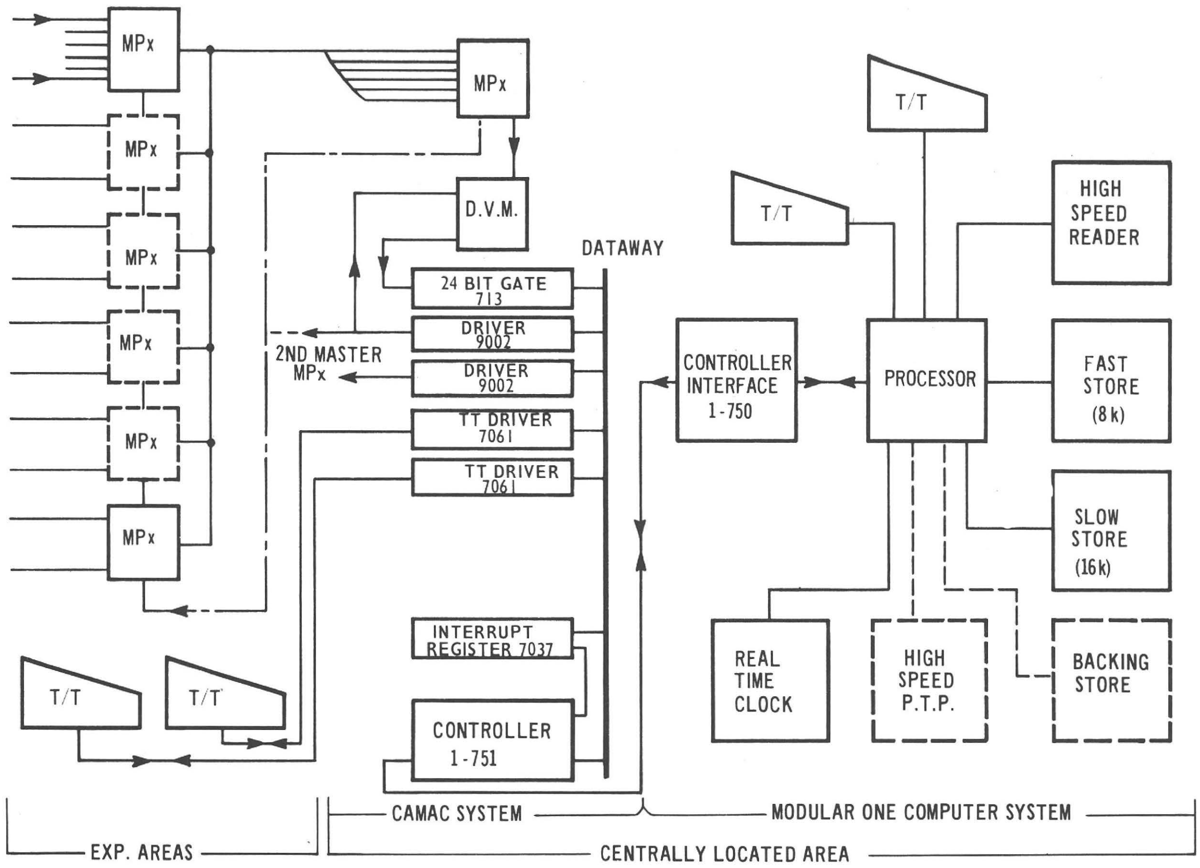


FIG. 1 BLOCK DIAGRAM OF CDAPS HARDWARE

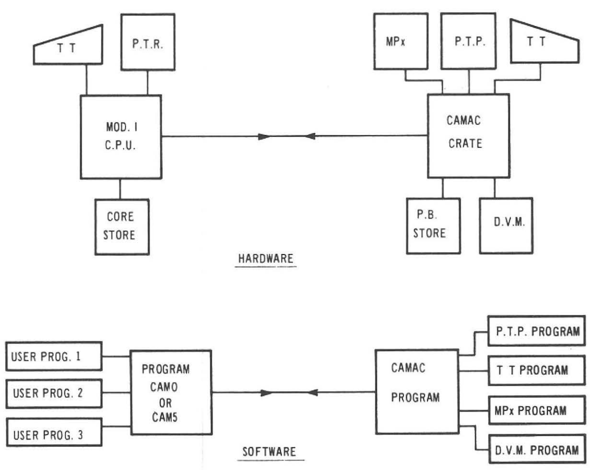


FIG. 2 BASIC CDAPS ARRANGEMENT

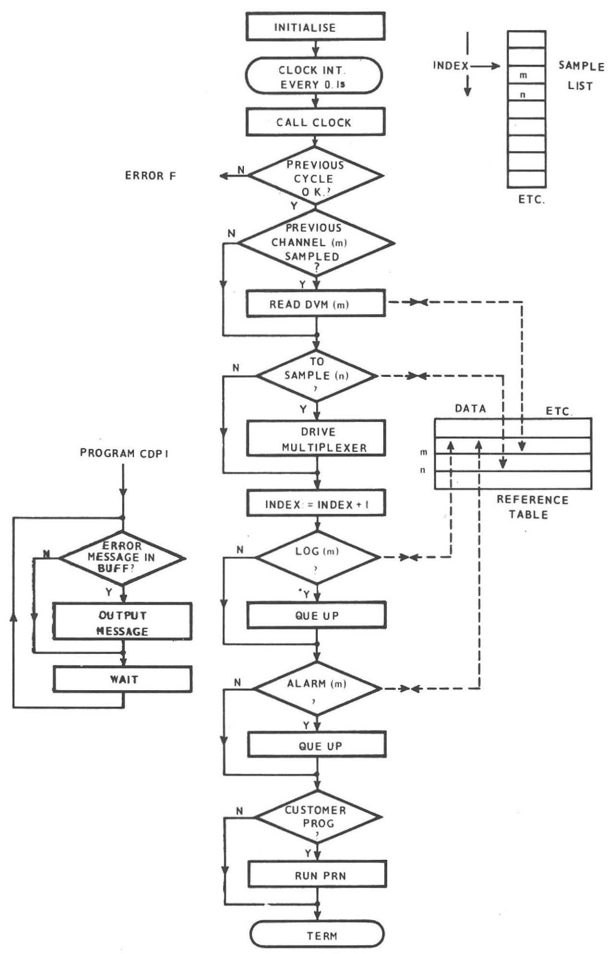


FIG. 3 CDAPS SCANNING PROGRAM

PROCESS CONTROL OF THORIUM ELEMENT REPROCESSING EXPERIMENT

B. Brodda, H. Halling
Kernforschungsanlage Jülich GmbH, Institut für chemische Technologie and Zentrallabor für Elektronik, Jülich, Germany

ABSTRACT

A data acquisition and processing system is built to study the reprocessing of High-temperature-reactor-fuel-elements in a prototype industrial process-loop. Besides acquiring data from the process and different measurement equipment, book-keeping and storage-management of the hot samples is performed. An interactive communication network supports access to the bulk storage system for information retrieval and updating.

ZUSAMMENFASSUNG

Es wurde ein Datenerfassungs- und -verarbeitungssystem aufgebaut zur Untersuchung der Aufarbeitung von Hochtemperaturkernreaktor-Brennelementen. Neben der Datenerfassung aus dem Prozess und von verschiedenen Messeinrichtungen verwaltet es die "heissen" Proben und ihre Lagerung. Ein interaktives Kommunikationssystem gewährt den Zugriff zu einem Massenspeichersystem für die Informationswiedergewinnung und den Änderungsdienst.

RESUME

Un système d'acquisition et de traitement des données a été construit en vue d'étudier le retraitement des éléments combustibles des réacteurs à haute température dans un prototype de boucle de traitement industriel. Ce système permet non seulement de recueillir les données fournies par le dispositif et par divers appareils de mesure mais aussi d'effectuer la comptabilité et la gestion du stock d'échantillons radioactifs. Un réseau de communication permet l'accès au système de mémoire de masse pour le renouvellement et la mise à jour des informations.

Introduction:

To study the reprocessing of High-temperature-reactor-fuel elements (1o g THO_2 + 1,1 g UO_2 per element, enriched to 93 % and embedded in graphite) a prototype industrial processloop is built. It is expected to get information about the flow of fission material under realistic conditions. The problems one faces here seem to be typical for a laboratory automation system.

The processloop with its control station is located in the hot cells. It consists of 55 retorts. Each retort delivers two measurement values (pressure and temperature) per second. Samples of high activity are taken from the retorts, filled into glass bottles and sent to the warm cells in the laboratory by pneumatic post. There the samples are stored in a depot from which they are later fetched for different measurements such as spectroscopy, chromatography, automatic titration, density measurement, x-ray fluorescence and γ-measurements. Some of these measurements need sub-samples which are prepared using semiautomatic apparatus and which are also stored in the depot for later use. Each measurement station in the laboratory, the process-control-station and the sample input station at the process-loop contain a terminal to enable interactive dialogue with the main-processor.

The most important job of the processor-system is to identify the paths of the samples, to supervise the depot-management, to establish a block on the disc per sample and fill it with information which may come from the person taking samples, the experimentors in the laboratory, the system manager or the results of the different measurements.

There must be high security that no samples are exchanged or information belonging to a sample is sent to another one. Copies of the files to secondary storage medium (Dual DEC tape) are made regularly. The final data processing and calculations are done on a IBM 37o/165 but the system should allow small intermediate calculations (like a desk calculator) simultaneously with the operation of the system.

Hardware: (Fig. 1)

The hardware system is divided into 2 separate parts. The first part is a data acquisition and control-system including simple display features. Due to the great variety of equipment CAMAC is the best approach to meet these demands. The flexibility of CAMAC makes it easy to expand the system step by step.

There are two crates, each holding about 15 modules. One crate is located in the hot cells 2oo m from the main processor, the other one is in the laboratory 2o m from the processor. Both crates use BORER type U crate-controllers which are directly connected to the Unibus. The unibus is extended by using balanced pair extenders. It is worth mentioning that there is no software-change due to the extensions.

Seven types of modules are used. Relay-multiplexer, A/D converter, display, input register, output register, interruptmodule and timer. All of the CAMAC actions are started by the appropriate LAM signal.

In the hot cells at the process loop a timer generates LAMs to switch the multiplexers, to digitize and read the pressure and temperature values (11o per second). Each sample is filled into a 5 ml glass bottle which is marked with a magnetic pattern. This pattern is read automatically using a "datapen"into a CAMAC input register generating a LAM. Upon competion of initial data-taking, the sample is transferred to the laboratory by a pneumatic post system. There it is stored in a depot. The storage slot number is added to the informationblock for later checks.

The second CAMAC crate is located in the laboratory at a distance of about 2o m from the processor. It mainly contains input modules for data acquisition from the experiments and output modules to control simple actions. All CAMAC I/O is again interrupt driven on a single word transfer basis.

The second system is the communication system. It consists of standard peripherals connected to the computer by way of bitserial lines. It includes Teletypes, DEC-writers TEKTRONIX 4o1o displays and VTo5 character displays. The communication system was chosen to work outside CAMAC mainly due to available software and to obtain better fail-soft features.

Software: (Fig. 2)

The heart of the system is the DOS system from DEC. The files holding the information about the samples and subsamples are block-oriented and each block holds the sample information in a predefined frame. This allows fast updating and searching. The fileframs are set up at initialisation time by the system manager.

At runtime a simple statement sharing system -where the duration of the statement execution must be limited - changes from one user to the next executing a single statements for each of them. Each user (i.e. each terminal) has its own interpreterset which is tailored to his activities. Two terminals allow BASIC calculations, one terminal allowes manipulations on datasets using a display. Most of the terminals only have access to the depot management and to their appropriate information in the sample blocks e.g. the sample taking terminal to the retort number where the sample was taken, the process-controll station-display-terminal to the present pressure and temperature values and the concentration and volume information derived from it etc.

The users-except the system manager - are therefore only faced with simple parameter oriented interpreterstatements. They need not know anything about the DOS system and they are not faced with CAMAC.

The CAMAC handlers are fast interruptroutines filling or emptying buffers with which the interpreterprograms are able to communicate controlled by synchronising variables. Only the system manager has to set up or release the interrupt connections of the CAMAC handler-routines. The handlers are programmed using a set of MACRO's - a simple preversion of the IML set. Because each module has its own interrupt vector, the routines can be fast and should not run longer than 2oo μsec.

The whole hardware- und softwaresystem will go into operation step by step. First the file handling and the interpretersystem running under DOS is tested under operating conditions. The final version will be finished in late 1974 when the chemical process will be installed. We think that this flexible concept is also of interest for similar applications in the field of laboratory-automation.

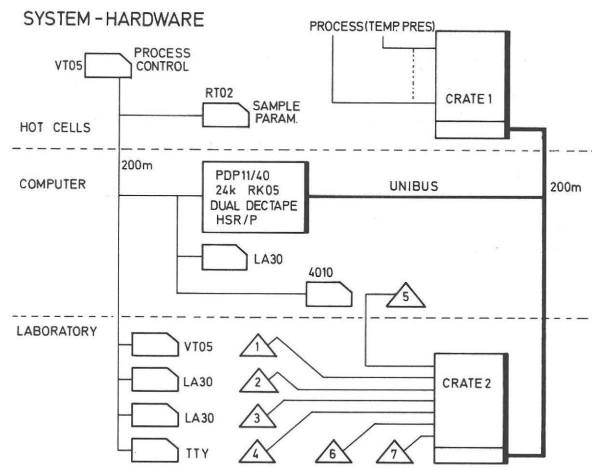


FIGURE 1. The Hardware System

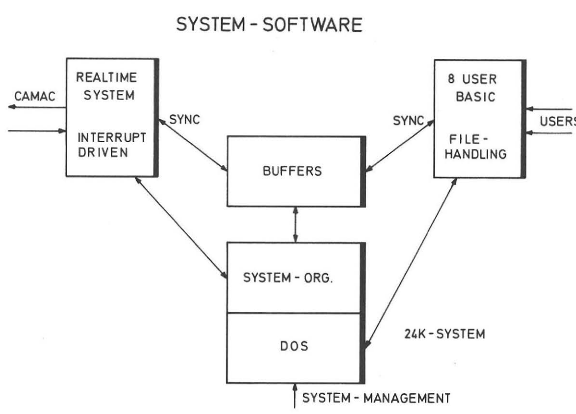


Figure 2. The Software System

DISCUSSION

Q - R. Hagelberg

You used a dedicated crate controller which was connected to the computer via the Unibus. What can be the length of the Unibus from the computer to the crate without having troubles in a normal physics experiment surrounding, and without using special extenders or expensive Unibus transmitters and receivers?

A - H. Halling

The manufacturer restricts the bus without extenders to 50 feet. One has to subtract from this the connections from the bus to the attached peripherals and also take care of the bus-loads. For connections longer than 5 m between the computer cabinet and the crate containing the Type 'U' controller use differential extenders.

THE APPLICATION OF CAMAC TO THE EVALUATION OF AXIAL FLOW FANS

W. P. Gertenbach
Instrumentation Division, Atomic Energy Board, Pelindaba, South Africa

ABSTRACT

A CAMAC instrumentation system was used in an automatic computer-based control and data acquisition system for the evaluation of axial flow fans. The system measures fan characteristics at constant fan speed and static-pressure ratio between fan inlet and outlet. An evolutionary operations (EVOP) strategy was employed for controlling fan speed and static-pressure ratio.

ZUSAMMENFASSUNG

Ein CAMAC-Instrumentierungssystem wird in einer automatisierten, von einem Rechner betriebenen Anlage zur Steuerung und Datensammlung bei der Auswertung von Axial-Ventilatoren benutzt. Das System misst Ventilatoren-Charakteristiken bei konstanter Ventilatordrehzahl und konstantem Verhältnis des statischen Druckes zwischen Ventilator-Einlass und Auslass. Zur Regelung der Ventilatordrehzahl und des Verhältnisses des statischen Druckes wird eine Verfahrensweise angewendet, die unter dem Namen "evolutionäre Betriebsweise" (EVOP) bekannt ist.

RESUME

Un système d'instrumentation CAMAC a été employé dans un système de contrôle et d'acquisition de données, basé sur computateur, pour l'évaluation d'éventails de flux axial. Le système prend les mesures des propriétés d'éventail, à vitesse d'éventail et proportion de pression statique constantes entre l'entrée et la sortie de l'éventail. Une stratégie d'opérations évolutionnaires (EVOP) a été employée pour le contrôle de la vitesse de l'éventail et de la proportion de pression statique.

1. INTRODUCTION

An evaluation program for axial flow fans proved to require a considerable amount of data collection and reduction; it was therefore decided to construct an automated system to perform this task. Furthermore, the delivery schedule for the system was fairly critical and it was decided that the system should be programmable in a high-level language such as BASIC in order to minimize programming effort. It was decided to use CAMAC as the instrumentation and interfacing system in view of its modularity and system expandibility; it was also seen as an interesting application of our inhouse-developed hardware.

2. SYSTEM DESIGN

The fan is driven by an internal combustion engine. the speed is controlled by adjustment of the throttle of the engine by means of a stepping motor. The fan output flow is throttled with a damper (also driven by a stepping motor) in order to adjust the static-pressure ratio R between fan inlet and outlet.

The data collection and control system consists of a 16-bit minicomputer with a 16k memory as shown in Fig. 1. It is connected via a CAMAC Branch Driver, Crate Controller and CAMAC modules to the engine and fan. The computer and the Branch Driver are located about 50 metres from the experimental site in order to provide a relatively quiet environment for the operator.

A control algorithm adjusts fan speed N and pressure ratio R to 100 different points (on a N,R response surface). The data collection program collects 780 measurements at each of these system states. Data are collected in blocks and stored sequentially on a cassette-type magnetic tape unit. An off-line data reduction program, also written in BASIC, reads the data from the magnetic tape unit and reduces it to a suitable format.

3. MEASUREMENTS

Measurements of fan speed, engine torque, ambient temperature, atmospheric pressure, and three other sets of pressures and temperatures are required. These three sets of measurements are taken, as shown in Fig. 1, at three measuring stations which are at the inlet and the outlet, and in between the rotor and the stator of the fan.

At each measuring station the static pressures on the casing and hub of the fan are measured, using a piezometer ring. Furthermore, in order to determine the radial-pressure profile between these two points, three pressure probes for three-dimensional flow measurements[1] are positioned at 120 degree intervals around the circumference of the fan at each measuring station. These probes are positioned radially in the air flow stream by stepping motors and each provides six pressure signals from which the dynamic pressure, static pressure, pitch and yaw differential pressures at that particular radial position can be deduced as shown in Fig. 2. However, in order to save transducers, the 18 pressure signals from the three radial-pressure probes at each measuring station are multiplexed by means of solenoid valves to four pressure transducers, as shown.

Thermocouple rings measure the temperature differentials between measuring stations and the environment.

4. THE CAMAC SYSTEM

CAMAC hardware developed and manufactured by the Instrumentation Division of the South African Atomic Energy Board (using semi-automatic wire-wrapping techniques) has been used in the interfacing system.

The computer has to be interfaced to 11 stepping motors (positioning pressure probes, controlling engine speed and damper aperture), 18 differential pressure transducers, four temperature measuring devices, engine speed and torque transducers, and nine sets of six solenoid valves.

Stepping motors are driven by means of CAMAC stepping-motor modules. The direction of rotation, motor speed (any of eight speeds), and number of steps (1 to 4095) are written into the module using the CAMAC write command. One CAMAC module contains the logic to pulse three stepping motors. The CAMAC pulser output pulses are routed to stepping-motor driver cards. These are housed in a separate Elmaset rack and contain the stepping-motor switching logic, drive transistors, series resistors (for improving the motor torque characteristics) and, on the front panel, pushbuttons and switches for manual control.

Analog signals are conditioned in CAMAC signal-conditioning modules. From these modules, each containing eight low-drift differential input amplifiers (with gain adjustment by CAMAC command), the analog signals are routed to 16-channel multiplexer modules. These modules, using COSMOS quad bilateral switches as the multiplexing elements, multiplex the analog signals to a 14-bit integrating-type analog-to-digital convertor (ADC). The analog channels can be selected randomly by the multiplexer using the appropriate CAMAC command or, as in the data acquisition program, the multiplexer and ADC are used in a scan mode whereby the channel scan range is written into the multiplexer module using the appropriate CAMAC command. The ADC is then initiated. When the conversion has been completed, the ADC temporarily stores the data in a latch on the module, generates a "Look At Me" signal to the crate controller, sends a pulse to the multiplexer to select the next channel and then starts the next conversion during which the CAMAC system transfers the digital data in the latch to the computer.

The solenoid valves are controlled from a CAMAC digital output register. Each set of valves can be closed or opened using a different CAMAC command.

5. INTERFACING TO BASIC

In order to provide the users with a simple means of programming the system, a CAMAC driver routine, which can be called from BASIC, was written in Assembler language and integrated into the BASIC interpreter. To communicate with the process, the programmer uses the following statement:

```
10 CALL (1,F,C,N,A,D,Q)
```

where F,C,N,A denotes the required CAMAC command, D the data parameter and Q represents the response from the module.

The driver routine transfers parameters to and from the BASIC interpreter, assembles the CAMAC command word, and writes it into the CAMAC command register in the Branch Driver. It assembles the CAMAC data word from the data parameter (in the write case) and starts command execution. It transfers data from the data register in the Branch Driver to BASIC (in the Read case), and also transfers the Q response. During these procedures the BASIC driver routine (in conjunction with the Branch Driver) performs a number of system checks. It first of all determines whether the Branch Highway is available (the Branch Driver has been designed to allow several others to be connected to the same Branch Highway). It checks on the validity of the CAMAC command codes. It checks that the Branch Timing B line (BTB) of the addressed crate is in the normal state. It checks for correct BTB response and also command accepted response (X). It also checks for data errors. (A parity line has been added in the Branch Highway.)

A similar driver routine was written for the magnetic tape unit.

6. SYSTEM CONTROL AND DATA COLLECTION

The performance of the fan has to be measured at 100 points on a hypothetical N,R response surface. The operation of the system control and data collection program is illustrated by the flow diagram in Fig. 3.

The system control strategy devised is responsible for the calculation of the stepping-motor control vector in order to control process output as desired. The process, which can be described as a two-input-two-output cross-coupled system, is highly nonlinear and its characteristics are different for different fans. The control strategy has been implemented in two control routines A and B, as shown in Fig. 3. For the purpose of routine A, the N,R response surface has been divided into a 7 × 7 matrix of elementary response surfaces. The stepping-motor setpoint values which will set the system within one of these elementary surfaces are stored in a 7 × 7 array. Control routine A looks up the required motor positions in this array, calculates the number of pulses to drive the stepping motors to the required setpoints, loads the CAMAC pulser with this data and initiates it. The initial values of the 7 × 7 motor setpoint array are determined by a separate BASIC program for each fan size using an interactive experimentation method. These values are adapted to variations in engine and fan characteristics at regular intervals as discussed below.

While the system is pulsed autonomously by the CAMAC pulsers to the vicinity of a new desired state, the data collection program sets the rest of the system up for data collection, i.e. it positions pressure probes and sets the solenoid valves. It then monitors N and R, and, when these are stable, compares them with the desired values. If reasonably close, it measures the 24 analog signals from the transducers and stores them together with the N and R values on the magnetic tape unit. This completes a measurement cycle.

The data collection program then repeats the abovementioned operations with the other two pressure probes (it also checks N and R, and adjusts if necessary). The whole procedure is also repeated at ten radial-pressure probe positions and 100 N,R points on the response surface.

If, in the above procedure, N and R are found not to be within the limits desired, some means must be used to get the system into exactly the specified state. This is done by control routine B (see Fig. 3). Several methods for doing this have been considered, e.g. EVOP[2], SSDEVOP and also certain search methods. A modified EVOP approach was finally selected.

In this approach, described in more detail in the next paragraph, a system steady-state identification experiment is conducted. From the results of this experiment the coefficients of a simple linear empirical model of the process, representing process behaviour in the current elementary response surface, are calculated. From the model and the desired values of N and R, new stepping-motor setpoints, which will drive the system closer to the desired state, are calculated.

Similarly, the setpoints of the stepping motors are calculated in order to set the process to the centre of the current elementary response surface. The latter results are then used to replace the corresponding values in the 7 × 7 stepping-motor setpoint array mentioned previously and thereby adapts this array to any possible changed fan or engine characteristics. In accordance with the first set of calculations, the motors are driven to the calculated positions. The system response is again measured (after steady state is reached) and compared to the desired state. If the response is close enough to the desired value, the computer continues the data collection program. If the response is not close enough to the desired value, the computer starts a new identification experiment around the current process state; however, this time it decreases the response surface area over which the identification experiment is conducted. This procedure is continued until the required process state is reached. In the second and subsequent identification experiments, the adaptation procedure of the 7 × 7 array as mentioned previously, is not performed since it is not necessary and may produce inferior results in a highly nonlinear system.

7. PROCESS IDENTIFICATION

The process steady-state mathematical model used is given by the following equations:

$$N = A_0 + A_1 M_n + A_2 M_r \qquad (1)$$

$$R = B_0 + B_1 M_n + B_2 M_r \qquad (2)$$

A_0, A_1, A_2, B_0, B_1, B_2 are empirical constants, and M_n and M_r are the speed and damper control stepping-motor positions. In order to simplify the calculation of model parameters from the identification results an orthogonal identification experiment was designed and the stepping-motor setpoint values were normalized[2]. The model, using normalized motor setpoints, is given by the following equations:

$$N = a_0 + a_1 X_1 + a_2 X_2 \qquad (3)$$

$$R = b_0 + b_1 X_1 + b_2 X_2 \qquad (4)$$

where X_1 and X_2 are normalized values of M_n and M_r respectively, calculated from the following equations:-

$$X_1 = \frac{M_n - M_{no}}{\frac{1}{2}\Delta_n} \qquad (5)$$

$$X_2 = \frac{M_r - M_{ro}}{\frac{1}{2}\Delta_r} \qquad (6)$$

M_{no} and M_{ro} are the centre points around which the identification experiment is conducted. Δ_n and Δ_r are the ranges between which the stepping-motor setpoints are adjusted.

In the orthogonal experimental design, the stepping-motor positions M_n and M_r have to be adjusted between two positions, the numerical values of which are chosen so as to make X_1 and X_2 equal to either +1 or -1.

Four runs (excluding run 0, the centre-point run which has already been done) are necessary for the identification experiment and are tabulated with the results as shown in the following Table:-

RUN	N	R	X_1	X_2
0	N_0	R_0	0	0
1	N_1	R_1	-1	-1
2	N_2	R_2	+1	-1
3	N_3	R_3	-1	+1
4	N_4	R_4	+1	+1

N_1 to N_4 and R_1 to R_4 are the results of the identification experiment.

The coefficients of the normalized process model are calculated, using the following equations:-

$$a_0 = \frac{1}{5}(N_0 + N_1 + N_2 + N_3 + N_4) \qquad (7)$$

$$a_1 = \frac{1}{4}\left[(N_2 + N_4) - (N_1 + N_3)\right] \qquad (8)$$

$$a_2 = \frac{1}{4}\left[(N_3 + N_4) - (N_1 + N_2)\right] \qquad (9)$$

Similarly:

$$b_0 = \frac{1}{5}(R_0 + R_1 + R_2 + R_3 + R_4) \qquad (10)$$

$$b_1 = \frac{1}{4}\left[(R_2 + R_4) - (R_1 + R_3)\right] \qquad (11)$$

$$b_2 = \frac{1}{4}\left[(R_3 + R_4) - (R_1 + R_2)\right] \qquad (12)$$

The values of the parameters in the nonnormalized model are given by

$$A_0 = a_0 - \frac{2a_1 M_{no}}{\Delta_n} - \frac{2a_2 M_{ro}}{\Delta_r} \qquad (13)$$

$$A_1 = \frac{2a_1}{\Delta_n} \qquad (14)$$

$$A_2 = \frac{2a_2}{\Delta_r} \qquad (15)$$

Similarly:

$$B_0 = b_0 - \frac{2b_1 M_{no}}{\Delta_n} - \frac{2b_2 M_{ro}}{\Delta_r} \qquad (16)$$

$$B_1 = \frac{2b_1}{\Delta_n} \qquad (17)$$

$$B_2 = \frac{2b_2}{\Delta_r} \qquad (18)$$

Calculation of the process model parameters from eq. (7) to eq. (18) completes the identification process.

8. CONCLUSIONS

A locally developed CAMAC system has been used with success as the instrumentation system for a computer-based axial flow fan evaluation system. The modular design of the CAMAC system and standard structure facilitated both hardware implementation and software developments.

REFERENCES

[1] BRYER, D.W., WALSKE, D.E., GARNER, H.C. Pressure probes selected for three dimentional flow measurement. Brit. Aeronaut. Research Council Repts. and Mem. 3037 (1958).

[2] MULAR, A.L. (University of British Columbia, Canada), BULL, W.R. (Colorado School of Mines, USA). Mineral processes: their analysis, optimization and control. Summer School Notes (1969-1973).

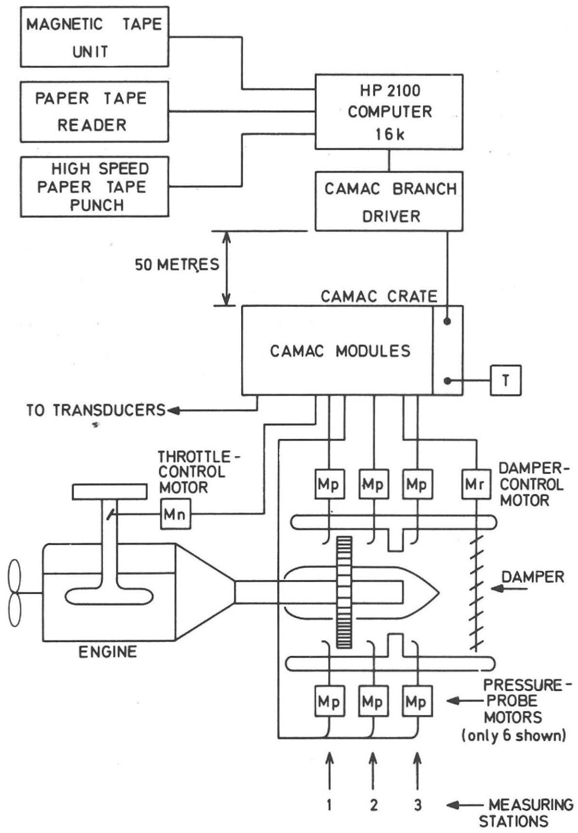


Fig. 1: Data acquisition and control system

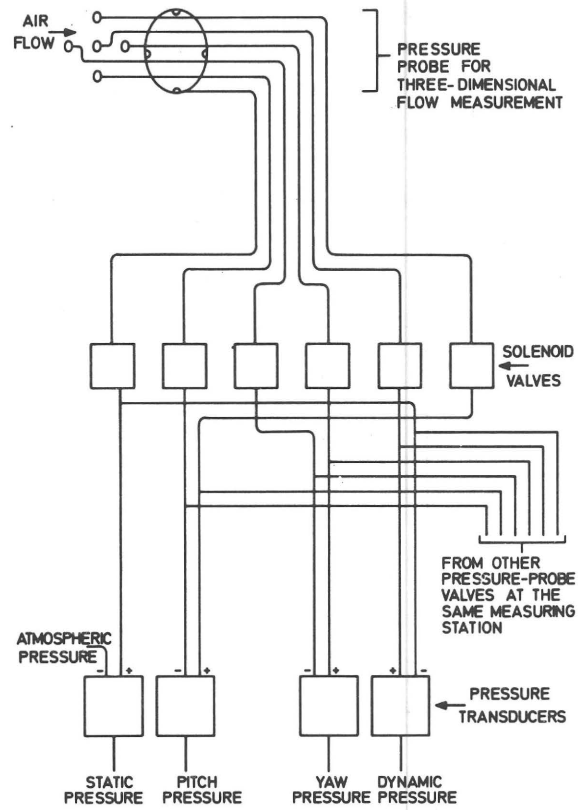


Fig. 2: Pressure-probe instrumentation

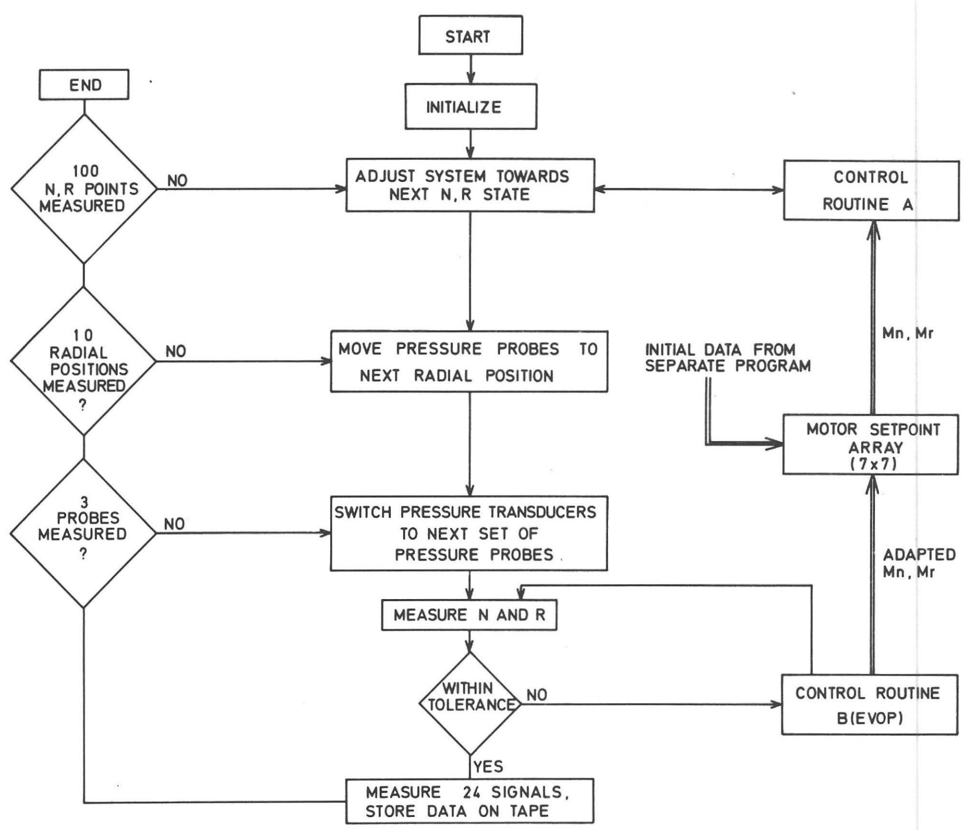


Fig. 3: Control and data acquisition flow diagram

EINE PROZESSRECHNERANLAGE MIT CAMAC-AUSRÜSTUNG FÜR DIE INGENIEUR-AUSBILDUNG

H. Wagner
Siemens AG, Karlsruhe, Germany

ZUSAMMENFASSUNG

Die Fachhochschule Dortmund erhält eine Prozessrechneranlage Siemens-System 300/330 mit einer CAMAC-Prozesseinheit, die über einen langen Branch an die Zentraleinheit angeschlossen wird. Neben der Beschreibung des Anlagenaufbaues, und hier besonders der CAMAC-Prozesseinheit, werden auch die Gründe dargelegt, warum eine Prozessrechneranlage mit CAMAC-Prozesseinheit an einer Fachhochschule eingesetzt wird.

ABSTRACT

The Technical High School at Dortmund has received a Siemens-System 300/330 process computer which has a CAMAC process unit attached by a long branch to the central unit. In addition to a description of the parts of the equipment, particularly the CAMAC process unit, there is an explanation of the reasons why a process computer with a CAMAC process unit should be installed in a technical high school.

RESUME

L'école technique supérieure de Dortmund a été dotée d'un calculateur de processus Siemens 300/330 comprenant une unité de traitement CAMAC reliée à l'unité centrale par l'intermédiaire d'une branche de grandes dimensions. Le présent exposé décrit la construction de l'installation et notamment l'unité de traitement; il donne en outre les raisons qui ont déterminé la mise en service d'un tel calculateur dans une école technique supérieure.

Die zunehmende Rationalisierung und Automatisierung macht es erforderlich, die Automatisierungsmittel zu standardisieren. Es ist unwirtschaftlich und unzweckmäßig, für jedes Problem eine eigene spezielle Automatik zu entwickeln. In faßt idealer Weise bietet sich hier der programmierbare Digitalrechner als universelles Automatisierungshilfsmittel an. Damit ist es möglich, die Automatisierungsmethoden zu vereinheitlichen und allgemeingültiger abzufassen. Das Hilfsmittel Digitalrechner, im technischen und technisch-wissenschaftlichen Bereich generell als Prozeßrechner bezeichnet, führt die unterschiedlichsten Ingenieurwissenschaften immer häufiger zusammen, um miteinander zu reden sowie Anforderungen, Ergebnisse und Erfahrungen auszutauschen.

Vor diesem Hintergrund sind die Bemühungen der technisch-wissenschaftlichen Ausbildungsstätten zu sehen, die Informatik als Hauptfach und den Digitalrechnern als Hilfsmittel für die anderen Disziplinen in die vorhandenen Ausbildungsgänge zu integrieren.

Es ist nicht möglich, jedem interessierten Fachbereich seine eigene Prozeßrechneranlage mit der entsprechenden Peripherie trotz sinkender Systemkosten zur Verfügung zu stellen, weil die Folgekosten (z.B. Unterhaltungskosten) und der Mangel an qualifiziertem Personal enge Grenzen setzen. Damit aber zwischen Wunsch und Wirklichkeit keine all zu große Diskrepanz entsteht, bemühen wir uns, ein System einzusetzen, daß den Forderungen der Ausbildungsstätte weitestgehend gerecht wird. Die wichtigsten Forderungen an das System sind:

- das System muß die Ausbildung einer großen Anzahl Studenten in relativ kurzer Zeit in den gängigen Programmiersprachen FORTRAN, BASIC und in der systemeigenen Assemblersprache ermöglichen. Dies muß im Time-sharing-Betrieb ablaufen.

- Zum Studium von technischen on-line-Problemen ist eine Prozeßeinheit erforderlich, die so angeordnet werden kann, daß jeder interessierte Fachbereich seine eigene Prozeßeinheit bekommt. Dies ist deshalb notwendig, weil die Labors in der Regel nicht in der Nähe des Rechnerraums liegen.

- Die Prozeßeinheit muß flexibel sein und sich schnell ändernden Forderungen ohne großen Projektierungsaufwand anpassen lassen. Das ist mit CAMAC gegeben.

- Bei einer solchen Anordnung muß die Arbeit in der Ausbildung und am Experiment durch umfangreiche System-Software unterstützt werden.

Das Bild 1 zeigt die projektierte Anlage für die Fachhochschule Dortmund. Sie besteht aus:

Siemens-Prozeßrechner 330,
ausgebaut bis 24 K-Worte,
max. Ausbau bis 64 K-Worte

Multiplexer für
Ein/Ausgabeblattschreiber 200 Bd
Lochkarteneingabe 500 Karten/min.
Lochstreifeneingabe 120 Zeichen/s
Lochstreifenausgabe 30 Zeichen/s und
Matrix-Zeilendrucker 200 Zeichen/s.

Plattenspeichereinheit mit 2 x 2,4 Mio Worte
Nettokapazität
CAMAC-Prozeßeinheit, bestehend aus

System-Controller 330 (SC 330)
4 CAMAC-Crates mit Crate-Controller,
Typ A1 und CAMAC-Differential-Branch-Extender zum dezentralen Aufbau der einzelnen CAMAC-Crates.
Branch-Termination
Diverse CAMAC-Module wie statische und dynamische Digital-Ein- und Ausgaben, ADC, Multiplexer, Stromgenerator, Zeitimpulsgeber und Kurzzeitwecker, DAC, Datenweganzeiger.

Der Aufbau eines langen Branch ist das hervorstechende Merkmal dieser Prozeßrechneranlage. Unter einem langen Branch verstehen wir eine Anordnung, wodurch Einsatz geeigneter Sender/Empfängerbausteine die Übertragungseigenschaften des Branch so verbessert werden, daß Kabellänge bis ca. 1 km möglich sind.

Zur Zeit sind die Labors der Fachbereiche Maschinenbau, Nachrichtentechnik und Energietechnik angeschlossen. Da die einzelnen Fachbereiche ihre Standardversuche erst im Laufe der Zeit aufbauen werden, steht noch nicht endgültig fest, an welchen Plätzen die dezentrale CAMAC-Prozeßeinheit überall benötigt wird. Dies bedeutet aber keine grundsätzliche Schwierigkeit. Der CAMAC-Rahmen ist in einem fahrbaren 19"-Schrank, 1 m hoch, untergebracht. Er kann leicht transportiert werden. Das Branch-Kabel erhält Anschlußstellen, die im Prinzip aus Bild 2 hervorgehen. Das Branch-Kabel wird dabei aufgetrennt und auf eine Seite eines 8 x 25-teiligen Lötösenstreifens (Fernmeldeverteiler) aufgelegt. Die andere Seite des Lötösenstreifens führt über ein flexibles Kabel zu einer Branch-Steckbuchse. Jede Trennstelle besteht letztlich aus zwei Branch-Steckbuchsen, die mit einem Branch-Kabel kurzgeschlossen oder mit einem Crate-Controller Typ A1 bzw. den Sender/Empfängerbausteinen eines CAMAC-Crate verbunden werden. Innerhalb eines Labors lassen sich mehrere Anschlußstellen anbringen bzw. nachrüsten, so daß die Verbindungen zwischen Versuchsaufbau und CAMAC-Prozeßeinheit immer kurz gehalten werden können.

Die CAMAC-Peripherie wird über den SC 330 an die Zentraleinheit angeschlossen. Der SC 330 regelt den Datenverkehr zwischen dem Prozeßrechner 330 und dem CAMAC-Branch-Highway und bestimmt daher wesentlich die Leistungsfähigkeit der angeschlossenen CAMAC-Prozeßeinheit. Er soll deshalb etwas näher beschrieben werden.

Der SC 330 ist modular aufgebaut. Er belegt 8 CAMAC-Breiten. Der Grundausbau enthält neben dem programmgesteuerten Betrieb eine Hardware-Interrupt Verarbeitung, die kurze Software-Reaktionszeiten ermöglicht. Über ein Statusregister kann zu jeder Zeit der aktuelle Zustand des SC 330 und der angeschlossenen Crates abgefragt werden.

Die Rechnerschnittstelle ist so eingerichtet, daß durch Einsatz mehrerer SC 330 ein Multi-Branch-System aufgebaut werden kann.

Das Bild 3 zeigt den funktionellen Aufbau des SC 330. Der SC 330 ist in 3 Funktionsgruppen gegliedert:

- Rechnerspezifischer Teil mit

  Sender/Empfängerbaugruppe
  Synchronisier-Schaltung
  Adressenerkennung

- Interne Logik mit

  Informationsregister
  CNAF-Register
  Alarmbearbeitung, Status-Register
  Blocklängen- und Adressregister für die Blockübertragung
  Inkrement-Mode
  Random-List-Mode

- Branch-Driver mit

  Sender/Empfänger-Baugruppe
  Abschlußwiderstände

Die Schreib- und Leseregister des SC 330 bestehen aus 2 Teilen zu jeweils 16 und 8 Bit Länge, so daß 16- und 24-Bit-Operationen möglich sind.

Der Alarm-Adress-Generator besteht aus einem Alarm-Basis-Adress-Register, einem Prioritätsnetzwerk und einem Maskenregister. Mit Hilfe des Maskenregisters können einzelne oder alle Alarme maskiert werden. Die Steuerung stößt aufgrund eines Sammelalarms zum frühest möglichen Zeitpunkt automatisch eine BG-Operation an. Die schnelle Alarmbehandlung durch den Alarmadressgenerator setzt in einem Branch mit mehr als 24 Alarmquellen einen LAM-Grader pro Crate voraus.

Für schnelle Daten-Ein/Ausgaben wird die Blocksteuerung eingesetzt. Der Datenverkehr wird hierbei im direkten Speicherzugriff ohne Zentralprozessorbeteiligung abgewickelt, so daß die Datenrate faßt ausschließlich vom Branch und den Modulen bestimmt wird. Die Adressenverwaltung (Speicheradresse und CAMAC-Adresse) und die Terminierung übernimmt der SC 330. Der Start kann per Programm oder über einen CAMAC-Alarm erfolgen. Das N- und A-Register ist jeweils als Zählregister ausgebildet und kann nach jeder Übertragung inkrementiert werden. Es können bis zu 6 Blocksteuerungen simultan arbeiten.

Der Inkrement-Mode dient zur Erfassung von Zählereignissen, Häufigkeitsverteilungen und ähnliches. Dabei wird ein zu verarbeitendes 16-Bit-Binärmuster als Zentralspeicheradresse interpretiert, diese Adresse gelesen, der Inhalt der Adresse um 1 erhöht und anschließend wieder unter dieser Adresse abgelegt.

Der Random-List-Mode ermöglicht die autonome Bearbeitung unterschiedlicher CAMAC-Befehle, ohne den Zentralprozessor zu belasten. Dabei wird aus einer Zentralspeicherliste der jeweilige CAMAC-Befehl und gegebenenfalls ein zugehöriges Datum über direkten Zentralspeicherzugriff gelesen und anschließend die Operation ausgeführt. Dies wird so lange fortgesetzt, bis das Listenende erreicht ist. Die Steuerung der Listenbearbeitung kann durch entsprechende Komandos erfolgen, die in der Liste angeordnet werden (z.B. Warten).

Die beiden letztgenannten Funktionen werden zur Zeit für das Projekt noch nicht eingesetzt. Da der SC 330 jedoch modular aufgebaut ist, kann jederzeit eine Nachrüstung erfolgen.

Die Entscheidung, eine dezentrale CAMAC-Prozeßeinheit für die Ingenieur-Ausbildung einzusetzen, ist auf folgende Kriterien zurückzuführen:

- Sie wird an einer modernen Prozeßrechneranlage betrieben, die auch über eine entsprechende Software für die Ingenieur-Ausbildung verfügt, wie z.B. das Mehrfachzugriffsystem 330 (MZS 330), das batch und on-line-Aufgaben für max. 32 Benutzer koordiniert.

- Der gewählte Systemaufbau mit der CAMAC-Prozeßeinheit ist äußerst flexibel und kann ohne großen Projektierungsaufwand anderen Aufgabenstellungen angepaßt werden.

- Jeder Fachbereich hat seine eigene Prozeßeinheit.

- Der SC 330 läßt einen komfortablen und schnellen Betrieb mit dem Prozeß zu.

- Die einzelnen Fachbereiche können die CAMAC-Module gemeinsam verwalten und verfügen so über ein großes Magazin verschiedenster Funktionen.

- Die Beschaffung weiterer Module beansprucht wenig finanzielle Mittel.

- Man ist in der Lage, selbst Module in Form von Ingenieurarbeiten entwickeln zu lassen, ohne sich dabei wegen der Schnittstellen oder anderer Bedingungen um Lizenzen oder Patente kümmern zu müssen.

- Die Frontplatten der Funktionsmodule sind anwenderfreundlich. Der Einsatz von Zwischenverteilern ist nicht notwendig.

- Die Verbindungen zwischen Experiment und CAMAC-Crate können wegen der verschiedenen Anschlußstellen immer kurz gehalten werden.

- Die Tatsache, daß am DMA-Bus mehrere SC 330 betrieben werden können, läßt für die Zukunft jede Erweiterung zu.

- Der Einsatz von CAMAC nimmt in der Industrie ständig zu, so daß nicht an der Wirklichkeit vorbeigearbeitet wird.

- Die Entwicklung der CAMAC-Technik ist noch nicht abgeschlossen und zeigt im Bereich Datenübertragung bzw. serieller Branch weiter interessante Perspektiven für den Ausbildungsbereich.

Das Programmieren von Experimenten gehört heute unbedingt zu einer praxisnahen Ingenieur-Ausbildung. Es bietet:

- Verknüpfung des reinen Programmierens mit der on-line-Datenverarbeitung.
- Darstellung der Diskrepanz zwischen Theorie und Praxis (Störquellen, Meßfehler, Auflösung usw.)
- Erkenntnisse der Unterschiede der analogen und der digitalen Meßtechnik
- Studieren der Einflüsse von Zeitbedingungen bei Automatisierungsaufgaben
- leichte Erfassung auch schneller Vorgänge zur Analyse des Dynamikverhaltens
- bequemes Aufnehmen umfangreicher Meßwertreihen und deren Archivierung
- Entlastung des Lernenden von routinemäßigen Auswertungen
- Auseinandersetzung mit den Real-time-Forderungen beim simultanen Ablauf mehrerer Programme

Die eingesetzte CAMAC-Prozeßeinheit, davon sind wir überzeugt, wird hier dem Studenten entscheidend mithelfen, das gesteckte Ziel zu erreichen. Er kann sich voll auf sein Problem konzentrieren und braucht nicht komplizierte und zeitaufwändige Rangierungen oder Verdrahtungen vorzunehmen, bis sein Experiment angeschlossen ist.

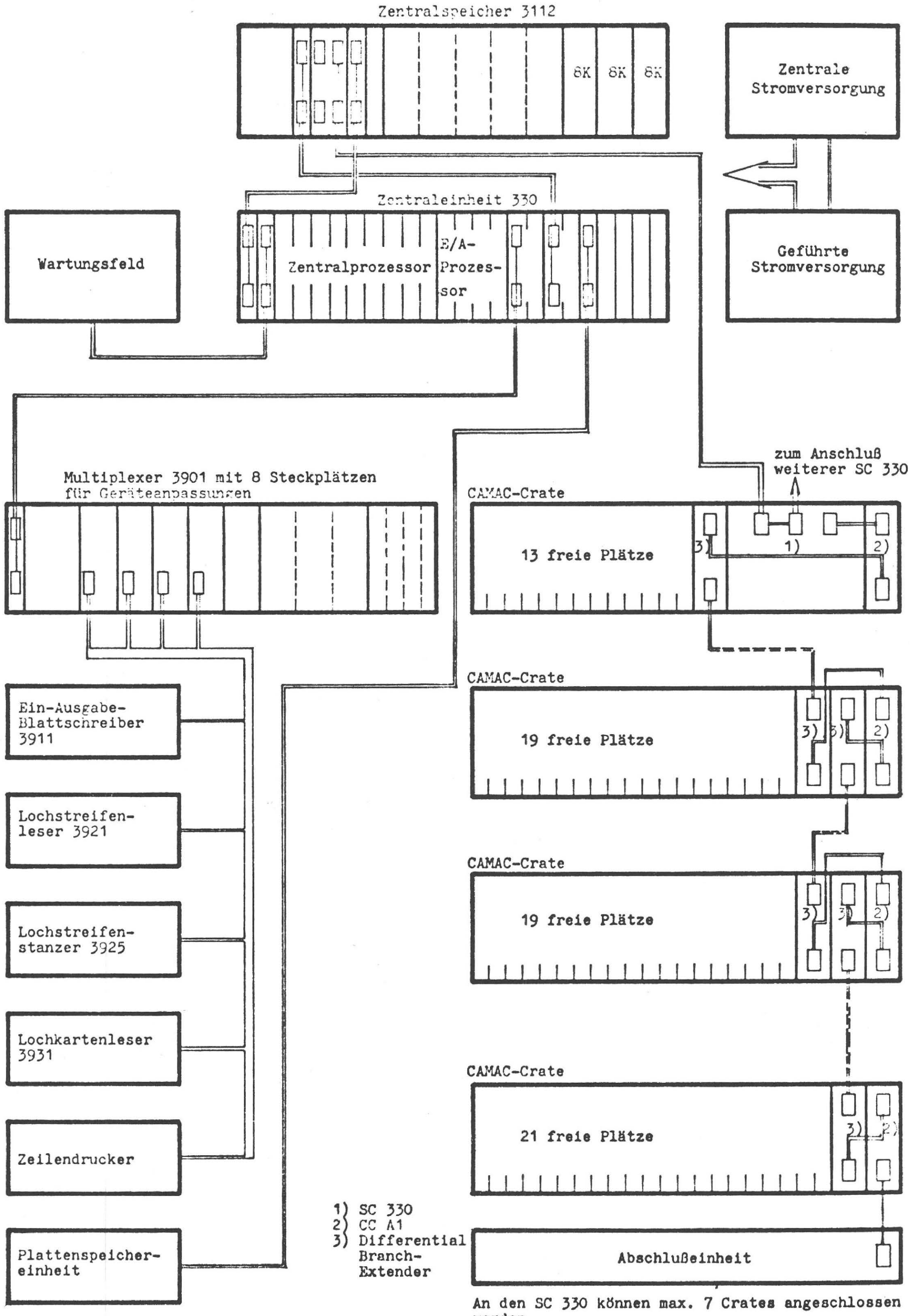


Bild 1 : Pr-Anlage der FHS Dortmund mit Standardperipherie und dezentraler CAMAC-Prozeß-einheit.

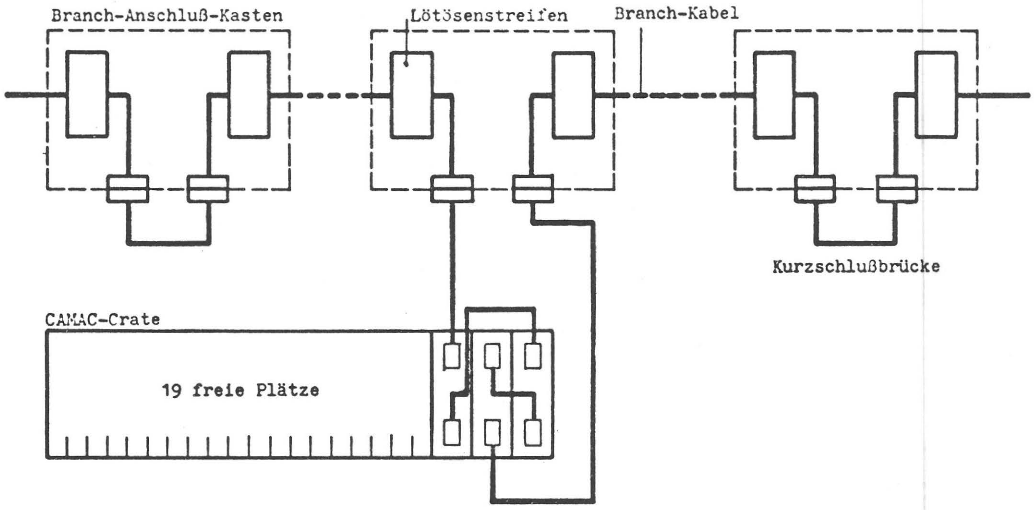


Bild 2 : Branch-Auschlussstellen

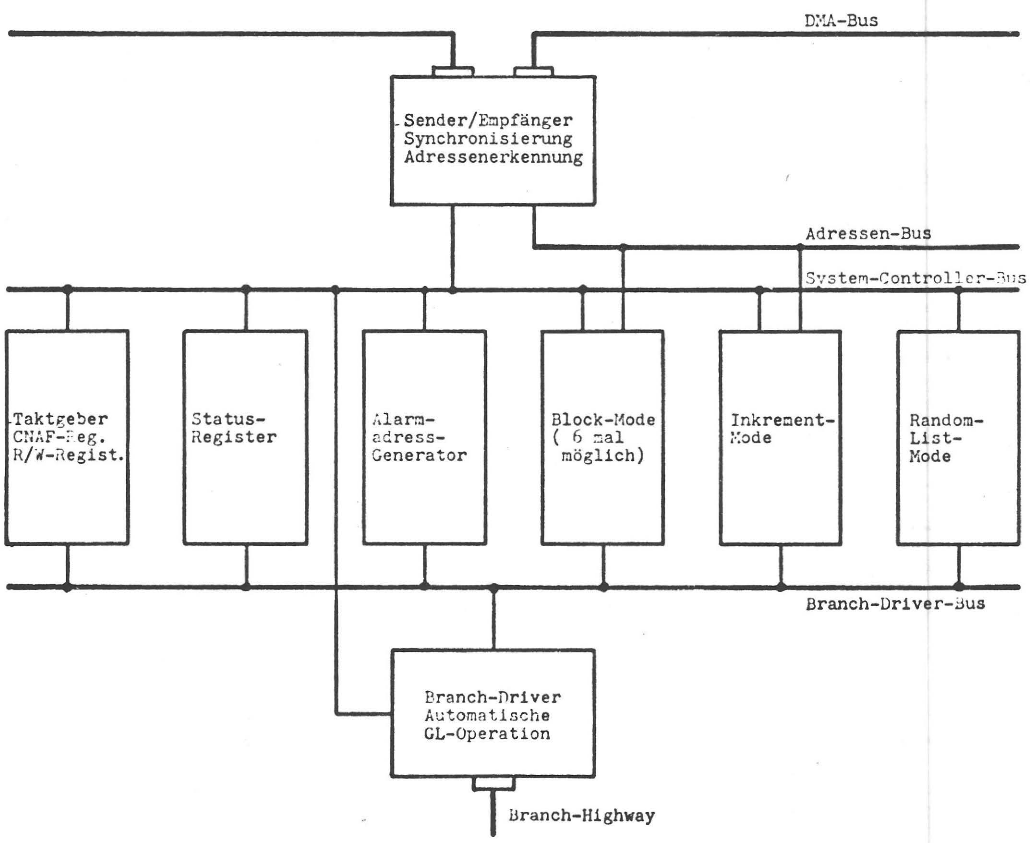


Bild 3 : SC 330 Blockschaltbild.

CONTROLE AUTOMATIQUE DE LA RADIOACTIVITE DES EFFLUENTS DE SACLAY

J. -P. Richard
Commissariat à l'Energie Atomique, Centre d'Etudes Nucléaires de Saclay,
Services d'Electronique de Saclay, France

RESUME

Les mesures de radioactivité provenant de dix stations de prélèvement automatique des effluents de Saclay sont centralisées dans un châssis CAMAC équipé d'un contrôleur simplifié le reliant à un miniordinateur MULTI-20.

L'ordinateur traite les alarmes s'il y a lieu et édite, après calcul, les bilans journaliers et mensuels.

L'entrée pour traitement de données provenant de mesures manuelles peut se faire par la télétype.

ZUSAMMENFASSUNG

Die Messergebnisse von 10 automatischen Abwasserprobeentnahmestationen werden in einem CAMAC-Rahmen zentralisiert.

Er ist mit einer vereinfachten Rahmensteuerung, die an einen Klein-Rechner MULTI-20 angeschlossen ist, ausgestattet.

Der Rechner bearbeitet die anstehenden Alarmsignale und erstellt nach Berechnung die Tages- und Monatsbilanz.

Die Dateneingabe von nicht-automatischen Ergebnissen kann mittels einer Teletype durchgeführt werden.

ABSTRACT

This system ensures measuring and recording of radioactive contents put down on a filter by evaporation of samples of Saclay's effluents. 10 identical measuring stations, RAME-10, are connected to an INTERTECHNIQUE MULTI-20 minicomputer via one CAMAC Crate with a dedicated Crate Controller.

Alpha, Beta, Beta background activities, count time, water sample volumes rejected water volumes are transmitted to the computer by several 4 x 16 bits scalers and 2 x 24 bits parallel input CAMAC modules.

The computer treates the alarms if any and edits the daily and monthly average calculated results.

1 - L'INSTALLATION.

L'installation décrite est destinée à assurer la modernisation de la station de contrôle des effluents du Centre d'Etudes Nucléaires de SACLAY.

Ce contrôle est assuré d'une part par des mesures continues de la radioactivité des eaux effectuées par des ensembles de pulvérisation et évaporation (RAM-10) et de comptage alpha beta (ES-20), et d'autre part par des mesures discontinues manuelles et hétérogènes.

Les différents traitements à faire subir aux informations brutes pour les rendre cohérentes et en extraire des bilans étaient effectués en différé par un gros ordinateur qui recevait ces informations sous forme de cartes, perforées à partir de bordereaux imprimés à la station de mesure.

L'ensemble rénové se compose d'une partie mesure qui a été conservée : RAM-10 - têtes de comptage alpha,beta, mouvement propre, débit, d'une interface dont le noyau est un châssis CAMAC et d'un miniordinateur MULTI-20 équipé de deux télétypes.

Les données sont mémorisées, classées et traitées en temps réel ; elles font l'objet d'une édition journalière et d'une édition mensuelle. En cas d'incident, l'impression des données à entrée automatique s'effectue cycle par cycle ; à la fin de l'incident, le bilan des activités moyennes et totales est imprimé.

2 - INTERFACE.-

Le problème posé par l'interface peut être résolu par trois approches différentes.

La première est de confier tout le travail de prise en compte des mesures à l'ordinateur et de se servir des coupleurs standards de celui-ci ; ceci conduit d'une part à une configuration inhabituelle du calculateur et d'autre part à une programmation assez délicate.

La seconde est de réaliser un automate "Hardware" chargé de condenser les résultats de chacun des cycles de mesure. Le calculateur est alors déchargé de la quasi totalité des tâches en temps réel mais le système électronique est alors complexe et unique en son genre, ce qui pose toujours des problèmes de maintenance.

La troisième est de confier à l'interface le soin d'intégrer les informations pendant un cycle tout en laissant à l'ordinateur la tâche de déterminer le début et la fin d'un cycle de mesure. La programmation est très allégée par rapport à la première solution et l'interface, moins particulière que dans la seconde solution, peut alors être réalisée en éléments standards.

La série, maintenant très riche, des tiroirs CAMAC a permis de retenir cette solution qui présente des avantages très importants :

1) La fiabilité du système est élevée: le MTBF des tiroirs industrialisés est grand et le temps d'immobilisation est très petit, car la substitution est possible.

2) L'étude de l'interface se réduit à quelques adaptations de signaux.

3) Les modules de programme permettant l'entrée des données existent, ce qui réduit le temps de programmation.

L'interface se présente en trois parties :

2.1 - Interface entre la prise de l'information et le châssis CAMAC.

Cette partie est nécessaire car les signaux très disparates issus des postes de prise de l'information doivent être normalisés.

Le châssis d'adaptation des signaux permet non seulement l'adaptation électrique des signaux, mais aussi une disposition pratique des prises de liaison. Il comporte un contrôle visuel des signaux émis par les stations ainsi qu'un générateur de test pouvant simuler une voie.

2.2 Le dispositif CAMAC, qui comporte:

- 10 JEB-20 ; 1 JEB-20 par voie permettant le comptage des alpha, beta, MP.
- 3 JEB-20 intégrant les impulsions de débit des 10 voies.

  (JEB-20 : 4 échelles binaires aveugles de 16 bits associables 2 par 2; fréquence maximale : 30MHz.)

- 1 JRE permettant la prise en compte des états des RAM-10.
- 1 JRE permettant la prise en compte des états des balises.

  (JRE : registres de configuration 2 × 24 bits. Permet sur un signal extérieur la mémorisation des états présents à l'entrée et l'appel du contrôleur de châssis).

- 1 JMP-10, permettant la mise en marche et l'arrêt de la télétype pour l'édition des alarmes.

.../...

2.3 - Interface CAMAC-ORDINATEUR.

Un contrôleur de châssis JCM8-30, qui est aussi un élément standard industrialisé, assure la liaison entre le châssis CAMAC et l'ordinateur MULTI-20.

Le JCM8-30 effectue la liaison MULTI-20 au niveau du bus standard à 1 châssis CAMAC. Il fonctionne soit en mode programmé, soit en mode entrée-sortie simultanées.

La figure (1) présente le synoptique général de l'installation, la figure (2) l'ensemble de l'interface.

3 - PROGRAMMATION.-

Le programme se compose de 6 parties.

3.1 Un squelette composé de fichiers implantés en mémoire centrale.

Il contient des informations de plus en plus élaborées et condensées, ainsi, le premier fichier contient les informations brutes copiées dans les registres des tiroirs CAMAC, le dernier fichier l'historique des 40 jours de mesure. La liaison entre ces fichiers.est assurée par des tâches dont la complexité varie de la simple copie au calcul d'activité corrigé de l'activité des éléments à période courte.

3.2 Des modules d'édition qui assurent sous forme claire, la présentation :

- des bilans journaliers,
- des bilans mensuels,
- des bilans incidents, si des alarmes apparaissent.

3.3 Un module de gestion des interruptions externes qui permet l'entrée des informations issues des registres des tiroirs CAMAC, ainsi que l'interprétation des états des RAM-10 (déplacement, pulvérisation, mode).

3.4 Un module de gestion des interruptions horloge qui tient à jour un calendrier et gère un échéancier des tâches à effectuer.

3.5 Un module de dialogue assurant à l'opérateur :

- un accès aisé aux différents fichiers,
- la possibilité d'utiliser les moyens de calcul du programme pour traiter des mesures discontinues relevées manuellement.

3.6 Un superviseur détermine l'exécution ou la mise en attente des tâches demandées par les autres modules suivant les priorités qui leur sont attachées.

L'encombrement mémoire de ce programme est de 16K octets, y compris les octets occupés par les fichiers et a nécessité 6 mois programmeur d'analyse de programmation et de mise au point.

SYNOPTIQUE GENERAL DE L'INSTALLATION

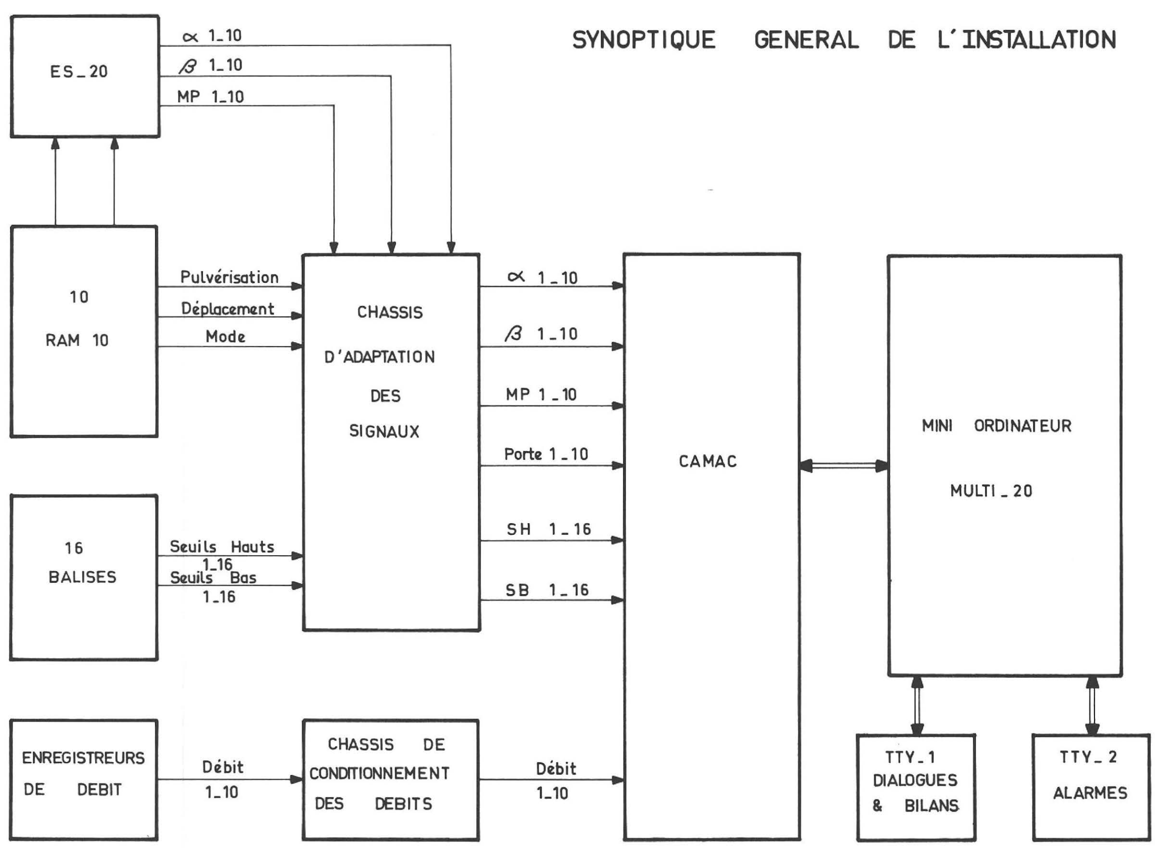


Fig. 1

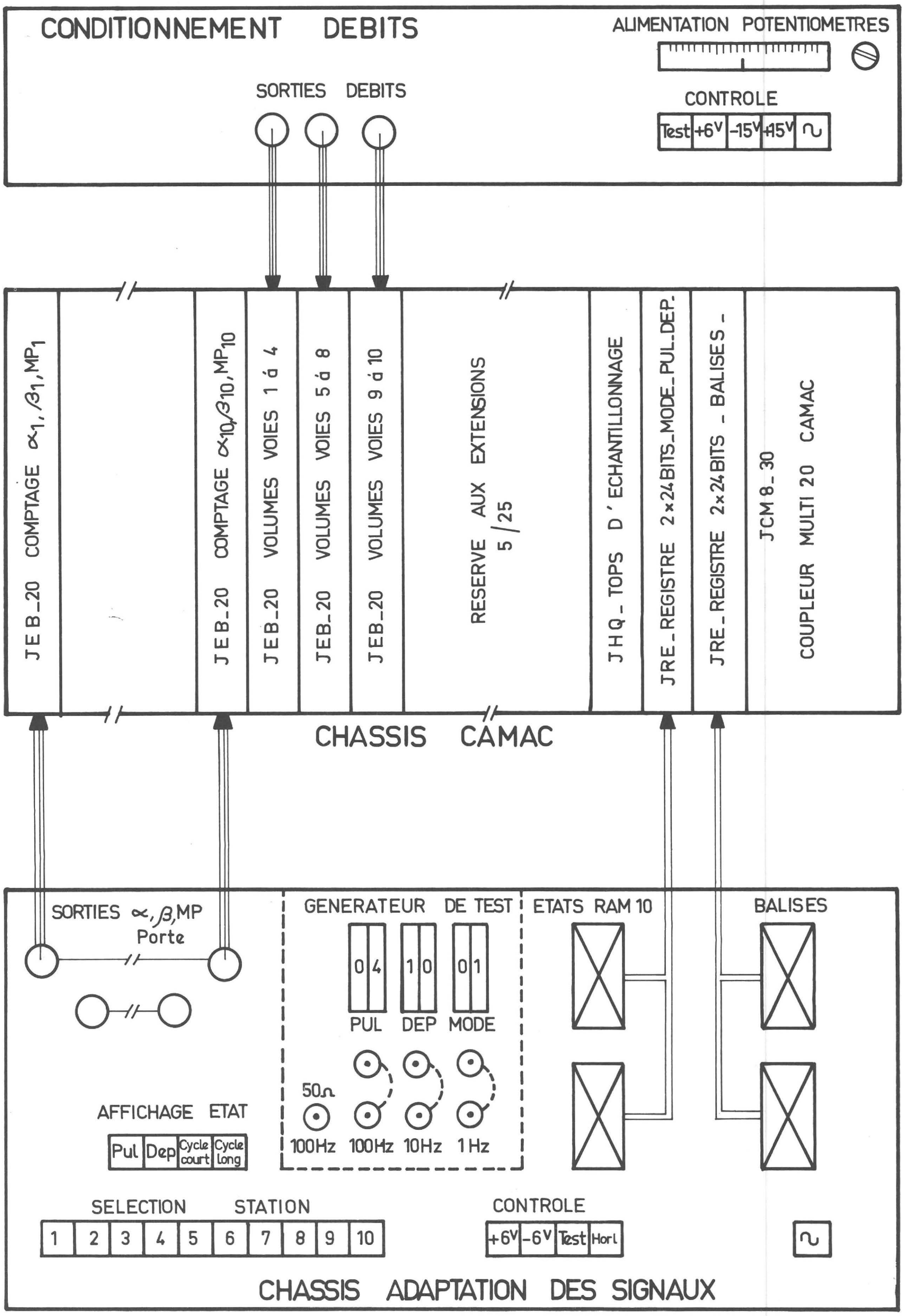


FIGURE 2

INTERFACE

AN ASTRONOMICAL APPLICATION OF CAMAC

I. G. van Breda, P. W. Hill, R. J. Campbell,
A. E. Lynas-Gray, D. M. Carr
University Observatory, St Andrews, England

ABSTRACT

A single-crate CAMAC system is described. The system is interfaced to a laboratory microdensitometer, for the measurement of photographic plates, and to a photoelectric photometer mounted on a telescope, for on-line measurement of the light fluxes from stars. Data may be recorded on magnetic tape for processing on a large computer.

ZUSAMMENFASSUNG

Es wird ein CAMAC-System mit nur einem Rahmen beschrieben. Das System ist an ein Labor-Mikrodensitometer zur Auswertung photographischer Platten und an ein photoelektrisches Photometer angeschlossen, das auf einem Teleskop montiert ist und zur On-line-Messung der Lichtstärke von Sternen dient. Die Daten können für die Verarbeitung in einem Grossrechner auf Magnetband aufgezeichnet werden.

RESUME

Description d'un système CAMAC monochâssis. Le système est relié à un microdensimètre de laboratoire, pour la mesure des plaques photographiques, et à un photomètre photoélectrique monté sur un télescope, pour la mesure en ligne de l'intensité lumineuse des étoiles. Les données peuvent être enregistrées sur bande magnétique en vue de leur traitement à l'aide d'un gros ordinateur.

1. INTRODUCTION

This paper describes the CAMAC system currently in operation at the University Observatory in St Andrews. CAMAC is used in data acquisition and control for two instruments, a Joyce-Loebl microdensitometer and a photoelectric photometer. The microdensitometer, which is housed in the same laboratory as the computer, is used to analyse photographic plates after they have been exposed at night on a telescope and then developed during the daytime. The photometer, which is mounted on a telescope situated in a dome some 50 metres from the computer, is on-line while the actual observations are being made at the telescope, and results may be presented to the astronomer during an observing session, enabling him to modify the observing sequence if the results show that this is needed.

2. HARDWARE

The laboratory arrangement is shown in Fig. 1. The computer system comprises a 316 Honeywell CPU with 12K words of core store, teletype, paper tape reader and punch and a 9-track magnetic tape transport. Data from both instruments may be recorded on magnetic tape for subsequent processing on the IBM 360/44 computer of the University Computing Laboratory. A single CAMAC crate is interfaced to the H 316 through a type 7022 controller connected to the computer I/O bus. A small Vero crate is located next to the CAMAC crate for auxiliary circuits that do not require the CAMAC dataway. The microdensitometer is shown at the right in Fig. 1 with a hardware control panel added for user convenience. A second Vero crate is housed underneath the control panel for power supplies, stepping motor control and interrupt-generating logic.

The dome equipment is shown in Fig. 2 with the photometer mounted on the telescope. A Tektronix 4010 graphic computer terminal is used to interact with the computer.

2.1 Microdensitometer. Normally astronomical photographs are in the form of negatives: the more blackened an area of a plate the brighter is the corresponding area of the image. It is possible, by careful calibration, to convert from the degree of blackening, or more precisely photographic density, to intensity of the image. The microdensitometer enables this to be done by measuring the average density over a small area of plate, as a function of position. Thus its output may be converted into a map of intensity of the image. The image may be one-dimensional, as in a spectrum, or two-dimensional as in the photograph of a galaxy shown in Fig. 3. By taking two photographs through different coloured filters, intensity maps may be obtained for two wavelength regions of the spectrum and the colour of the galaxy may then be calculated at each point on the map. This provides more astrophysically useful information than a simple intensity map.

A block diagram of the CAMAC interface for the microdensitometer is shown in Fig. 4. In the system described here, the distribution of density over a defined rectangular area of plate may be recorded by using a television-type raster scan of the area. The carriage supporting the photographic plate is moved underneath a measuring microscope, in rectangular coordinates, by two stepping motors which are controlled by a delayed pulse generator. The phase of each motor is fed as a two-bit binary word to an input gate so that a check can be made on correct operation of the motors. A check is also kept on the power supply to the stepping motors and an interrupt is generated if a power failure is detected. The power supply is also switched off by limit switches if the carriage is moved too far in any direction.

Certain of the control panel switches may generate an interrupt to initiate action. The state of the switches is fed to the input gate along with the wedge encoder output of the microdensitometer which gives the current density reading as a 10-bit Gray code.

2.2 Photoelectric Photometer. The photometer is used for measuring the strength of the H-beta Balmer line in hot stars. When the light enters the photometer it is passed through a beamsplitter which directs the light to two photomultipliers. A narrow band filter centred on H-beta is placed in one beam and a wider filter of the same centre wavelength is placed in the other. The spectrum line affects the narrow band reading relatively much more than it does the wide band and the ratio of the readings may be used as a measure of the strength of the line. A great advantage of this technique is that it may be used in poor observing conditions, since variations in atmospheric extinction, caused, for example, by light cloud, will have the same relative effect on the two readings. The method may be applied to other spectrum lines of interest.

A block diagram of the photometer is shown in Fig. 5. The intensity of light incident on the photomultiplier cathodes is measured by photon (pulse) counting. The signals from the photomultiplier anodes are passed through SSR Model 1120 amplifier/discriminators, mounted on the photometer head itself to minimize interference pick-up in the anode leads. The outputs from the discriminators are passed to 50-ohm line drivers and thence along co-axial cables to the laboratory. To avoid earthing problems between the two buildings, the signals are passed through optical isolators which convert to TTL signal levels. The pulses are counted in two stages of a Nuclear Enterprises type 709 24-bit quadscaler.

A Tekronix 4010 terminal is also located in the dome and is connected to a teletype interface running at 9.6 kilobaud. The 20 mA signals of the module are converted to and from 50 ohms for remote communication with the dome. Again optical isolators are used at the receiving ends of the lines.

The telescope is controlled from a moveable access platform which is not large enough to accommodate the Tektronix terminal. To enable the astronomer to control the course of an observation while at the telescope, two buttons are provided on the photometer to start and clear observations, together with two status switches. The start and clear switches each generate a lower-case ASCII character, to distinguish it from a keyboard character, which also contains the status information. The character is passed to the terminal minibus, serialized by the data communications interface and passed to the CAMAC teletype interface.

The heart of the system is the quadscaler.

Register A(0) is used as a Universal Time clock and counts 1 Hz pulses. Registers A(1) and A(2) are used to count pulses from the two photomultipliers. Register A(3) is a timer and is fed by 1 kHz pulses. It is preset at the start of a measurement to the complement of the integration time in milliseconds, and overflows at the end of the count period. The overflow, which is available as a signal level at the front panel, terminates the count in A(1) and A(2) and raises a LAM signal to indicate to the computer that the count is complete.

3. SOFTWARE

At the present time, the microdensitometer and photometer have separate programs and cannot be operated simultaneously. As microdensitometer usage grows, it will be necessary to merge the programs into a single interrupt-driven system.

3.1 Microdensitometer. The microdensitometer program has been written in DAP-16 Mod 2 Assembly Language and is interrupt-driven. Extensive use has been made of the Honeywell MAC Macro Preprocessor in conjunction with conditional assembly facilities to enable CAMAC operations to be specified in a simple and flexible manner. The program is modular in structure. Tasks are entered by operating control buttons on the front panel, by typing in commands on the teletype, or by operating the computer start button.

In order to execute a scan, the user first positions the photographic plate so that the object to be measured is located under the measuring microscope. This is done by choosing one of eight stepping directions by means of four direction switches on the control panel, and then activating fast, slow or single step control buttons, thereby causing an interrupt. For the fast and slow buttons, a task is entered that generates a stream of pulses at a preset rate from the delayed pulse generator, which also controls the mode of operation of both motors simultaneously by means of static output levels. The pulse stream is terminated when the control button is released, as determined by reading the contents of the input gate after each pulse. With the single step button, only one step is executed and the button must be released and re-activated for each subsequent step.

The rectangular area to be scanned is set up using the 'Frame' switch. The carriage is moved so that each of three of the corners of the area are centred in turn in the microscope, and the 'Frame' switch is pressed on each occasion, causing entry to a task that sets up the scan parameters. If 'Frame' is pressed a fourth time, the scan parameters are printed on the teletype and the carriage is moved around the perimeter of the area so that the user may see the size and position of the scan area on the plate, before starting a measurement.

The sizes of the sampling mesh in X and Y may be set up on the teletype, i.e. the number of steps between samples. These will depend upon the gearing used and the size of the sampling aperture of the microdensitometer as projected on to the photographic plate. Finally, a scan may be initiated by activating the 'Scan' switch, causing entry to the appropriate task.

The speed at which a scan can be made depends upon the method of measurement. In the Joyce-Loebl microdensitometer a linear density wedge is moved by a suitably damped servo-motor until the intensity of the beam of light passing through the photographic plate and a separate comparison beam passing through the wedge match each other. The position of the wedge gives a measure of density of the plate at that point. At each sampling point the wedge code is read at intervals of 15 mS, after an initial wait of 25 mS. When successive readings via the input gate agree within one bit, the wedge is taken to have settled and the reading is recorded for that sample point. Data is passed in blocks of up to 512 words to the magnetic tape output.

Since an important feature of work with the microdensitometer is to be able to compare plates of the same object taken through different filters at different times, reference co-ordinates must be set up for each scan using selected stars near the object. The first time a measurement is made, the positions of these stars are measured with respect to the scan frame. Each star is positioned in turn under the measuring microscope and the 'Centre' switch is activated. This causes a task to be entered that traverses the image in perpendicular directions (see Fig. 3) to find the centre of the star image. When the same object is measured again on subsequent occasions, the same stars are also measured, and the user is provided with information to align the scan so that it covers the same area of sky in the same manner.

A scan may be stopped temporarily by activating the 'Hold' switch. It is restarted by pressing 'Hold' a second time. It may also be suspended and the program dumped out on magnetic tape, for a short while, if the computer is required for another purpose. In addition, the program may be terminated by the 'Abort' control button which causes a pseudo stepping motor power failure by switching off the motor power supply.

3.2 Photometer. Because of the greater amount of data processing needed, the photometer program has been written in Fortran and uses polling techniques. To access CAMAC, a subroutine was written in Assembly Language. The call is of the form given in (1):

CALL CAMAC(IC,IN,IA,IF,IQ,ID),

where ID is either a two-dimensional integer array, or a real variable. The routine was also adapted for use with BASIC. Although slow in operation, this was found to be very useful for testing procedures from the teletype keyboard before incorporating them in Fortran or Assembler routines.

The photometer program is initialized on the teletype in the laboratory. The output device (magnetic tape unit or high speed paper tape punch) is selected, and the current Universal Time is loaded into the A(0) register of the quadscaler. Control is then transferred to the Tektronix 4010 terminal in the dome from where subsequent control of the program may be effected.

The astronomer may choose the integration time, and whether a measurement is to be made on a star or on the dark count rate of the photomultipliers. When making measurements on a star, the sky background must also be measured, since the diaphragm located in the focal plane of the telescope (to isolate the star image being measured from other star images) also passes some light from the sky background. The background is measured by displacing the telescope slightly and a measurement is made on clear sky without star images.

Integrations may be started by the start switch on the photometer head or from the terminal. Each integration is split into a series of sub-counts which are displayed on the terminal for both channels. At the end of each observation the average star counts in the two channels, corrected for sky background, are displayed. The average counts may be corrected for pulse overlap, and the ratio of the signals through the two filters expressed on the magnitude scale, a logarithmic scale used in astronomy for specifying the brightnesses of stars. The resulting figure is called the beta index (2).

The quality of the observations may be assessed by examining the consistency of the sub-counts and by repeating complete integrations to obtain more than one value for the beta index. At the end of a complete integration, the observer may decide whether the data is to be recorded on the chosen output medium or rejected. If accepted, a summary of the observation is printed on the laboratory teletype as an insurance against loss of data.

Acknowledgement. This work has been funded jointly by the University of St Andrews and the Science Research Council.

REFERENCES

(1) STEPHENS, C.L., VAN BREDA, I.G. The Use of CAMAC for Telescope Instrumentation. Proceedings of ESO/CERN Conference on Auxiliary Instrumentation for Large Telescopes, eds. Laustsen, S., Reiz, A., (1972), 499.

(2) CRAWFORD, D.L. Two-dimensional Spectral Classification by Narrow-band Photometry for B Stars in Clusters and Associations. Astrophys. J., 128, (1958), 185.

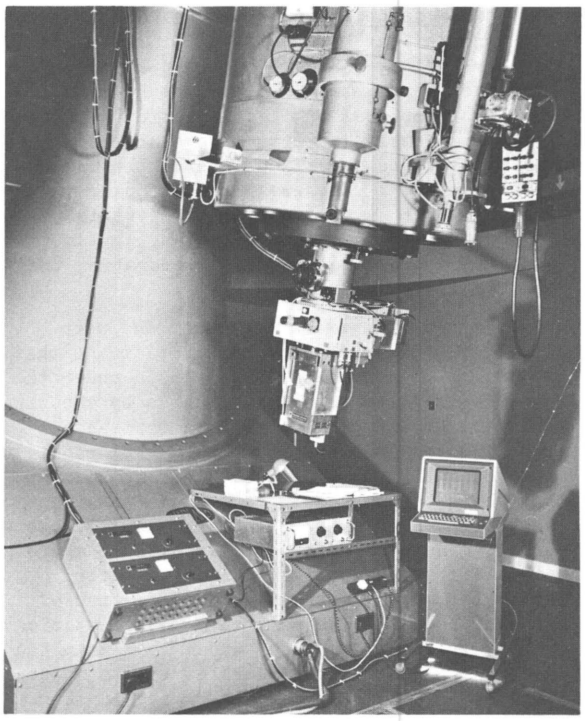


Fig. 2. Photoelectric Photometer Mounted on Telescope.

Fig. 1. Laboratory System.

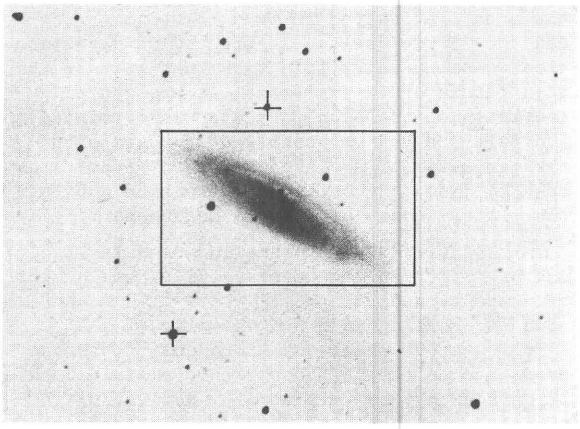


Fig. 3. Negative Exposure of a Galaxy. (Taken by C.W. Fraser)

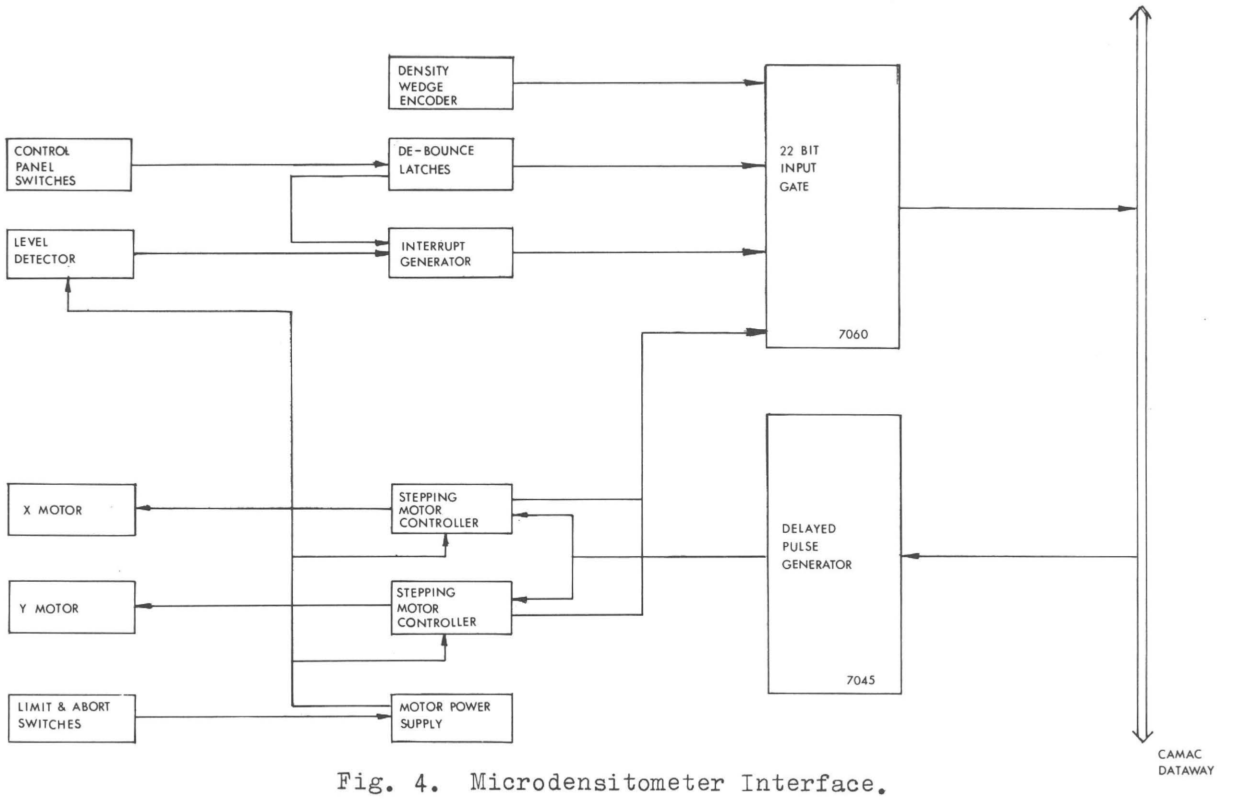


Fig. 4. Microdensitometer Interface.

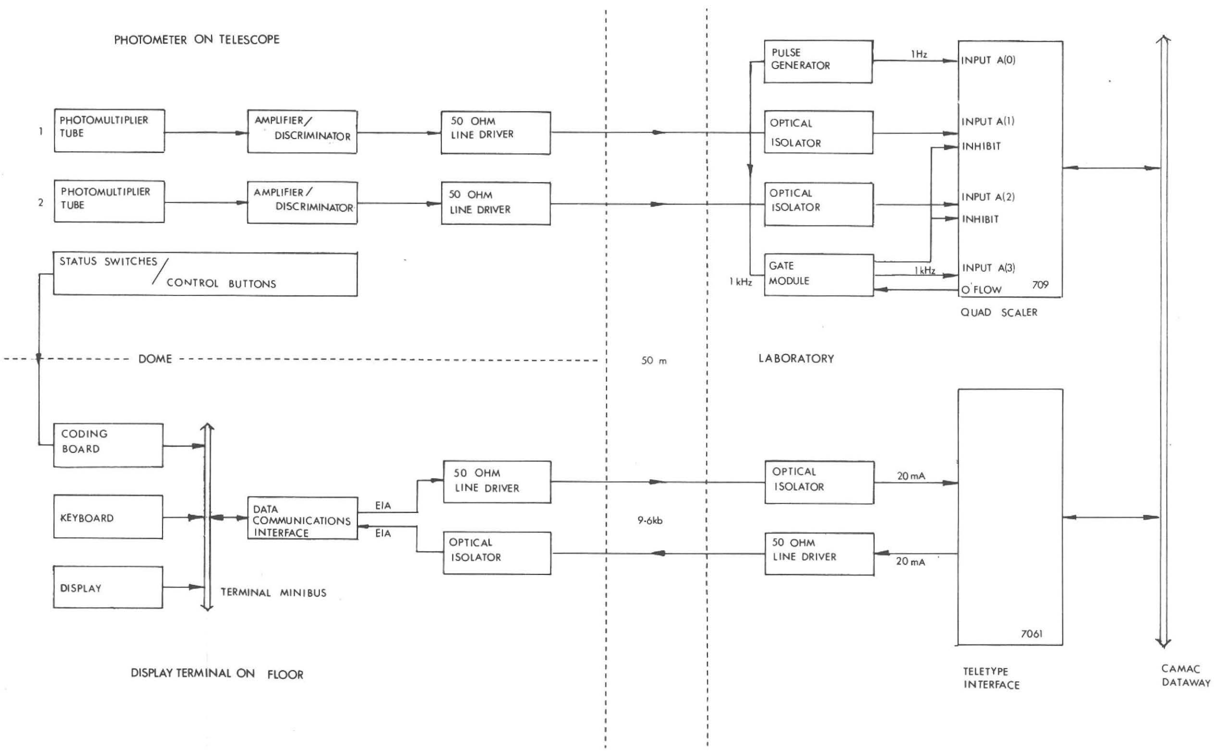


Fig. 5. Photoelectric Photometer Schematic.

COMPUTER CONTROLLED MEASUREMENT OF SEMICONDUCTOR DEVICES

P. Abend, H. Becker, D. Bräunig, R. Bublitz, G. Herdam,
A. Spencker, H. G. Wagemann, W. Wawer
Hahn-Meitner-Institut für Kernforschung GmbH, Berlin, Germany

ABSTRACT

A computer controlled measuring equipment for fully automatic measurements in on-line data processing is described. It is used for measuring semiconductor devices such as transistors and Zener diodes, as well as resistors. The integration of commercial high precision measuring instruments is shown and how data transfers between the measuring instruments and the on-line-computer via the CAMAC-system is accomplished. In addition to data acquisition the computer supervises and controls the continuous measuring sequence.

ZUSAMMENFASSUNG

Es wird eine rechnergeführte Messelektronik zur vollständigen Automatisierung von Messungen in der on-line-Datenverarbeitung beschrieben, mit der Halbleiterbauelemente wie Transistoren und Zenerdioden sowie ohmsche Widerstände gemessen werden. Der Einsatz digital arbeitender, kommerzieller Präzisions-Messgeräte und der Datenverkehr dieser Messinstrumente mit einem Prozessrechner über das CAMAC-System wird beschrieben. Dabei übernimmt der Rechner neben der Datensammlung die Aufgaben der Überwachung und Steuerung des Messablaufs.

RESUME

Description d'un appareillage de mesure commandé par ordinateur et utilisé pour les mesures entièrement automatiques dans le traitement en ligne des données. Cet appareillage est utilisé pour les mesures des semi-conducteurs, par ex. transistors, diodes Zener et résistances ohmiques. On décrit la mise en oeuvre d'instruments de mesure de haute précision disponibles dans le commerce ainsi que l'échange d'informations entre les instruments de mesure et l'ordinateur à l'aide d'un système CAMAC. En plus de l'acquisition des données, l'ordinateur assure la supervision et le contrôle de la séquence de mesure en continu.

1. Introduction

The testing of large numbers of semiconductor devices for the purpose of fabrication control, selection and sorting, or for reliability and length of life tests, is usually carried out with the aid of automatic testing devices. These automatic devices are usually designed for testing particular semiconductor devices and measure particular characteristic parameters at particular working points specified in advance. These data are insufficient for the accurate assessment of a semiconductor device, especially in reliability tests. What are needed complete characteristics or sets of characteristics which are obtained by time-consuming individual measurements, usually in analog form (X, Y plotter graphs).

This paper describes a measuring system capable of achieving complete automation of such measurements without sacrificing the accuracy and flexibility of analog individual measurements. Analog quantities are measured by commercially available digitally working instruments controlled by a computer, and are preset by current, voltage or function generators. These devices are controlled by an on-line computer capable of carrying out measurements according to a flexible program. The computer monitors all the instruments and experiments parameters, stores the data and, if necessary, can evaluate these data immediately and display the results continuously on a visual display screen for direct supervision of a run of measurements. The measuring instruments and the generators are coupled to the computer by units of the CAMAC (Computer Application of Measurement and Control) system. This modular system is internationally standardised and is manufactured commercially in Europe and the USA. It makes it possible to carry out any processes or experiments regardless of the organisation and structure of particular computers. This system is described below on the example of an irradiation resistance test of semiconductor devices which was carried out for the France-German satellite project "Symphonie".

2. The particular measurement task

The irradiation resistance test was to be carried out on various components including 10 specimens each of 7 different types of bipolar transistors, 3 types of Zener diodes, and carbon mass resistors. The corpuscular radiation in the satellite orbit was simulated by 1,5 MeV electron radiation. Three series of measurements were carried out on each specimen-one series before irradiation, one series after a first test irradiation with 10^{14} electrons per cm^2, and one series after a second test irradiation with 10^{15} electrons per cm^2 - so that the total amounted to about 780 measurement series with about 16000 measurement points on average. In measuring the forward characteristics of transistors the average measuring time required was 1 second per measurement point. This average measuring time includes the setting time for the parameters, the conversion times, the time required for measuring point commutation, and the data output on the visual display screen.

2.1 Transistors

Fig. 1 shows the basic circuit used for measuring the forward characteristics of transistors, the base current I_B and the collector current I_C as a function of the base-to-emitter voltage U_{BE}, and the collector leakage current I_{CBO} in function of the collector-to-base voltage U_{CB}. The voltage U_{BE} was varied in steps of 25 mV and the currents were measured in the range from 10 pA to 20 mA. The leakage current was measured at a voltage U_{CB} which varied from one transistor type to another. The measurement temperature was held constant at $(25 \pm 0{,}1)^{\circ}$ C.

Fig. 2 shows a semi-logarithmic plot of the forward characteristic of a transistor produced by a digital plotter controlled by the computer. The variation of the base current with the voltage U_{BE} provides detailed information on the physical properties of the individual transistor tested and on its resistance to irradiation [1-4]. From the characteristics it is possible to calculate the DC amplification as a function of the collector current I_C.

2.2 Zener diodes and resistors

Fig. 3 shows the basic circuit used for measuring the reverse characteristic of Zener diodes in the current range from 100 nA to 4 mA at three temperatures (-55° C, 25° C and 100°C). In these measurements the current I is impressed and the resulting Zener voltage U is measured with an accuracy of 0,1 mV. The same basic circuit is used for testing resistors.

The reverse characteristics recorded for each diode at the three test temperatures provide information on the reverse current specified for the diode, the Zener voltage, the mean temperature coefficient of the Zener voltage and the break-down properties of the diode [6, 7]. The irradiation-induced changes in the electrical properties of the diodes studied in these experiments were extremely slight.

The resistors showed no irradiation-induced changes.

3. Measuring instrumentation

The measuring instrumentation is shown in the block diagram Fig. 4. It consists of the on-line computer with the system control unit and of the following three instrument groups:

a) CAMAC instrumentation,
b) Control units and level commutators,
c) Precision measuring instruments and generators.

The data flow between the computer and the measuring electronic circuits, and

the monitoring of the latter by the computer, take place via the CAMAC system [8, 9].
As there are as yet no CAMAC instruments for measuring analog quantities with the sensitivity and accuracy required for the tests described in this paper, other high precision, commercially available measuring instruments were used for these experiments. These instruments have as a rule a data interface incompatible with the CAMAC system and therefore necessitate for connection with the CAMAC system the control units and level commutators shown in Fig. 4.
The following high precision instruments were used for measuring and generating analog quantities:

1. Two ammeters with a current range from 10^{-12} to 2×10^{-2} A and a very low input resistance.
2. One voltmeter with a range from 100 μV to 1000 V and a high input resistance.
3. One constant current-constant voltage generator for a current range from 10^{-7} to 1 A and a voltage range from 1 mV to 1000 V.
4. A thermostat chamber for a temperature range from -73,3 °C to 200 °C and a measuring position commutator for 100 measuring positions.

These instruments can be combined for a wide variety of current and voltage measurements over and above the special measuring tasks discussed in this paper. The measuring instruments are digitally integrating ammeters and voltmeters with automatic ranging. After the end of each conversion the data are transferred in parallel to the computer.

4. Measurement sequence

Fig. 5 shows the flow chart for the measurement of forward characteristics of transistors.
After the start of the program the CAMAC instrumentation is reset to zero, all external instruments are checked for their starting conditions, and any setting errors are corrected as directed by the computer. Then, via a CAMAC output register (Fig. 4) and an external digital potentiometer, the computer sets the temperature in the thermostat chamber to a value determined by the program. The digital potentiometer makes it possible to preset any resistance value between 1 Ω and 512 Ω in steps of 1 Ω. The resistance values are coded in a binary code and can be set in parallel with a 9 bit command. Then a certain time is allowed to elapse for temperature equilibration before starting the actual measurement sequence.
The computer program reads from a list of values the values assigned to each transistor, such as the specimen number, parameters, limit data and measuring position scan number, and sets the external instruments accordingly.
The use of a list of values makes it possible to use any desired sequence of measuring positions and, if necessary, to assign to each measuring position different parameters and limit data. The thermostat chamber has 100 measuring positions, each with four connections, for connecting the transistors to be tested. The measuring position read by the program from the list of values is selected via a CAMAC output register and a measuring position commutator unit. This commutator unit is equipped with a stepping electromagnet. It has a supplementary studboard with 100 contacts by means of which the measuring position number is scanned, via an encoder and an input register, as soon as the stepping switch has stopped on a contact. If no valid measuring position number is found, an error bit is reported back to the computer.
After a zero check of the two ammeters for the collector and the base currents of the transistor being tested, the voltage U_{BE} supplied by the Ul generator is set according to the list of values by a CAMAC output register. As this current-voltage generator, because of its data structure, must be set by means of two output registers, this necessitates the signal distribution shown in Fig. 4. By means of this arrangement the computer can not only set this generator to a present current or a preset voltage mode of operation but also set limit values for each of these two modes. The voltage U_{BE} supplied by the Ul generator is measured by a digital voltmeter; this measured value is reported via a CAMAC input register back to the computer and is checked. Then, after a certain time required for conversion, the digitalised values of the currents I_C and I_B are available at the appropriate CAMAC input registers and are read in by the computer.
The computer transforms these values to a form suitable for visual display, and stores them in this form. They are then displayed on the visual display screen on a semilogarithmic scale as $I_C = f(U_{BE})$ and $I_B = f(U_{BE})$.
The computer compares the measured I_C value with a specified maximum value stored in the list of values. This comparison decides whether the measurements on the particular transistors concerned are completed or not. If I_{Cmax} has not yet been reached the computer raises U_{BE} by a fixed increment, and the measurement sequence is repeated starting with the U_{BE} check. If I_C has reached the specified maximum value the voltage supplied by the UI generator is reset to zero and the data accumulated in the computer are punched out on tape.
The computer then decides on the basis of the list of data whether or not all the transistor specimens have been measured. If there still remain transistors to be measured, the computer selects the next measuring position

in the list of values and the measurement sequence is repeated starting with the zero check.
When all the transistors have been measured, the computer signals this by teletype and waits for a teletype command as to whether or not the entire set of measurements has to be repeated for all the transistors at another temperature. If so, the computer sets a new temperature for the thermostat chamber and, after waiting for temperature equilibration, the measurements are re-started from the first measuring position.
The electronic measuring circuits are equipped with safety circuits protecting the devices to be measured from destruction. In the event of a fault the measuring sequence is stopped, the external instruments are reset to zero, and the fault is printed out.
Before every measurement the computer checks that the thermostat chamber can be correctly set, that the digital voltmeter is switched for the correct operating mode, and that the current-voltage generator is at the starting setting. The function of the digital voltmeter is to check the voltage set by the computer on the U1 generator. If this voltage is too high or too low, a message is printed out on the teletype. If the preset limit current or the preset upper voltage limit is exceeded, the current-voltage generator transmits a signal to the computer via a CAMAC input register. Such a signal causes a program interruption which stops the measuring sequence.
Connected to the measuring positions 1 and 2 are test transistors with known characteristics by means of which the faultless functioning of the complete system can be tested in a preliminary measuring sequence before the actual measurements.

5. Data logging, data input, data output

The measurements described in this paper were carried out using a Siemens on-line computer S 301. The CAMAC process instrumentation was connected via a system control unit (the CAMAC branch driver) to the accumulator interface of this computer.
The program for the measuring sequence is stored in the core storage of the computer. When the computer requires data, it carries out a CAMAC operation via the CAMAC branch driver. In such read-in or read-out operations the process data are temporarily stored in the data register of the CAMAC branch driver before they are transferred into the accumulator of the computer or into the register of the CAMAC module concerned.
The measured data read out from the modules are processed further by the computer as stipulated in the program, so that intermediate results can be displayed on a visual display device. This enables the user to follow the measured data during the measurement, and to check that the system is functioning correctly. These intermediate results are stored in the computer and are then punched out on tape at the end of each measurement for subsequent evaluation.

This text has been published in full length in Kerntechnik, No. 3, 1973.

Literatur References

[1] Sah, C.T., R.N. Noyce und W. Shockley: Proc. IRE 45 (1957), 1228
[2] Spencker, A., G. Schnell, H.G. Wagemann und O. Meinhardt: Die Schädigung von bipolaren Transistoren durch Bestrahlung mit Protonen im Energiebereich 2,5-4,5 MeV und Elektronen der Energie 1,5 MeV. HMI-Bericht B 61 (1967)
[3] Goben, C.A., und C.H. Irani: IEEE Trans. NS 16 (1969), 43
[4] Spencker, A., und J. Siegel: Bericht über Strahlenresistenzuntersuchungen am RTL-Schaltkreis FBH 162. HMI-Bericht B 92 (1970)
[5] Bäuerlein, R., und K. Wohlleben: Raumfahrtforschung 12 (1968), 16
[6] Wagemann, H.-G., und K. Zander: Wirkung von Reaktor-Neutronen auf die Eigenschaften von Silizium-Zener-Dioden. HMI-Bericht B 63 (1967)
[7] Wagemann, H.-G., und K. Zander: Vergrößerung des Tunnelstroms in Silizium-Zener-Dioden bei einer Bestrahlung mit Reaktor-Neutronen. HMI-Bericht B 71 (1968)
[8] CAMAC - A modular instrumentation system for data handling. Revised Description and Specification. Bericht EUR 4100e (1972)
[9] CAMAC - Organisation of Multi-Crate System - Specification of the Branch Highway and CAMAC Crate Controller Type A. Bericht EUR 4600 e (1972)

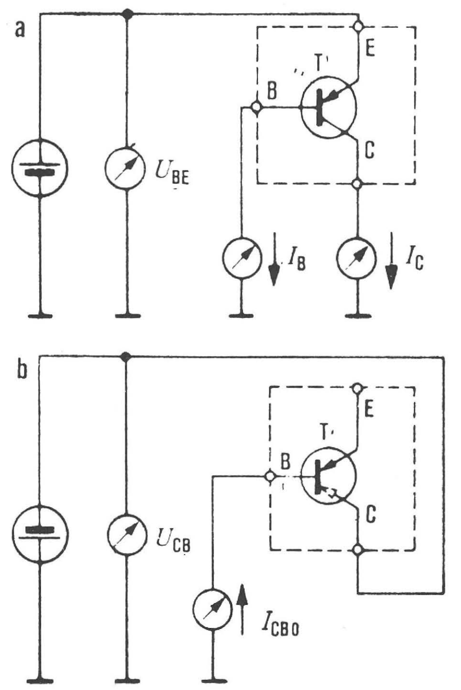


Fig. 1 a Prinzipschaltung zur Messung des Basisstroms I_B und des Kollektorstroms I_C in Abhängigkeit von der Basis-Emitterspannung U_{BE} bei verschiedenen Temperaturen T.

Basic circuit for measuring the base-current I_B and the collector-current I_C as a function of the base-to-emitter voltage U_{BE} at different temperatures T.

Fig. 1b Prinzipschaltung zur Messung des Kollektor-Reststroms I_{CBO} in Abhängigkeit von der Kollektor-Basisspannung U_{CB} bei der Temperatur T.

Basic circuit for measuring the collector leakage current I_{CBO} as a function of the collector-to-base voltage U_{CB} at the temperature T.

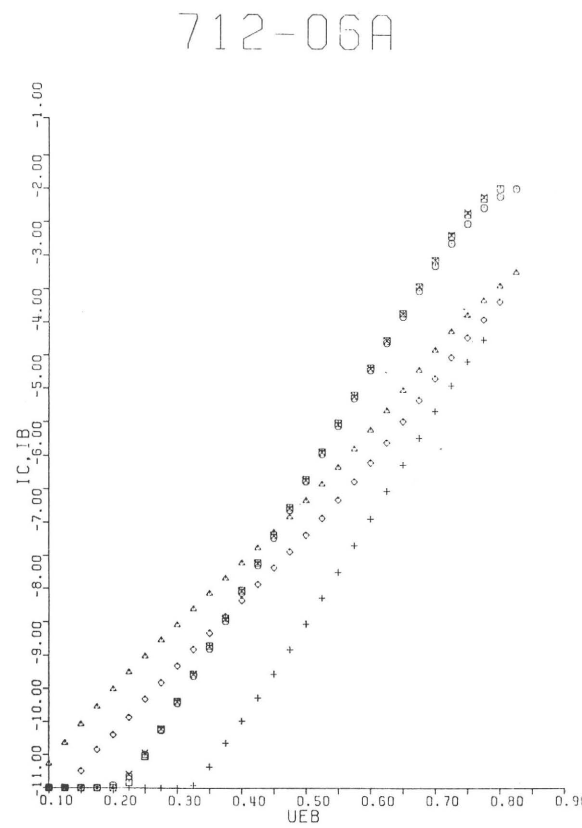


Fig. 2 Die halblogarithmische Darstellung des Basisstroms I_B und des Kollektorstroms I_C über der Basis-Emitterspannung bei verschiedenen Bestrahlungszuständen. Die Stromachse ist mit den Zenerpotenzen des Stroms in A beschriftet, die Spannungsachse in Einheiten V.

The semi-logarithmic diagram of the base-current I_B and the collector current I_C versus the base-to-emitter voltage at different irradiations. The vertical axis shows the current in decimal power, the horizontal axis shows the voltage in units.

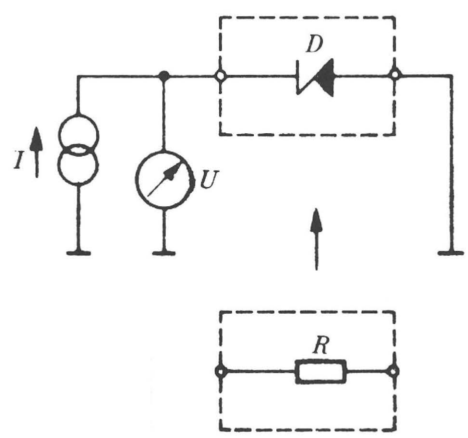


Fig. 3 Prinzipschaltung zur Messung der Zenerspannung U und der Spannung am Widerstand in Abhängigkeit des Stromes I bei verschiedenen Temperaturen T.

Basic circuit for measuring the Zener voltage U and the voltage at resistance as a function of the current I at different temperatures T.

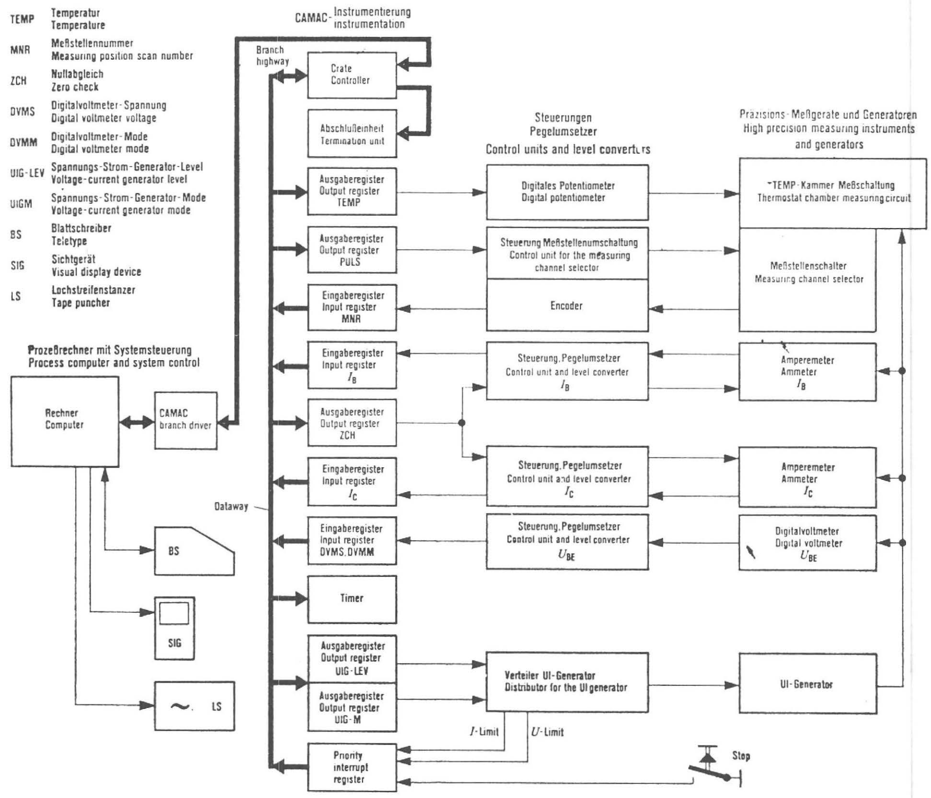


Fig. 4

Fig. 4 Blockschaltbild der CAMAC-Instrumentierung und der externen Geräte

TEMP - Temperatur, MNR - Meßstellennummer, ZCH - Nullabgleich, DVMS - Digitalvoltmeter-Spannung, DVMM - Digitalvoltmeter-Mode, UIG-LEV - Spannungs-Strom-Generator-Level, UIGM - Spannungs-Strom-Generator-Mode, BS - Blattschreiber, SIG -Sichtgerät, LS - Lochstreifenstanzer.

Block-diagram of the CAMAC-Instrumentation and the external equipment

TEMP - temperature, MNR - scan number of device under test, ZCH - Zero Check, DVMS - digital voltmeter voltage, DVMM - digital voltmeter mode, UIG-LEV-voltage current generator level, UIGM - voltage current generator mode, BS - teletype, SIG - display, LS - punch tape.

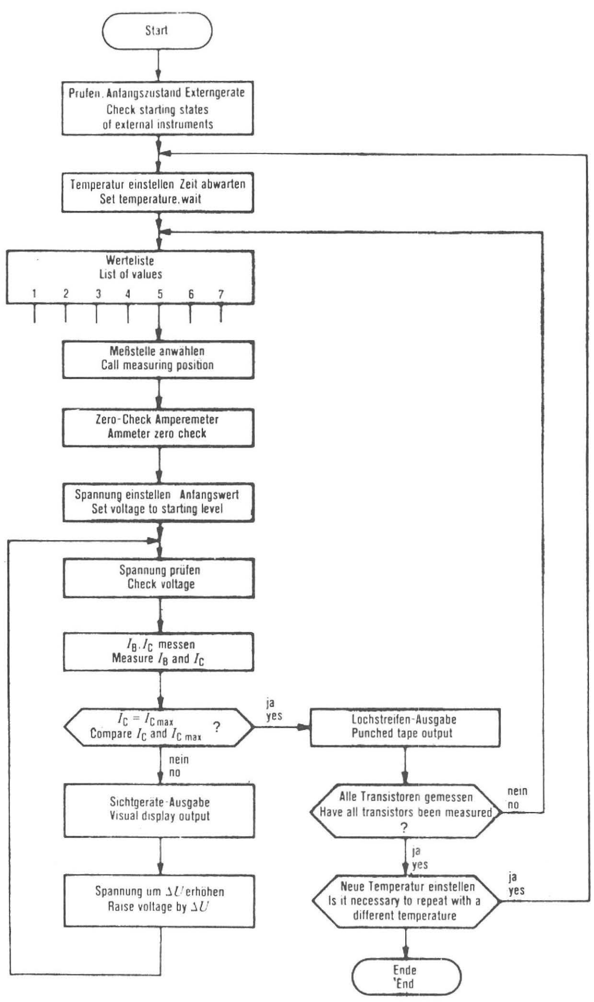


Fig. 5

Fig. 5 Flußdiagramm für die Messung von Vorwärtskennlinien an Transistoren.

Flow chart for measurement of forward characteristics of transistors.

CAMAC AS A COMPUTER PERIPHERAL INTERFACE SYSTEM

A. C. Peatfield, K. Spurling, B. Zacharov
Science Research Council, Daresbury Laboratory, Warrington, England

ABSTRACT

CAMAC is used extensively in the laboratory for data acquisition purposes in conjunction with small computers. The use of CAMAC has been further extended into the area of peripheral interfacing, with advantages both in convenience and cost. This development has made possible, for the first time, standardization of peripherals and interfaces while allowing advantage to be taken of new mini-computer developments.

ZUSAMMENFASSUNG

In dem Laboratorium wird CAMAC in Verbindung mit Kleinrechnern auf breiter Basis für die Datenerfassung eingesetzt. Die Verwendung von CAMAC wurde auch auf den Anschluss von Peripherie-Geräten ausgedehnt, was in bezug auf Annehmlichkeit und Kosten vorteilhaft war. Dank dieser Entwicklung war es zum ersten Mal möglich, Peripherie-Geräte und Interface-Einheiten zu standardisieren; aus der Weiterentwicklung der Kleinrechner kann dadurch Nutzen gezogen werden.

RESUME

Le système CAMAC est utilisé très fréquemment au laboratoire, conjointement à des ordinateurs de petites dimensions. L'utilisation du CAMAC a été étendue à la connexion périphérique, apportant dans ce domaine des avantages quant à la facilité de manipulation et aux dépenses. Ce développement a permis de normaliser, pour la première fois, les périphériques et les interfaces et aussi de tirer profit des progrès réalisés par les mini-ordinateurs.

1. INTRODUCTION

Until comparatively recently, it has required a considerable degree of specialist knowledge to interface computer peripherals to processors. Moreover, the problem of coupling a given peripheral could only be solved, in general, within the context of one particular system from a given manufacturer. In this, an essential part of the problem was the need to understand not only the operation of the Input/Output highway of a particular processor but also that of the peripheral to be interfaced. In practice, it was necessary to rely on the computer manufacturer or, more recently, manufacturer of the peripheral for a solution to the interface problem. Such solutions have invariably been of a special nature, valid only for the particular combination of processor and peripheral chosen.

The advent of CAMAC[1] has made possible a radical change in the pattern of computer peripheral interfacing. By adopting only a single standard highway, it has become possible to find new solutions to the problem that do not need to be changed or modified for each new processor to which peripherals are coupled.

At the Daresbury Laboratory there is a significant number of small computers whose applications and configurations differ from time to time. For these computers, it would have been too costly to provide all the peripherals that they were likely to require in their lifetime, if the peripherals were attached by the manufacturer. But since these computers have CAMAC attached for data acquisition purposes, it was a logical step to use CAMAC as the system for coupling any peripherals that would be required. There is a wide range of small computers in the laboratory, including Honeywell DDP-516, H-316 and H-112, Digital Equipment PDP-8 and PDP-11, Ferranti ARGUS 400 and IBM 1800.

2. CAMAC INTERFACE MODULES IN USE

2.1 Paper Tape Equipment

In order to provide high speed paper tape input and output, equipment was selected that would be both economic and reliable.

The punch chosen was the well known Facit 4040, giving an output rate of 75 characters per second. The CAMAC driver module is single width and has a Group 1 register for its data, being accessed by the WT1 or F16[3] command. The module has some sophisticated facilities such as the ability to be loaded with two characters at once in one sixteen bit word. The module may be synchronized by either TST (F27) operations or by using LAMs.

The reader selected was the Lynwood 500 having an input rate of 500 characters per second; the performance of the reader has been very good. The CAMAC driver module is single width and it has a Group 1 register for its data. This is accessed with the RC1 (F2) command. In order to facilitate both the building of hard-wired program loaders and keyed-in program loaders, the module is built to incorporate both the read and test operations in one single CAMAC function. By this is meant that when a RC1 operation was performed the Q response would be dependent upon the validity of the data at the time of the CAMAC operation in question. Thus all that it is necessary to do, to read a paper tape, is address the module with RC1 commands, and save the data or repeat the command depending upon the Q response. In more recent times, this procedure became known as the "Repeat Mode" (EUR 4100). The module can also be driven conventionally with TST operations or by LAMs.

2.2 Read Only Memory

It was seen at an early stage in the process of interfacing peripherals to CAMAC that the provision of a Read Only Memory module would ease some of the system implementation problems; keyed-in program loaders and hard-wired loaders are much simpler if it is possible to read in some program from a device that does not have to be synchronized with the program and has the same word length as the processor. Of course, a read only memory should be sufficiently fast to be always ready for successive CAMAC cycles. In addition to the area of program loading there is also a need for simple diagnostic programs to establish in which area any error conditions might lie. So, in addition to program loading facilities, the read only memories have been programmed to provide read and write program loops to access switch and lamp CAMAC modules. This allows for easy checking of the data lines in a system. There is also a program to allow for easy checking of the Q response line. Versions are available to allow a user to easily load his program from the central computer system through his data link, both for a high speed parallel link or a synchronous modem.

The module is of single width, with a Group 1 register for access to the program and a Group 2 register to set the address pointer to any one of the 256 sixteen bit words. The Group 1 register is read with the RD1 (F0) operation, and the address pointer is automatically incremented after each RD1 operation. The address pointer may be set with the WT2 (F17) command. This operation is required as the memory contains a table of addresses for the easy selection of the available options contained in its 256 words; for example, load paper tape, clear core, test data transfers or load program from the data link.

2.3 Moving Head Disc

While it is not usual at Daresbury to provide any bulk storage at the remote terminal computer systems, because direct access facilities are provided by high speed data links to the central computer, there is a requirement for disks in the area of the front end multiplexor to the 370/165. At present, this comprises an IBM 1802 with a number of small PDP-11's each with one or two CAMAC crates. It is this system that handles all the data to and from the experiments, the video display terminals and data links to locations outside the laboratory. The disk used is the D.R.I. Model 31 with a capacity of 24 million bits, the module being a double width unit. It is necessary, of course, for it to be used with a CAMAC system that has Direct Memory Access capability.

2.4 Displays

All the early output of a graphical nature has been performed at Daresbury by using commercially available display drivers, and providing users with software support to allow character and vector generation. All this graphical output was provided on 611 storage oscilloscopes, for which there existed no module to perform the remote control operations inherent in the operation of a storage oscilloscope. Therefore in order to augment the display drivers, a 611 Mode Control Unit has been produced to give direct connection to the remote programming connector on the 611 scope. This module has given the programmer the ability to erase the screen, put the unit into "write through" mode, or "non store" mode. This pair of units then form the principal output medium for experiments in the laboratory. These units are also used as program output devices on some of the small computer systems; they have been allocated to the operating systems in place of the line printer.

In order to allow CAMAC to be used as the vehicle for sophisticated interactive graphics, a new display driver has been designed. The module generates its own characters and vectors and is capable of being driven by DMA (Direct Memory Access) to allow a refreshed oscilloscope to be attached.

As mentioned previously, displays are driven from commercially available modules. Extensive use has been made of teletype drivers connected to video display terminals. Apart from their use as a control console for the remote systems, they have been used to provide access to the central computing facilities. These terminals are being used for such tasks as information retrieval, stock control, text-editing and implementation of IBM's Time Sharing Option. Some are used as control consoles for the 370/165 to facilitate job scheduling and system development. These are controlled by a PDP-11 with two crates of CAMAC. The two crates are taken up with interfaces for the 40 terminals and the connection to the central computer; this connection channel being a CAMAC interfaced data link.

2.5 Line Printer

The Tally 2200 printer, (200 lines per minute) has been interfaced with a single width module, and is used for a number of applications including remote job entry. The module has a Group 1 register which is loaded with a WT1 operation that transfers two ASCII characters to the line buffer of the printer. The module may be patched to allow either of the bytes to be transferred first into the buffer. When a line has been loaded, the printer is commanded to print the line with an XEQ operation. The module has it's status, mask and LAM requests accessed at Group 2 registers with RD2(F1), WT2(F17), SS2(F19), SC2(F23) operations. Status bits are provided for data ready, buffer ready, printer error, buffer full and paper out. An additional Group 2 register is provided for carriage control functions.

2.6 High Speed Plotter

An interface has been developed for the Versatec range of high speed plotters. These give a plotting speed of about one inch per second with a resolution of one hundred points per inch. They are being used as output devices on the central computing facility. The module closely resembles that designed for the line printer in the arrangement of its CAMAC functions and registers.

2.7 Card Reader

A CAMAC interface has been built for the Documation Model M600L card reader. This is a single-width module, and is designed to allow for high speed data transfers by direct memory access, since the card reader has a capability of 600 cards per minute. The module has the usual Group 1 register for accessing data with an RC1 operation. This may be used in "repeat mode" if required. The data ready signal may be brought out on the front panel or to a patch pin for DMA transfers if this is required. It has status, mask and LAM request registers at Group 2 addresses being accessed by RD2, WT2, SS2 and SC2 operations. Status bits are available for data ready, reader ready, hopper check and data over-run.

2.8 Modem Driver

In order to allow linking of remote users to the central computer network through telephone lines, a double width module has been produced that will interface to a synchronous modem. This module has been designed to connect to a communications line which employs binary synchronous transmission methods with CCITT V24 compatable equipment. The module contains circuitry to achieve the parallel to serial and serial to parallel conversion of data, byte synchronization, logic level conversion and status bits to allow the software to implement a complete communications system. The module provides Group 1 registers for reading and writing data with RC1 and WT1 operations. A timer LAM is provided from the module's internal timing to allow easy program time-out implementation. Status, mask and LAM requests are provided by Group 2 registers and RD2, WT2, SS2 and SC2 operations. Status bits are provided for request to send, byte synchronization established, carrier detect, data set ready, clear to send, timer interrupt, O/P data ready, I/P data ready and data over-run.

2.9 Badge Reader

As an aid to those systems concerned with stock control, a badge reader has been developed to assist in areas of identification and authorization. The badge reader reads a plastic badge with a ten by ten hole matrix. It may be connected directly to one of the Video Display interfaces running at 4800 baud and is therefore compatible with the many other serial devices in use.

3. CONVENTIONAL CAMAC SOFTWARE

The adoption of some form of convention in referring to CAMAC operations allows Input/Output routines for a particular peripheral to be easily transferred from one computer type to another. This has been achieved by the adoption of the CAMACRO[3] definitions at Daresbury, where the same operands are used on all CAMAC systems. This has also lead to very much improved communication between hardware designers and programmers, because they can now have a common language either to describe existing interfaces or to specify what facilities are required in new interfaces.

Without this standardization the object of making CAMAC a transparent interface between high level language program segments and the Input/Output devices is not readily achievable.

4. REMOTE JOB ENTRY WORKSTATION

A vivid demonstration of the potential that these developments provide has been the implementation of a remote job entry workstation. This facility has been put together using many of the interfaces and peripherals described above. The basic workstation provides for remote usage of a large computer system by means of a line printer (200 lines per minute), card reader (600 cards per minute) and video display terminal (4800 baud). Connection to the central computer is by means of a 4800 baud synchronous modem and line, together with a CAMAC interface.

The modules are fitted into a CAMAC crate controlled by a 4 K PDP-11/05. The CAMAC controller is interfaced directly to the PDP-11 UNIBUS and DMA facility is provided to allow for multileaving of the peripheral devices and the modem communication.

The workstation program runs under a simple supervisor (MFT-11, Multi-programming with a Fixed number of Tasks for the PDP-11) that has been written to handle CAMAC interrupts efficiently and to allow for easy expansion. The workstation communicates with the central computer using IBM HASP[4] protocol. At the central computer (IBM 370/165) is a PDP-11 with a CAMAC crate fitted with a number of modem drivers. These communicate with both CAMAC workstations and other computers acting as workstations (HASP emulators).

It should be noted that the workstation program in the central computer operates under OS 370 with MVT. This gives several advantages over using the HASP operating system including roll-out/roll-in capability. External workstations can be made to operate as if HASP existed in the 370/165. This is the only example of such operation that has come to our attention.

The basic workstation can be easily expanded by the addition of CAMAC interfaced paper tape reader, paper tape punch, high speed plotter and interactive graphics facility. Video display terminals can be added to give easy access to the central computer's Time Sharing Option. If data acquisition is required at the remote station then it is also an easy matter to add an extra CAMAC crate for this purpose. This of course makes available to a remote user all the facilities of the DNPL integrated data acquisition system[2] that are used by local users in the laboratory.

5. CONCLUSIONS

From early beginnings as a system for coupling nuclear instrumentation to processing systems, it has been a logical step for CAMAC to be used also as the interface for computer peripherals. The importance of this development is not only that it represents a convenient and indeed economical solution to the problem of coupling peripherals, but also because in this way the solution is essentially computer independent, standard and modular.

It is clear that the developments described in this present paper are yet another illustration of the versatility of the CAMAC system, and the range of applications of CAMAC has, by these means, become ever further broadened.

REFERENCES

1. ZACHAROV, B., CAMAC Systems - a pedestrians guide, Daresbury Laboratory Report, DNPL/R23 (1972).
2. SHERMAN, H.J., DANIELS, T., PEATFIELD, A.C., RICHMOND, P.M., and ZACHAROV, B., Proc. Int. Conf. on Instrumentation for High-Energy Physics, Frascati (May 1972), Daresbury Laboratory Preprint, DNPL/P157 (1973).
3. GOLDING, F.R., PEATFIELD, A.C., and SPURLING, K., CAMACRO, an aid to CAMAC interface programming. CAMAC Bulletin No.5 (November 1972) 29. Daresbury Laboratory Preprint, DNPL/P121 (1972).
4. HASP - IBM Program No. 360D - 05.1.014

EQUIPMENT MENTIONED IN THIS TEXT IS SUPPLIED BY :

Facit Office Equipment Limited
Maidstone Road
Rochester
Kent
England

Lynwood Scientific Developments Limited
Hartley Mews
Hartley Wintney
Basingstoke
Hants.
England

Data Recording Instrument Co. Limited
Hawthorne Road
The Causeway
Staines
Middlesex
England

Tally Corporation
8301 South 180th Street
Kent
Washington
U.S.A.

Versatec
Sintrom (U.K. agents)
2 Arkwright Road
Reading
Berkshire
England

Documation Inc.
P.O. Box 1240
Melbourne
Florida 32901
U.S.A.

CONTROLE DE REACTEUR PAR MINICALCULATEUR

J. Kaiser
Commissariat à l'Energie Atomique, Centre d'Etudes Nucléaires de Saclay,
Services d'Electronique de Saclay, France

RESUME

L'emploi de petits calculateurs dans le contrôle de processus permet une décentralisation des tâches en utilisant des terminaux intelligents reliés éventuellement à un calculateur central.

Dans le contrôle des réacteurs, un tel terminal peut en même temps assurer la fonction d'une voie de mesure neutronique très complexe et la fonction d'un dispositif de contrôle et de régulation.

ZUSAMMENFASSUNG

Das Benutzen kleiner Rechner in der Verfahrensregelung erlaubt eine Dezentralisierung der Regelkreise, indem man intelligente Terminals einsetzt, die eventuell an einen Zentralrechner angeschlossen sind.

In der Reaktorkontrolle kann ein solches Terminal gleichzeitig die Funktion eines komplexen Messzweiges als auch die Funktion des Reglers ausüben.

ABSTRACT

The use of smalls computers in the process control allows to decentralize the control loop. This is possible with intelligent terminals connected eventually to a central computer.

In the reactor control such a terminal may assure as well a measure function as a control function.

1 - GENERALITES.-

L'utilisation de calculateurs numériques dans le contrôle et la commande de processus industriels est devenue courante.

Dans le domaine des centrales nucléaires, la tâche du calculateur numérique était souvent réduite à l'acquisition des mesures et à la rédaction du carnet de bord. Le manque de petits calculateurs conduisait à une centralisation importante au niveau du calculateur. La sécurité, exigeant un haut degré de disponibilité et de fiabilité de l'ensemble, imposait le doublement du calculateur.

Les problèmes posés par la reprise de secours et la multiprogrammation etc.. ont fait apparaître un besoin de spécialisation du calculateur, tendance encore accentuée par l'apparition de petits calculateurs à des prix très compétitifs.

L'emploi de petits calculateurs permet la décentralisation des différentes tâches. Ceci mène à une configuration (figure I) comportant un calculateur central avec mémoire de masse, organes d'entrée et de sortie tels que imprimante, pupitre etc.., et de différents terminaux intelligents.

Dans le cas de centrales nucléaires, ces terminaux peuvent être spécialisés dans des tâches différentes telles que mesure et régulation thermique, détection de rupture de gaine, chargement de combustible, calcul de la marge de sécurité, mesure et régulation neutronique etc...

Le rôle du terminal consiste à acquérir les mesures nécessaires à la régulation. En plus, il transmet vers l'organe central les données devant être enregistrées ou utilisées par d'autres parties de l'installation. Il reçoit des données fournies par d'autres terminaux ou par l'organe central telles par exemple que le point de consigne et les paramètres nécessaires pour une régulation optimale.

Mais, son fonctionnement reste autonome et il assure la fonction de régulation même en cas de panne de l'organe central. Ceci est un point très important au point de vue sécurité.

Nous avons tenté de définir et de construire un ensemble assurant la fonction mesure et régulation neutronique d'un réacteur nucléaire.

Le prototype réalisé par les Services d'Electronique de Saclay est installé auprès du réacteur de recherche ULYSSE de l'Institut National des Sciences et Techniques Nucléaires de Saclay.

L'évolution rapide des petits calculateurs nous a obligés à rendre l'instrumentation indépendante du type de calculateur. Ceci mène obligatoirement à une standardisation de l'interface.

Nous avons choisi le standard CAMAC se prêtant facilement à un découpage modulaire des différentes fonctions. Au moment du départ de l'étude, le catalogue des fonctions réalisées en standard CAMAC ne satisfaisant pas nos besoins, nous avons dû étudier et réaliser un certain nombre de fonctions plus ou moins spécifiques à notre problème.

Le succès remporté par le CAMAC même dans le domaine de la régulation industrielle et la multitude des fonctions maintenant disponibles sur le marché nous ont donné raison dans le choix de ce standard.

2 - DESCRIPTION DE L'INSTALLATION.-

L'équipement est destiné à la mesure neutronique du réacteur sur une grande dynamique (10 décades) et à l'élaboration des signaux nécessaires à la régulation de la puissance. Sa tâche se divise ainsi en deux parties, l'une de mesure et l'autre de régulation.

L'ensemble de l'installation comporte un châssis CAMAC, le calculateur, les détecteurs avec les transmetteurs, un pupitre et un panneau de test et de télécommande (figure II). Pendant la phase de mise au point, une télétype permet d'entrer en communication avec l'ensemble. Mais la configuration finale ne prévoit plus d'intervention à l'aide d'une télétype, le calculateur étant considéré comme un élément fonctionnel de l'ensemble et non plus comme l'organe d'entrée. Par mesure de sécurité son panneau avec ses commandes manuelles sera condamné et mis hors service.

Un panneau de test et de télécommande et le pupitre sont les seuls éléments accessibles à l'opérateur. Le panneau de test et de télécommande est protégé par une clef et donne accès à une partie seulement des programmes. Un lecteur de bande installé sur ce panneau permet le chargement d'un programme.

2.1 Calculateur.

Le calculateur est un MULTI8/M301 de la société INTERTECHNIQUE. Sa mémoire a une capacité de 4K mots de 8 bits. Le contrôleur de châssis JCM8 utilisé une partie de la mémoire morte pour la programmation du transfert en entrée/sortie simultanée.

2.2 Modules CAMAC.

Le châssis CAMAC comporte le contrôleur JCM8, un tiroir de traitement de priorité des appels émis par les différents tiroirs et enfin les tiroirs reliant les organes d'entrée et de sortie au dataway.

Ces tiroirs assurent les fonctions suivantes :

2.2.1 Ictomètre (JIAC)

L'ictomètre est destiné à la mesure du taux de comptage fourni par les détecteurs neutroniques. Il contient une logique câblée permettant un prétraitement de l'information en fournissant au calculateur le résultat de la mesure en coups/sec. sous forme d'un nombre en virgule flottante. Son fonctionnement est autonome et permet la mesure du taux de comptage à précision constante sur une dynamique de 6 décades. Ceci est rendu possible par une adaptation du temps de comptage en fonction du taux (fréquencemètre automatique (figure III). Le résultat transmis sur les 24 bits R contient le nombre d'impulsions compté N, le temps de comptage t, un bit de parité et 3 bits indiquant le mode de fonctionnement. Le calculateur averti de la disponibilité du résultat par une interruption, peut demander le transfert et calculer la puissance P du réacteur et la période d'évolution T de celle-ci.

Le calculateur a la possibilité de déterminer le mode de fonctionnement de l'ictomètre à l'aide d'un mot de commande de 8 bits. Il peut ainsi changer la précision de la mesure, procéder à un certain nombre de tests au niveau du transmetteur, tels que variation de la haute tension du détecteur et variation du seuil de discrimination de l'amplificateur.

Compte tenu de la position géométrique et de la sensibilité du détecteur associé, les quatre voies de mesure permettent de couvrir une dynamique de 10 décades avec chevauchement entre chaque plage individuelle.

Un bit de bon fonctionnement qui tient compte du bit de parité, de la continuité de chaîne etc.. signale une anomalie présente.

2.2.2 Convertisseur AN-NA (JCANA)

Dans l'installation existante, le point de consigne de la puissance du réacteur est fourni à l'ensemble sous forme de tension continue (potentiomètre). Ceci nécessite une conversion analogique-numérique.

L'affichage de laapériode d'évolution réalisé à l'aide d'un enregistreur potentiométrique exige une conversion numérique-analogique.

Nous avons réalisé les deux convertisseurs dans le même tiroir en prévoyant la possibilité de mettre en série les deux conversions pendant un temps très court.

Ceci permet une vérification du bon fonctionnement de la conversion analogique-numérique du signal consigné.

2.2.3 Entrée-sortie numérique (JES)

Le pupitre comporte un certain nombre de commandes et de dispositifs d'affichage du résultat de mesure. Ce pupitre ainsi que le panneau de test et de télécommande est relié à l'interconnexion CAMAC à travers des tiroirs entrée/sortie.

Ce tiroir est réalisé sous forme modulaire permettant l'entrée des signaux numériques fournis par des organes tels que contact, touche, roue codée, etc... et la sortie des signaux pour l'attaque de voyants, d'indicateurs NIXIE, relais ou sous forme de contacts.

Le panneau de test et de télécommande comporte un lecteur de bande qui est relié au dataway à l'aide d'un tiroir JLB.

2.3 Organes d'entrée-sortie.

Voie de mesure neutronique.

Les voies de mesure sont équipées de détecteurs neutroniques tels que chambres à fission délivrant un train d'impulsions dont le taux est proportionnel au flux neutronique et chambre à ionisation délivrant un courant proportionnel au flux (figure IV).

Le transmetteur relie le détecteur à la fonction ictomètre et comporte un discriminateur-amplificateur et une fonction test pour les voies de chambre à fission.

Dans le cas d'une chambre à ionisation, le transmetteur comporte en plus un convertisseur courant-fréquence évitant ainsi la transmission des signaux analogiques.

2.4 Pupitre.

Le pupitre permet la sélection du point de consigne, l'affichage du résultat de mesure (puissance, période, réactivité, etc...) la commande des barres et dans la version finale, le choix du programme de conduite du réacteur (démarrage, montée en puissance, fonctionnement en palier, etc..)

2.5 Panneau de test et de télécommande.

Il présente le seul moyen de communication avec l'ensemble et permet ainsi, entre autres, l'exécution des différents programmes de test (courbe de discrimination, tracé du spectre, etc..). L'ensemble des programmes est chargé par un lecteur de bande perforée logé sur le panneau.

2.6 Entrée et sortie.

Dans la phase définitive, l'ensemble assure un fonctionnement en ligne. Le point de consigne et les mesures déterminent le fonctionnement du réacteur et permettent une régulation de sa puissance.

Ceci est rendu possible en agissant sur la position des barres de régulation. Cette position est signalée par un certain nombre de contacts de position.

Le mouvement des barres est assuré par des moteurs. Ils sont commandés par des contacts (Interface JES) à action fugitive, c'est-à-dire la sortie d'un bit concernant la commande d'un relais moteur enclenche le relais pour un temps prédéterminé (ms) à la suite duquel la commande est interrompue. Il est nécessaire de renouveler, par le programme de sortie, la commande du moteur. Ceci interdit en cas de panne du calculateur une sortie excessive des barres entraînant une évolution dangereuse de la puissance du réacteur. Les barres de sécurité permettant l'arrêt immédiat du réacteur sont commandées par un signal spécial.

3 CONCLUSION.

L'ensemble décrit assure une fonction mesure et une fonction commande et régulation.

Depuis deux ans, la partie mesure fonctionne d'une manière satisfaisante sur le réacteur ULYSSE. La mise en exploitation de l'ensemble complet assurant aussi la commande et la régulation est liée à une prochaine refonte de l'installation de la pile ULYSSE.

Néanmoins, la première partie a pu être vérifiée et nous avons pu confirmer les résultats obtenus par une simulation sur un calculateur hybride EAI.

L'utilisation du standard CAMAC a permis une réalisation "sur mesure" des interfaces en gardant la possibilité de concevoir une nouvelle installation avec les mêmes modules en utilisant un calculateur plus moderne.

L'auteur tient à remercier Messieurs GAUTHIER, FUKS, HEDDE, MERCIER et LACONTAL pour leur collaboration à l'étude et à la réalisation de cet ensemble.

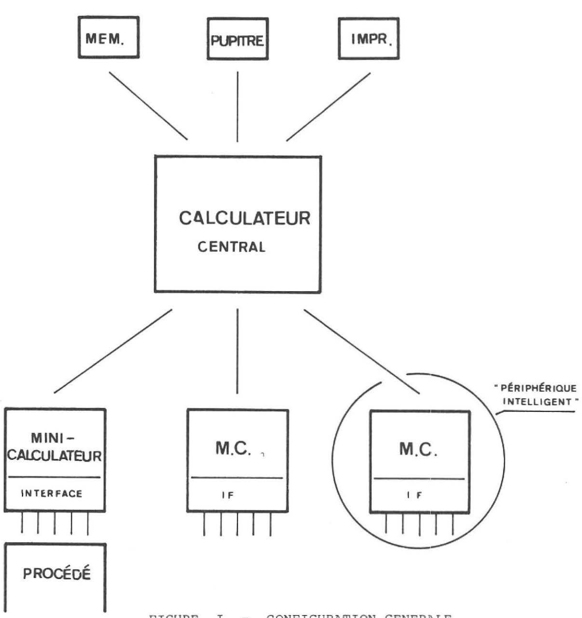


FIGURE I - CONFIGURATION GENERALE

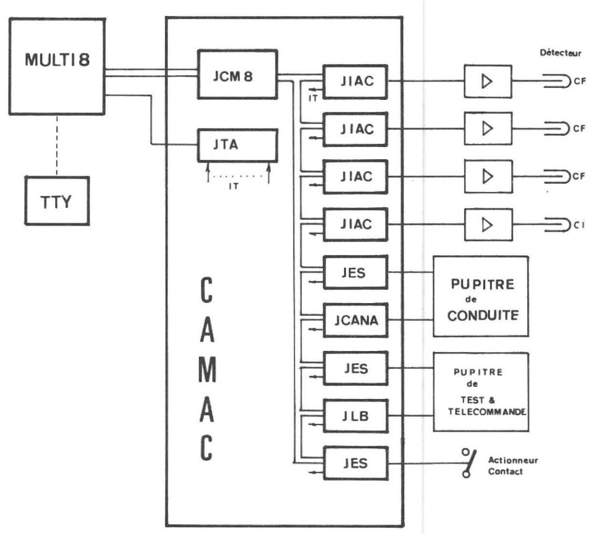


FIGURE II - CONFIGURATION DE L'ENSEMBLE

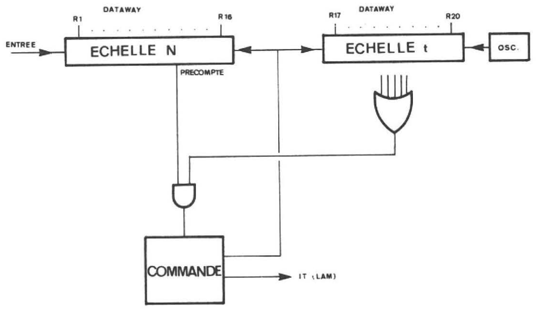


FIGURE III - BLOC DIAGRAMME DE L'ICTOMETRE

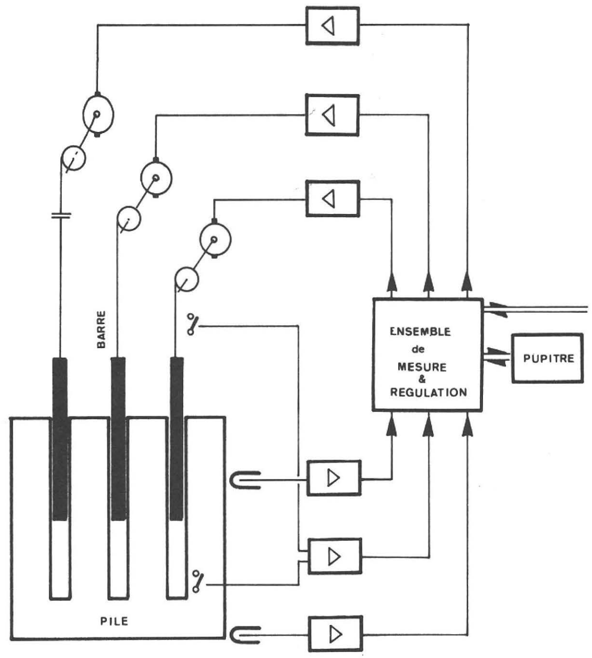


FIGURE IV - SYSTEME DE MESURE ET DE REGULATION

CAMAC AS A PROCESSOR INTERFACE IN COMPUTER NETWORKS

B. Zacharov
Science Research Council, Daresbury Laboratory, Warrington, England

ABSTRACT

From early beginning as a means of coupling instrumentation to processors, either for data acquisition or control, CAMAC has become used in much more extensive applications. In particular, because of standardization and modularity, and because of the wide range of processors with CAMAC drivers, there is now the possibility of using the CAMAC system as the interface between several processors in integrated computer networks.
The present paper discusses the role of CAMAC in computer networks, and describes the use of such systems in data links, terminal attachment, multiplexing and concentrating and in subsystem implementation.

ZUSAMMENFASSUNG

Während CAMAC zunächst nur die Aufgabe hatte, den Anschluss von Instrumenten an Datenverarbeitungsanlagen für Datenerfassung oder für Steuerzwecke zu gewährleisten, wird das System inzwischen viel weitgehender eingesetzt. Insbesondere die einheitlichen Regeln und der modulare Aufbau sowie die Vielzahl von Datenverarbeitungsanlagen mit CAMAC-Steuerung bieten heute die Möglichkeit, das CAMAC-System als Schnittstelle zwischen mehreren Datenverarbeitungsanlagen in integrierten Datenverarbeitungsnetzen zu verwenden.
Der Bericht erörtert die Rolle von CAMAC in Rechnerverbundnetzen und beschreibt den Einsatz solcher Systeme für die Datenübertragung, für die Verbindung mit Datenendstellen, sowie für Multiplexer- und Konzentrationsfunktionen und wenn Teilsysteme einbezogen werden sollen.

RESUME

Dans la première phase de son développement, CAMAC était utilisé comme couplage entre l'instrumentation et les processeurs, mais avec le temps, ses applications se sont multipliées. Compte tenu notamment, de sa standardisation et de sa modularité et compte tenu également du grand nombre des processeurs qui sont équipés de drivers CAMAC, on peut utiliser maintenant le système CAMAC comme interface entre divers processeurs dans les réseaux d'ordinateurs intégrés.
Le présent exposé étudie le rôle de CAMAC dans les réseaux d'ordinateurs; il définit l'emploi de ces systèmes dans les lignes de transmission des données, la connexion avec les terminaux, le multiplexage, la concentration, ainsi que dans la mise en oeuvre de sous-systèmes.

1 INTRODUCTION

There are many applications, both in nuclear science and throughout physics, which require the transmission of digital data between computers, instrumentation systems and terminal equipment. The importance of such communications channels is growing not only because of the needs for fast data-acquisition and terminal communications, but also because of the development of complete instrumentation systems involving direct feedback of processed data back to the data source and to human beings. One of the most important reasons which give rise to the need for communications links, however, is the integration of separate computer facilities into networks. Such integration is not merely economical and the only means of gaining reasonable access to unique computer facilities (such as data and program libraries), but there is now overwhelming evidence to show that individual users of network facilities can generally solve their problems with much greater ease than they could were they restricted to local systems only, for example by being able to use high-level languages, by eliminating local data buffering and by being able to interact with their computational processes on shorter time scales.

In the developments discussed here, there is a clear requirement for modularity and standardization, not only of the relevant interfaces but also of sub-system hardware and software modules. For these purposes, CAMAC turns out to be one of the most suitable and versatile systems in existence at the present time and, possibly just as important, it is the only standard of its type that is internationally accepted[1].

The present paper discusses the role of CAMAC in data communications applications, but it will do so with particular reference to the developments at Daresbury (DNPL), where extensive use of the CAMAC system has been made in the implementation of a powerful computing network. Several specific examples of CAMAC systems will be discussed, describing solutions to the problem of connection of experimental instrumentation systems to the network, the question of multiplexing user terminals and the problem of long-distance high-speed data communications.

2 THE DNPL DISTRIBUTED COMPUTER NETWORK

The DNPL computer network has been described elsewhere[2] and is shown in Fig.1. It is a network of radial topography with 370/165 as the hub. External equipment is connected to the network by any one of twelve high-speed wire links, which are multiplexed by a front-end IBM 1802 processor operating with a fixed, multi-task program. The links[3] are all identical and operate in full duplex mode at rates of about a megabyte/sec. They are very reliable and users of terminal equipment see only their programs in the central 370 computer; the links and front-end machine are transparent to users.

Each link terminates in one (sometimes two) of many small processors (such as H112, H316, H516, Argus 400, PDP8, PDP11, IBM1800), although there are some fixed facilities connected directly to a link, such as a complex of manual film measuring machines.

3 ATTACHMENT OF EXPERIMENTAL EQUIPMENT

Over the last three years, the principal means of connection of experimental instrumentation to the network, and hence into a high-level language module executing in the central 370 computer, has been by means of CAMAC. Each of the high-speed links can be used to attach CAMAC by means of a terminal computer and a CAMAC branch driver, and this has proved to be extremely useful, to the extent that all counter experiments at DNPL now perform all their data acquisition through CAMAC, into the network, and thus into the central computer.

The recent growth in CAMAC instrumentation, the need to communicate between the separate CAMAC systems of different users, and the wish to eliminate special interfaces have led, however, to the development of a new means of connecting CAMAC systems, the CAMAC-CAMAC link. Such a link is now replacing the purpose-made high-speed wire links previously used extensively at DNPL but, more important, it provides a rapid and simple means of inter-connection of already operational CAMAC sub-systems.

3.1 CAMAC-CAMAC Links

These consist simply of a multi-core balanced twisted pair line terminated at each end by a double-width CAMAC module, an A and a B module; the two modules are identical except for a number of line patches, allotting the correct pairs to control and status signals. Figure 2 shows one of the terminal modules, using printed circuit boards and high-density packaging.

The link has a single 24-bit data path used for transfers in both directions. In addition there are two groups of 5-bit paths, used for the transfer of a control word in either direction, if necessary concurrently. A data parity bit is generated at each end and transferred together with the data. In addition to the 5-bit control word, there are four status bits generated from the data and from the control word. The OR of the four status bits is used as a LAM signal, which can be enabled or disabled in the usual way by bits in a mask register.

Figure 3 shows a diagram of the link. Data transfers are normally made using a handshake system, as are control word transfers. Status bits are used to monitor system operation, and these may be accessed at any time. The Read Control register and Status register are normally addressed at Group 2, A15 and may be read by the CAMAC function F1. A channel synchronization facility is provided on both reading and writing, the synchronization signal appearing either as a LAM at (N-1) or on the P2 bus line, and being generated from the two special bits in the data register, a data validity and a data accept bit. In this way DMA transfers may be synchronized without program intervention, while still allowing a LAM at N to give indication of parity error or arrival of control information. It should also be noted that if operating data transfers by means of interrupts, then it is possible to receive these from the control register, since the 5-bit control path can be used during transfers.

TABLE I

COMMAND	FUNCTION	ADDRESS
Read Data	F2	A1
Test Read Data	F27	A1
Disable Channel Read	F24	A1
Enable Channel Read	F26	A1
Write Data	F16	A2
Test Write Data	F27	A2
Reset Write Channel	F25	A2
Disable Channel Write	F24	A2
Enable Channel Write	F26	A2
Write Control	F16	AO
Test Write Control	F27	AO
Read Status and Control	F1	A12
Test Read Control	F27	A12
Selective Clear Status	F23	A12
Write Mask	F17	A13
Read Mask	F1	A13
Test LAM Present	F8	AO
Reset Module	F25	AO

Table I shows the commands which are normally executed in the terminal modules, together with the appropriate sub-addresses. Using these commands, a simple sequence for read or write transfers is shown in Fig.4, where the Q-signal is used to test transfer validity. As has already been indicated, much more sophisticated and faster transfers are possible. With appropriate measures, the data transfer rates are limited only by the CAMAC cycle time (say 1-1.5 μs for 24 bits) and propagation delays.

It can be seen that the CAMAC-CAMAC link provides a very powerful and effective way of coupling systems together, and that it does so in a way that is extremely safe; if one CAMAC sub-system in a coupled network goes wrong, the worst that can happen is that it will signal the fact to the other coupled sub-systems.

In addition to the high-speed link just described, it is also possible at DNPL to couple two or more CAMAC systems by means of a slow, serial link, and this has proved to be very useful for isolated test facilities and for setting up experimental equipment. These serial links and relevant terminal modules will not be discussed separately here, for the equipment used is essentially similar to that to be described in the next section for multiplexing user terminals.

4 CONNECTION OF TERMINALS

There are already several ways in which terminals such as VDU equipment can be connected to a large computer. However, such ways are usually costly, restrictive and, in any case, special to each computing system and to each terminal. CAMAC provides an excellent way of connection of all types of terminal equipment to networks in a universal and most economical manner. This point can be illustrated by the way in which CAMAC is used at DNPL for multiplexing VDU terminals.

The high-speed links at Daresbury can allow the attachment of more than one terminal by means of a hardware data-link multiplexer, and one of these is used with a branch that terminates in a CAMAC module. This CAMAC module forms one part of a CAMAC sub-system of many crates, with a small processor (PDP-11) used as the controller for the Branch. Figure 5 shows the sub-system. In the particular example being described here, the crates contain mainly a number of single-width modules which attach remote alphanumeric VDU terminals coupled by twisted-pair serial links. The software and system at DNPL allows the communication then between any of the terminals and any program executing under the normal operating system of the central computer (MVT in this case). The role of the PDP11 control processor here is as a multiplexer and message concentrator, for which purpose there is a fixed stored program. However, the whole network system is completely transparent to any terminal user.

The terminals connected in the way just described are used for many purposes, including information retrieval, stock control, text editing and so forth. However, an example of application, which demonstrates the power and versatility of the system in a particularly vivid way, is that one terminal is actually used by the operators of the central 370/165 computer for job scheduling and other tasks, while another terminal is used by those people whose responsibility it is to further develop the hardware and software of the network and front-end system!!

An important comment to make here, is that it turns out much more expedient to attach VDU terminals in the way described that it would be using standard manufacturers equipment. There is already extensive and most successful use of VDU terminals at DNPL, for example for the implementation of the TSO system of IBM.

In addition to the applications just described, there is also at DNPL a similar CAMAC sub-system for connection of other terminal devices. An example of one such terminal device has already been mentioned, namely another CAMAC system coupled by slow (100 K bit/sec) serial link for the attachment of experimental instrumentation. However, another important set of terminal devices are modems, allowing the communication to distant 'work stations' at rates up to about 9.6 K baud.

5 MICROWAVE DATA LINK

The third example of CAMAC in computer communications has already been described[4] but, because of the relevance, will be briefly discussed here. In this development, the problem that confronted DNPL was how to extend the full power of the computer network including the central 370/165 processor, to remote users at distances of up to 150 kms. The solution to the problem is shown in Fig.6, in which standard Post Office video microwave equipment is used without modification for digital transmission at 10^7 bits/sec to a node at 35 km distance.

By using CAMAC and by developing only two new modules, namely a transmitter and a receiver module, a system has been implemented in which it is easy to connect to any terminal equipment, such as a work station, to the Daresbury network at distances of certainly 150 kms. The complete system has been extensively tested by coupling the transmitter and receiver at the remote node, and performing interactive graphical tasks with the results of computations transmitted an arbitrary number of times around the microwave link. With the available bandwidth (5 MHz at 6.8 GHz carrier), we have experienced no difficulty in achieving corrected error rates of better than 1 bit in 10^{10} for sustained transmission at 10^7 bits/sec. It is now planned to make extensive connection to remote users by means of mixed microwave link and video cable.

6 WORK STATIONS

Work stations have already been mentioned and will be described in detail elsewhere. However, it is useful to discuss them briefly here, because of the use of CAMAC in their implementation. Such stations are intended to be used remotely, not only for the usual purposes of remote job entry by cards and for reception of printed results, but also for the extensive connection of VDU terminals, the connection of data-generating instrumentation and eventually for the attachment of fully-interactive graphics terminals.

It has already been noted that there is no difficulty in making the remote end of data links, either slow wire lines or high-speed microwave links, coupled into a CAMAC subsystem by means of an appropriate module. Thus, DNPL work stations are, in effect, CAMAC subsystems with a local control processor. The CAMAC crates may then contain any driver modules required and, in particular, modules enabling the attachment of any standard items of peripheral equipment such as card readers, line printers, storage-tube displays, alphanumeric terminals and so forth. The use of CAMAC in this way has been developed extensively at DNPL and is described in another paper[5].

Figure 7 shows a typical work station using CAMAC.

Naturally, the method of attachment of CAMAC-based remote work stations into integrated computer networks of the type that exists at Daresbury will depend upon the distance and upon whether or not the data link is through a common carrier. In the domestic environment, the problem is solved simply as already described by a simple CAMAC-CAMAC link, with appropriate modules at each end. In the case of use of common carrier lines however, the solution is somewhat different, because of the need to employ approved modems or line adapters. This problem has been solved readily by implementing just another CAMAC module, namely a coupler to a modem; in our case, such modules have been developed for coupling to standard modems operating at up to 9,600 Baud (although there is no reason why higher speeds could not be accommodated).

7 CONCLUSION

Some representative examples of the use of CAMAC in computer communications have been discussed. In each of them, the existence of CAMAC has made possible solutions which are both effective and economical. Moreover, because of the modularity and standardization of CAMAC, many of the solutions to new problems have extensively used sub-systems or modules previously developed for other problems. There can be no doubt that the ability to incorporate existing CAMAC sub-systems into new systems, together with commercial availability, will be major factors in the further development of data communications.

REFERENCES

1. See for example, CAMAC systems - a pedestrians guide, ZACHAROV, B. Daresbury Laboratory Report DNPL/R23 (1972).
2. SHERMAN, H.J., DANIELS, T., PEATFIELD, A.C., RICHMOND, P.M., and ZACHAROV, B. The Daresbury on-line system. Proc. Int. Conf. on Instrumentation for High-Energy Physics, Frascati, May 1973, Daresbury Laboratory Preprint, DNPL/P157 (1973).
3. BROOKS, J.R., DANIELS, T., HEYES, P., PEATFIELD, A.C., and ZACHAROV, B. The use of data links to the IBM 360 system at Daresbury Nuclear Physics Laboratory. Proc. Int. Conf. on Remote Data Processing, Paris (1969), Publ. Editions Chirons 2, pp.1136-1155.
4. PEATFIELD, A.C., SHERMAN, H.J., WHITE, A.J. and ZACHAROV, B. Microwave data link for computer communication. Proc. IEE, 120, (1973) 186.
5. PEATFIELD, A.C., SPURLING, K., and ZACHAROV, B., CAMAC as a computer peripheral interface system. To be published in IEEE Trans. Nucl. Sci. (1973).

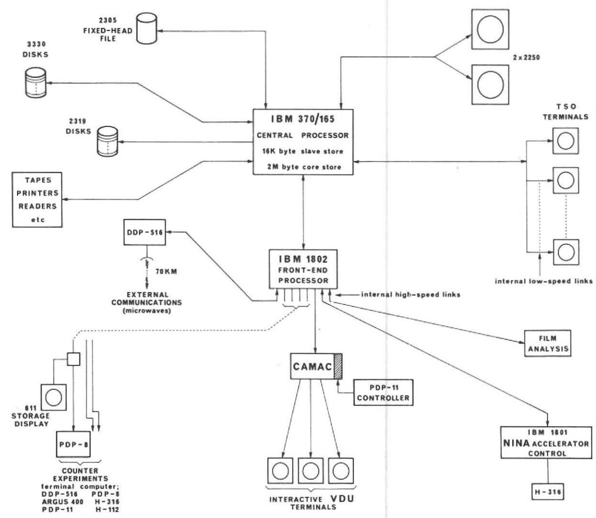


Fig.1 DNPL integrated computer network

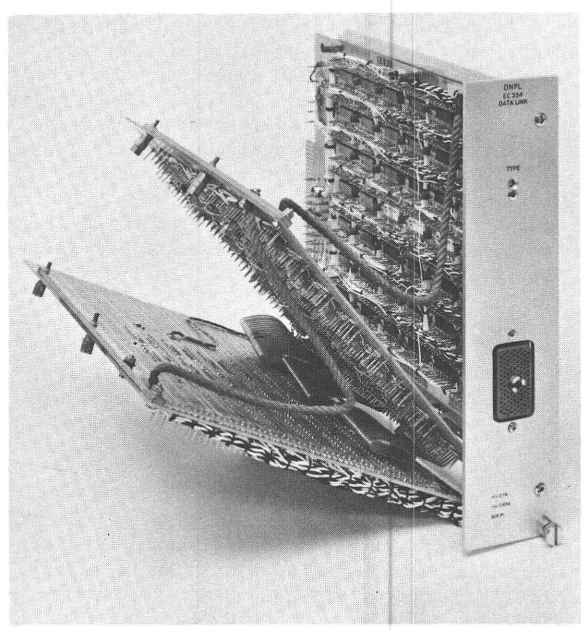


Fig.2
High-speed CAMAC data-link terminal module.

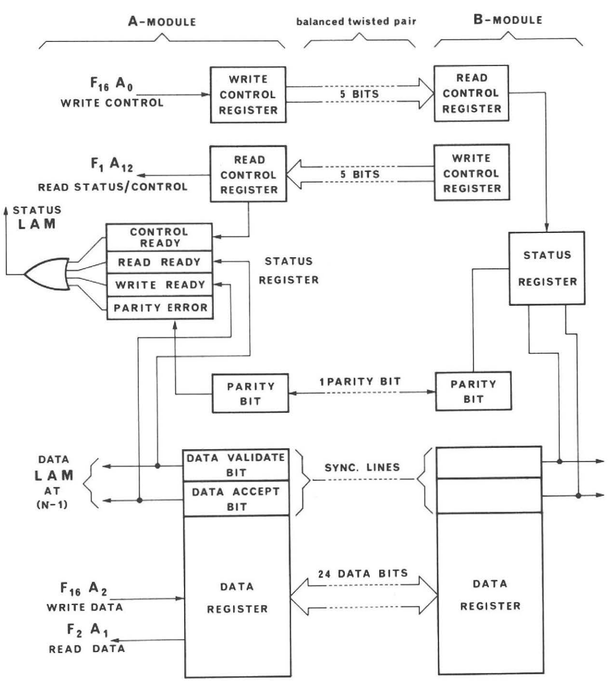


Fig.3 Block diagram of CAMAC-CAMAC link.

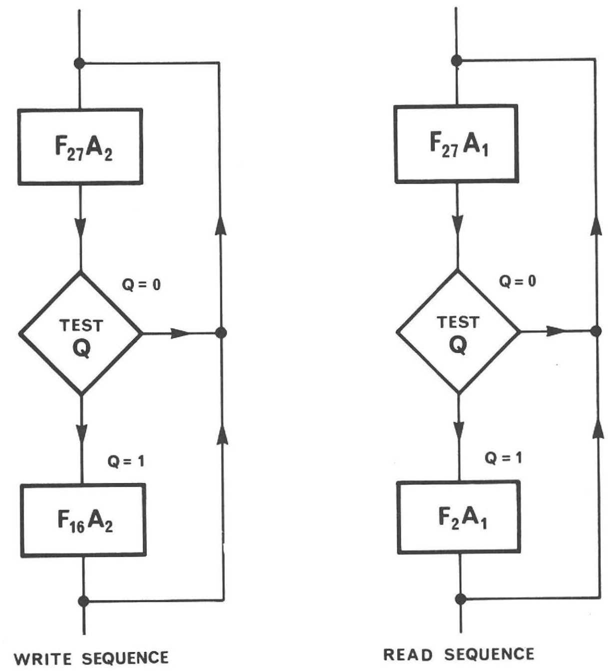


Fig.4 Simple command sequence for data transfers.

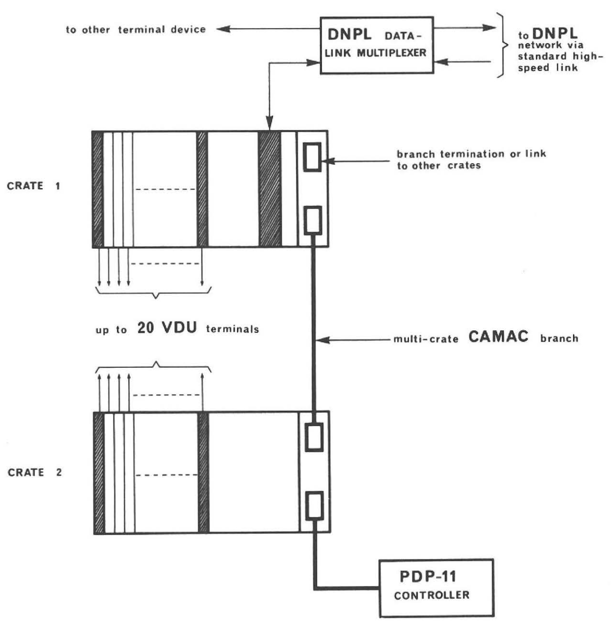


Fig.5
Use of CAMAC as a terminal multiplexer and message concentrating system.

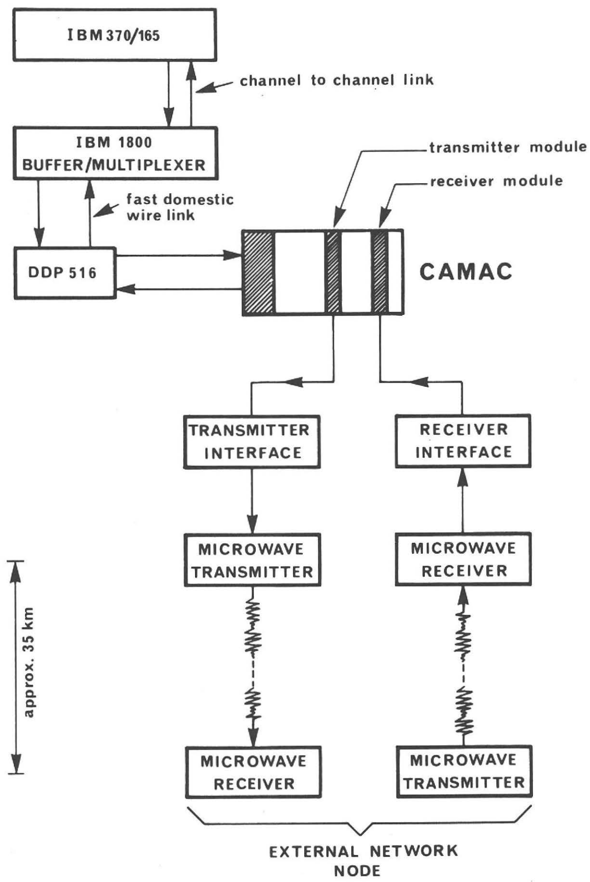


Fig.6 DNPL microwave data link.

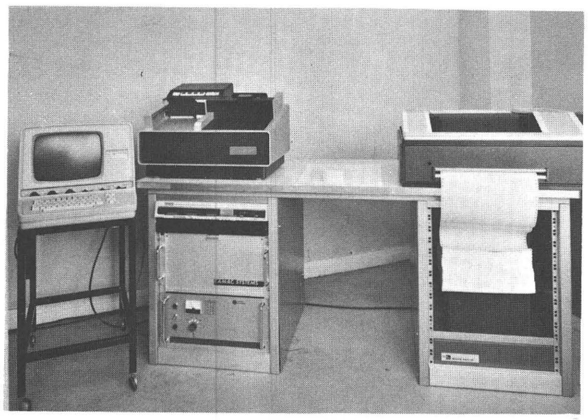


Fig.7 A CAMAC-based remote work station.

A COMPUTER BASED REAL TIME ELECTRONIC EQUIPMENT REPAIR AND COMMISSIONING FACILITY

P. N. Clout, G. Hughes
Daresbury Laboratory, Warrington, England

ABSTRACT

A multi-station real time laboratory facility is described which permits simultaneous use of a central computer, an IBM 370/165, via a high speed data link and local computer. The facility provides a number of working stations each of which can be used to test, commission, and repair a wide range of electronic equipment using the central computer as the control computer. Interactive control and diagnostic programs written in a high-level language are held in the central computer. The facility uses CAMAC to interface to the local computer all peripherals, as well as the equipment to be tested.

ZUSAMMENFASSUNG

Eine mehrere Experimentstellen umfassende Labor-Einrichtung für Echtzeitbetrieb wird beschrieben. Sie gestattet den Parallelzugriff auf einen Zentralrechner (IBM 370/165) über eine schnelle Datenanschluss-Stelle und auf einen lokalen Rechner. Die Anlage umfasst eine Reihe von Arbeitsstationen. Jede Station kann zum Testen, zur Inbetriebnahme und zur Reparatur einer Vielzahl elektronischer Geräte verwendet werden, wobei der Zentralrechner die Steuerung übernimmt. In einer Programmiersprache von hoher Ebene geschriebene Steuer- und Diagnoseprogramme für Dialogbetrieb sind in dem Zentralrechner gespeichert. Die Anlage benutzt CAMAC für die Verbindung aller Peripheriegeräte und der zu prüfenden Geräte mit dem lokalen Rechner.

RESUME

Description d'un dispositif de laboratoire multipostes fonctionnant en temps réel qui permettra l'utilisation simultanée d'un ordinateur central de type IBM 370/165, par l'intermédiaire d'une liaison à grande vitesse et d'un ordinateur périphérique. Le dispositif dessert un certain nombre de postes de travail dont chacun permet les essais, la mise en oeuvre et la réparation d'un grand nombre d'appareils électroniques qui utilisent l'ordinateur central comme ordinateur de contrôle. Le contrôle conversationnel et les programmes de diagnostics écrits dans un langage de haut niveau sont mémorisés dans l'ordinateur central. L'installation utilise CAMAC pour connecter à l'ordinateur local tous les périphériques ainsi que l'équipement qui doit faire l'objet d'essais.

1. INTRODUCTION

In any laboratory where extensive use is made of computers and electronic equipment for data acquisition and control there is the need for rapid and comprehensive repair and commissioning facilities. The organisation of the Daresbury Laboratory is such that the provision and subsequent maintenance of all electronic equipment supplied to experiments is through a centralised computing and electronics division. This gives opportunities for standardisation and high utilisation of equipment under these conditions. Indeed, without such standards, such as CAMAC, the number and variety of interfaces in a laboratory such as Daresbury, would present a considerably greater maintenance requirement. Even so, it is economic, if not essential, to provide an automated repair and commissioning facility to fully test each module. The need for this service at Daresbury is indicated by the number of instruments in use. There are approximately 14,000 items of which 2,500 are CAMAC modules, the remainder are typical high energy physics instruments.

2. DEVELOPMENT OF REPAIR AND COMMISSIONING SERVICES AT DARESBURY

It became evident when the Laboratory began to make extensive use of computers and interfaces that the only satisfactory method of testing and commissioning the many and varied instruments was to use a computer. A Honeywell 316 was selected and diagnostic programs were written in assembler langauage until the introduction of CAMINT, an interactive assembler program based on CAMACRO language (1). In practice, no significant library of diagnostic programs was built up and the congestion caused by many simultaneous demands to use the computer led us to develop a shared interactive computer system which allows several engineers to work simultaneously and autonomously sharing the Laboratory's computing facilities.

3. THE PHILOSOPHY OF THE NEW SYSTEM

The Laboratory makes extensive use of high speed data links to connect experiments to the central IBM 370/165 computer via satellite computers (2). In general, the control programs used are written in assembler language and are unique to that experiment. Because it is inefficient in program writing time to write custom programs for each user, the Laboratory has introduced a modular high level language programming system, BUTTRESS, which allows a computer at the remote end of a high speed data link to multiplex many users to the Time Sharing Option, TSO, facility in the IBM 370/165. By the use of sub-routines, added to the FORTRAN and PLI library, users' programs running as a foreground job are able to interact with a VDU or teletype and fully operate the users' CAMAC crates. TSO also provides easy methods of file creation and editing for the user. It was decided to implement this system for the repair facility and in this way the full power and facilities of the central computer are available at the users' CAMAC crate.

5. SYSTEM DESCRIPTION

The system layout is shown in figure 1. The remote end computer is connected to the central computer by a high speed data link and an IBM 1802 used as a front end processor. The Honeywell 316 computer acts as a message switcher working on interrupts and supports initially up to six repair bays. Each bay is equipped with a CAMAC crate which is used as the test bed and/or interface for the instrument to be tested. Interaction with the central computer is obtained through a VDU allowing the engineer to call down previously written test or diagnostic programs. Although the facility is especially useful in repairing and testing CAMAC modules, interfaces have been developed which allow non-CAMAC instruments to be excercised.

6. CONCLUSIONS

Inevitably the conclusion must be reached that to satisfactorily test or assess the performance of computer based electronic equipment it is essential to use a computer to carry out the exercising. It is often impractical and expensive to dedicate a computer to a single function or engineer. The system described accepts this limitation and provides on-line computational facilities to the engineer which a few years ago would have seemed impossible. These facilities have been provided at a fairly trivial additional cost compared to traditional bench testing.

REFERENCES

(1) GOLDING, R.F., PEATFIELD, A.C. and SPURLING, K. CAMACRO - An aid to CAMAC Interface programming CAMAC Bulletin No. 5, 29-30, 1972

(2) BROOKS, J.R., DANIELS, T., HEYES, P., PEATFIELD, A.C. and ZACHAROV, B. The use of data links to IBM 360 system at DNPL, Proc. Int. Conf. on Remote Data Processing, Paris 1969, Publ. Ed. Chirons.

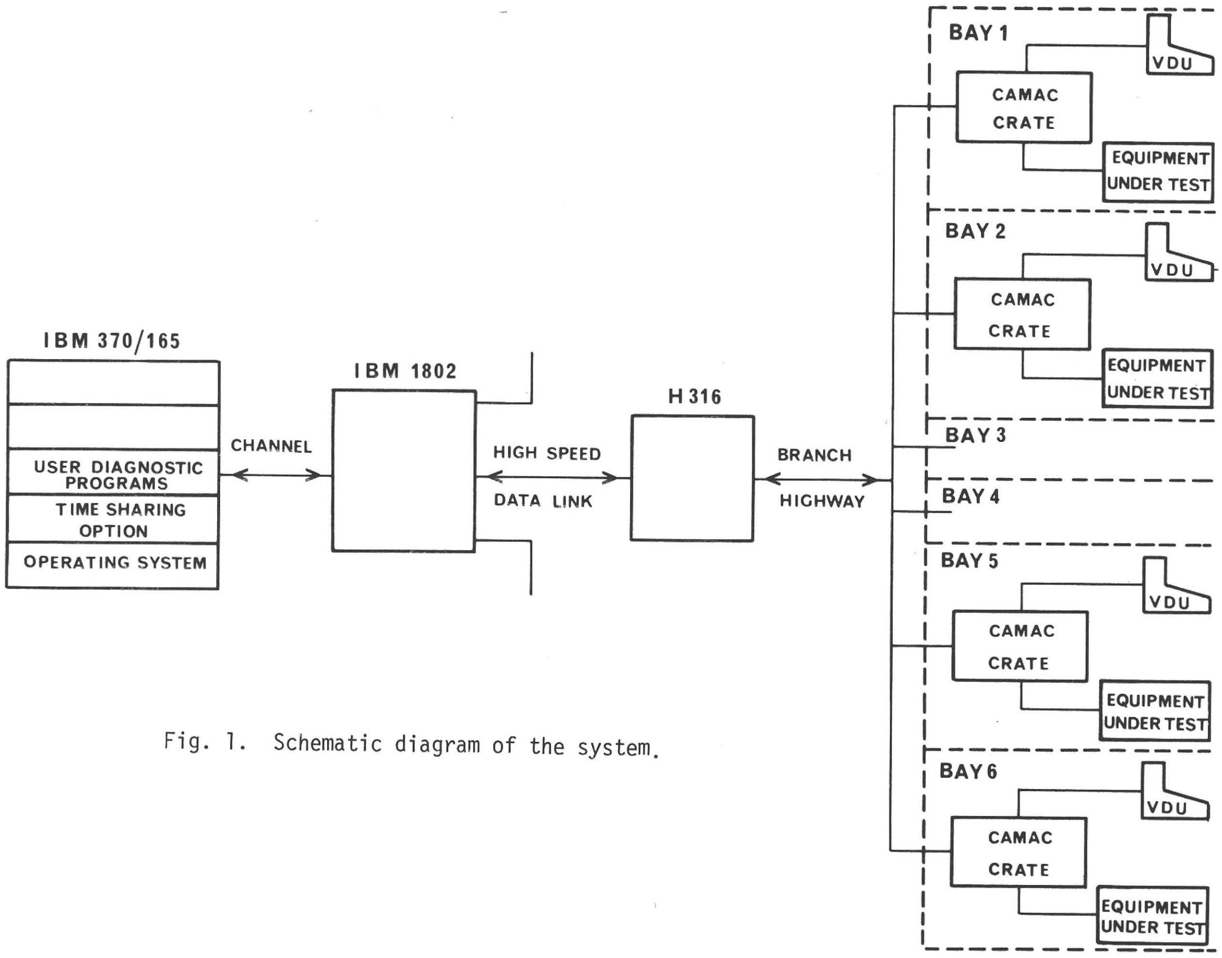


Fig. 1. Schematic diagram of the system.

A SECURE DATA LINK BETWEEN AUTONOMOUS DATA HANDLING SYSTEMS IN CAMAC

J. A. Laver
Central Electricity Research Laboratories Leatherhead, England

ABSTRACT

Existing CAMAC methods are inappropriate for data transmission over long distances. The requirements of a suitable system are discussed including the needs for remote station autonomy and error control. The description of a suitable system is outlined.

ZUSAMMENFASSUNG

Die bestehenden CAMAC-Spezifikationen beinhalten keine Regeln für eine Datenübertragung über grosse Entfernungen. Die Anforderungen an ein geeignetes System sowie die Erfordernisse in bezug auf die Autonomie der entfernt von Datenverarbeitungseinrichtungen befindlichen Datenendstellen und für die Fehlerprüfung werden erörtert. Ein geeignetes System wird beschrieben.

RESUME

Les systèmes CAMAC actuels sont inadaptés aux transmissions de données sur de grandes distances. On discute les exigences d'un système approprié, y compris celles nécessaires à l'autonomie des stations lointaines et le contrôle d'erreur; on ébauche la description d'un système approprié.

1. INTRODUCTION

Because CAMAC(1) originated for laboratory applications and has no inherent means of error control its scope for data transmission is severely limited. The possibility of extending the use of CAMAC into many commercial applications such as computer links, telecontrol, telemetering etc. depends not only on the economics but also on the availability of appropriate methods of secure high speed data communication over long distances. Clearly the philosophy of such a system must provide for some autonomy at the remote unattended station, so that in the event of a communication system outage the remote station cannot be left in a dangerous or irretrievable state.

Present CAMAC systems are usually interconnected on a master-slave basis and are thus totally dependent on the communication link integrity. Even if some autonomy is vested in the remote system this cannot be called into use without a request being passed over the communication link.

The paper examines this question of autonomy and makes specific proposals for a CAMAC/modem coupler which would provide a solution to the problem and could open up a whole new range of possible applications of CAMAC.

2. AUTONOMY OF OPERATION

In any normal master-slave system(2) the slave cannot take any action without either being directly controlled by the master, or at least first obtaining permission for independent action from the master. If the slave system is remote from the master, then such action or permission is totally dependent upon the integrity of the communications link. It is likely that most electricity supply industry uses of remote CAMAC systems will require a degree of autonomy. This requires that continued satisfactory local working should in no way be dependent on the communications path, which itself should only be responsible for the flow of data.

Moreover, since control signals or vital data could flow through such a data link, some form of error control should be introduced.

The following example illustrates the principle.

A remote CAMAC data-logging system monitors the effect of wind on a high structure. A computer controlled master data-logging system collects information from the remote system. At a prescribed wind velocity, alarm levels are exceeded. The master logger then transmits an instruction to the remote logger to increase the sampling rate and also to initiate local data recording so as to avoid total loss of information in the event of a communication failure.

If the remote system were not autonomous then, in the event of a communications failure, there would be no logging. If there were not some form of error control on the command link the remote site controller could jump into a mode of useless operation from which it cannot escape without direct local intervention.

Most industrial applications of tele-CAMAC (data logging, remote control, etc.) will require remote systems to maintain independent operation in the event of a communication failure, which then only prevents transfer of data between systems for the duration of the failure. Such autonomy can only be achieved if the communication system is solely a data link and does not also have to pass a stream of control information. However, restricting such a link to sending data alone does not prevent it from being used as a powerful control channel if desired.

3. LINK ORGANISATION

There are basically two types of link organisation, star (or radial) and highway. In the latter case there are two variants; the highway can be connected to a chain of terminals or it can be in the form of a ring, which can give a measure of security by redundancy.

Data transmission systems needing security of operation often use a combination of star and ring organisation with one system supporting the other.

Irrespective of the way in which the system is operated, the basic hardware should not impose any constraints on security. Clearly it is unacceptable to use any arrangement in which the continuity and freedom from error of the whole communication path is an essential to permit operation of any one remote part of the system such as a distant CAMAC "crate".

4. REMOTE CONTROL OVER A DATA LINK

In order to provide security of operation of a remote autonomous system, this must only be connected via a data link as opposed to a link conveying commands as well as data.

'Decision Software' is one method of obtaining control over a data link. This requires that some reserved characters should be recognized at the receiving end as commands to change mode, prepare for special data, etc., i.e. the system is not fully transparent. It would not be possible to use this method for any link requiring full dynamic range and code transparency of data.

An example of decision software is as follows.

Data word presented

Read data word

Compare with word, 'xyz', which it is likely to receive

If it is 'xyz', go to sub-routine n...

If it is not 'xyz', try word...etc.etc.

If not any acceptable word, carry on as before.

A more secure version of this might be preferable. This may be effected either (a) on a handshake basis or (b) by the use of multiple tests.

In (a) the system exchanges signals on the basis:-

..."I interpreted the word you sent me as 'xyz', was I right?"...

In (b) the system responds to incoming signals in the following way:-

... Word 'xyz' received, go to instruction n...

Wait for word 'abc' to follow within t seconds

If 'abc' appears within t seconds do...

Time = t seconds, word 'abc' not received, carry on as before.....

In this case a return communications channel is not needed.

5. ERROR CONTROL SYSTEMS

If data of small statistical significance is passing over a link, then possible errors could either be tolerated or, should they be grossly out of range or of incorrect format, erased. However, when data of high significance or actual control signals themselves are passing then an error control system is required to ensure secure delivery of the whole of the protected data without error.

For example in Section 2 some recovery procedure must be used to protect against system inhibition ("hang-up") if spurious or corrupted data are interpreted as being of control significance. Without this and in the event of such an error the remote system could be put into a mode in which it becomes unusable and from which recovery could not be achieved without local intervention. If the principle described in Section 4 is used to indicate that the next "n" words are to be stored and acted upon as a program modifier, then no corruption of the following "n" words could be tolerated.

Error control systems fall into two broad categories, Forward Error Correction (FEC) and those requiring handshake facilities.

With FEC a data block carries a parity structure such that any corruption in transmission may be both detected and corrected with a reasonable degree of certainty.

FEC systems usually require a considerable degree of intelligence. This intelligence may not always exist in CAMAC systems.

Handshake systems require only that the parity structure be used at the receiver to detect an error; the receiver then requests retransmissions until it is satisfied that the data is valid.

There are four types of handshake systems suitable for CAMAC use.

(1) Dump: If an error is detected in the received information, that word is deleted completely. This is used for data of small statistical significance.

(2) ARQ: If an error is detected at the receiver, a supervisory channel is then used by the receiving terminal to request automatic repeat of transmission, the new incoming character overwriting the previous corrupt character. This is one of the most widely used systems in data transmission.

(3) Echo: In this system a duplex transmission link is required. Every character received is sent back to the originating end, which checks that it is the same as was last transmitted. The originating end then transmits a code word 'accept' or 'reject' (which should itself be inherently error correcting). The original receiving end must then discriminate between these two possibilities. If it is found to be 'accept' then the original data character is released to the output, if 'reject' it is dumped and repetition is awaited. This system is not efficient in link utilization as data and control characters are interlaced and there is a full link path delay between each 'handshake'.

(4) Multiple Transmit: In this system the transmitting end sends the same character a number of times, while the receiving end has to undertake a majority vote action to select the character. It is not economical in link utilization.

____ " ____

The most practical system in CAMAC is error detection at the receiver from a transmitter-derived parity system, and the use of either the 'Dump' or 'ARQ' modes of operation described in (1) and (2) above.

6. PARITY STRUCTURE

A widely used method of error detection is to arrange the data in blocks with appropriate parity bits obtained from either long field parity or cyclic redundancy checks. When the block length is long there is little redundancy and good error detection performance is achieved. However in CAMAC applications it cannot be assumed that there will be a suitable data storage or processing intelligence at the terminals. Moreover, the assembly time for a block makes this approach too slow, and so a "block transmit" system is inappropriate.

In determining the parity structure, consideration must be given to the nature of the CAMAC terminal equipment and the path (d.c. links, lines or radio links using modems, etc.). Furthermore, most peripherals or peripheral ports on computers operate with an 8-bit byte format. It would therefore be useful to keep to this convention.

Moreover with synchronous modems, as soon as the receiver is synchronized with the transmitter, it puts out clock pulses to strobe the data stream. In the idle state the data output will be at a stream of mark (ONE) signals and any fully transparent system will treat this as a data stream of ONES unless special techniques are used to distinguish between idle and data ONES. Another problem with this type of modem is that synchronization can be lost if a continuous stream of ONES, ZEROS, or of reversals (depending on the setting of internal "idle state" links), are transmitted. The transmission system will have to avoid this problem while still maintaining full transparency for data transmission. Asychronous modems pose no such problem.

The parity structure proposed is of a two-coordinate type because this is the easiest to implement with the recent development of an 8-bit parity generator/checker which makes row, column and diagonal parity generation and checking a very easy matter.

The proposed system provides simple fast operation, 8-bit byte organisation and good security e.g. a conservative estimate indicates one probable undetected error in 5 years continuous operation at 2.4 k bauds. It also allows for full transparency, whilst avoiding synchronization problems.

By representing the 24-bit CAMAC data word as four 6-bit bytes, this particular device can effectively perform the modulo-2 sum of the data bits presenting both odd and even results. Both these outputs are used, one for bit 7 and the other for bit 8, forming an 8-bit byte, suitable for transmission, and with higher security than a single parity bit.

This also provides a solution to the synchronization problem, when a long string of 1's or 0's is transmitted. With bits 7 and 8 of each byte alternating there must be a change of logic state at least once every 8 bits, which is sufficient to keep the modem in synchronization.

The error control module will also organize stacking and unstacking for the full 24-bit words. It will also perform a column and/or a diagonal check, and add this as an extra byte, making a 24-bit CAMAC word into a 5 byte transmission format having 8 bits per byte.

Any two-coordinate parity system is proof against all one, two, three and higher odd-bit random errors, but will miss a few patterns of 4-bit and higher even-bit errors. The proposed system eliminates as many of these as possible and is proof against any burst errors except an exact inversion of all bits.

For one, two or three-bit errors the error coordinates are precisely known, and it is therefore possible to perform perfect FEC on these errors and inhibit retransmission thus increasing the possible throughput of the link.

7. TRANSMISSION SYSTEMS

For transmission of digital data over distances in excess of about 1 km there is almost no alternative to the voice frequency modem. Apart from a few special-purpose types of base-band modem, there are basically two types. Up to a speed of about 1200 baud asynchronous frequency shift keying (FSK) modems are often used. At higher speeds synchronous modems using phase modulation are typically used.

There is an internationally agreed CCITT V24 interface (3) for connection to a modem. However, its implementation is subject to some variations.

Ideally, data transmission systems should be arranged to maximise the true data throughput in relation to the information sent by the modem which includes parity bits etc. By this means communications costs associated with the much increased bandwidth which would otherwise be needed can be reduced.

For distances less than about 1 km, d.c. transmission systems are more economical and do not require VF modems. However in the Electricity Supply Industry high levels of electrical interference are encountered and it is possible to use a d.c. line driver over limited distances, with an opto-electronically isolated receiver to reduce these interference problems. The proposed coupler allows d.c. working at higher speed than a VF modem.

8. SYSTEM OPERATION

It is important that in the event of loss of communication between ends of the links satisfactory data transmission should be restored as soon as possible after the restoration of the communications links. There are two aspects of this to be considered. The first is the synchronization of the individual bits which is achieved within one bit by a fast acting servo loop. Secondly the data transmission system is restored by the use of the error control system.

Experience shows that one of the most common sources of trouble in data transmission systems is false detection of 'start' signals and logic must be incorporated within the module to reduce this as far as possible.

The use of a parity check in error control for an Automatic Request Repeat Transmission (ARQ) system relies on an error being detected by the receiver. A number of operating modes are used, all relying on one of a choice of three handshake bytes being sent by the receiver back to the transmitting end. These three characters ("acknowledge", "accept", "reject") must be recognized correctly despite possible errors having been introduced by the transmission path. However, it is possible to use an inherently self-correcting code for this byte as it does not form part of the fully transparent requirement of CAMAC data itself.

In the simplest mode of operation, received data is firstly checked for parity, and if valid is then offered to the CAMAC dataway. If it is found to be invalid it is dumped. For more important data, an "acknowledge" character is sent for each data word received, thus informing the original sending end that a character got to its destination. The full ARQ operation sends an "accept" handshake byte for each valid data word received. If the data parity check were not valid then a "reject" character is sent as a reply and this causes the original sending to continue sending the last character held in its buffer until it receives an "accept". While there is considerable time redundancy in this method it is quite a safe means of error control for systems of low intelligence.

In the event of a line outage in either direction, the system logic is designed to prevent "hang-ups" by the use of an "overwrite" flag in the transmitted data. If the last data word sent contains an error at the receiving end and a "reject" handshake byte is thus received at the sending end, then a flag in the next field transmitted indicates that this field is in response to a "reject" signal. The receiving end is arranged normally to accept this "overwrite" format only if the last field is in error (unless the modem circuits indicate a temporary incoming signal degradation or failure). Therefore, in the decision logic to decode the handshake byte it is safe to operate with a deliberate bias in favour of "reject".

A timer at the sending end is used to send repeats (in "overwrite" format) if no handshake is received in the normal round time of the system. This ensures that the data in the transmit buffer is cleared as soon as the path is restored in the event of a failure in either transmission path direction thus preventing any system hang-up. As the "overwrite" format used will be rejected by the receiving end (unless warranted) there will be minimum disturbance as a result of any outage.

In this system each item of data is delivered once only in spite of possible errors or outages of the transmission link which is an important requirement of systems using non-intelligent terminals.

9. BRIEF SPECIFICATION

The aim of the system design is to produce a single comprehensive device without incurring an excessive cost penalty in its use for less exacting applications.

The main properties of the device are as follows:-

9.1 Full CCITT V24 compatibility. Additionally "Fast Synch" circuit is provided.

9.2 Suitable for use with synchronous or asynchronous modems of any make or speed.

9.3 Simplex, Half or Full Duplex working, with Backward Channels.

9.4 8-bit word format for compatibility with peripherals and computer modem ports.

9.5 Self-synchronising servo loop for bit and word synchronisation.

9.6 Logic incorporated to reduce effect of false start signals.

9.7 Alternative facility for d.c. opto-electronically high speed transmission for short distances, using a self-clocking system.

9.8 Patch outputs for line condition monitoring.

9.9 Option for 24-bit working.

9.10 Full ARQ error control system option with 24-bit working. Handshake byte either by Backward Channel or interlaced into the fast data stream.

9.11 System logic incorporated to enable the module to restore operation as soon as possible after a path failure.

9.12 Interlace logic to enable throughput of a half-duplex link to be maximised.

10. CONCLUSIONS

It is possible to design a CAMAC module that will provide the features required for control of remote autonomous systems.

The features include:

(a) stacking and unstacking the 24-bit CAMAC word to an 8-bit compatible transmission format,

(b) rapid recovery from transmission errors and outages,

(c) high quality error control.

ACKNOWLEDGEMENTS

Acknowledgements are made to Mr R.E. Martin (C.E.R.L.) who initiated the design. The paper is published by permission of the C.E.G.B.

REFERENCES

(1) EUR 4100e, CAMAC, A Modular Instrumentation System for Data Handling, EURATOM, 1972.

(2) EUR 4600e, CAMAC, Organisation of Multicrate Systems, EURATOM, 1972.

(3) CCITT Recommendation V24 Functions and Electrical Characteristics of Circuits at the interface between Data Terminal Equipment and Data Communication Equipment. Geneva, 1964 - Blue Book, Volume VIII amended at Mar del Plata, 1968, - White Book, Volume VIII.

A SELF-CHECKING SYSTEM OF DATA ACQUISITION AND CONTROL USING A TWO BRANCH CAMAC HIGHWAY

III-16

M. G. Albrow, D. P. Barber, A. Bogaerts, B. Bosnjakovic, J. R. Brooks, A. B. Clegg, F. C. Erné, C. N. P. Gee, D. H. Locke, F. K. Loebinger, P. G. Murphy, A. Rudge, A. Lacourt, J. C. Sens, J. Timmer, CERN, Geneva, Switzerland; Daresbury Nuclear Physics Laboratory, Daresbury, England; Foundation for Fundamental Research On Matter, The Netherlands; University of Lancaster, England; University of Manchester, England; University of Utrecht, The Netherlands

ABSTRACT

A well tried CAMAC system used in a physics experiment at CERN is described. Information from various detectors is recorded on magnetic tape by an IBM-1800 computer. Real time analysis with interactive data sampling displays give the immediate experimental situation and automatic periodic checking ensures optimal running efficiency. Magnetic field settings and accurate mechanical positioning of large magnets and particle detectors are made from basic keyed-in experimental requirements. Our software philosophy shows that simple FORTRAN called subroutines enable complex control programs to be written by the non-specialist. The debugging routines assist the implementation of new facilities.

ZUSAMMENFASSUNG

Die Benutzung eines weitgehend erprobten CAMAC-Systems in einem physikalischen Experiment beim CERN wird beschrieben. Mit einem Rechner IBM-1800 werden die von verschiedenen Detektoren gelieferten Daten auf Magnetband aufgezeichnet. Die Echtzeit-Analyse mit interaktiver Sichtgerätanzeige von ausgetasteten Daten liefert Aufschluss über den momentanen Stand des Experiments; durch periodische automatische Kontrolle ist ein optimaler Wirkungsgrad für ein laufendes Experiment gewährleistet. Die Magnetfeldeinstellungen und die genaue mechanische Einstellung von grossen Magneten und von Teilchen-Detektoren werden den eingegebenen experimentellen Anforderungen entsprechend vorgenommen. Unsere Festlegungen in bezug auf den Software Einsatz zeigen, dass in FORTRAN aufgerufene einfache Unterprogramme das Schreiben komplizierter Steuerprogramme durch Nichtfachleute ermöglichen. Testroutinen unterstützen den Einsatz neuer Einrichtungen.

RESUME

Description d'un système CAMAC éprouvé, utilisé dans une expérience de physique au CERN. Les informations fournies par différents détecteurs sont enregistrées sur bande magnétique à l'aide d'un ordinateur IBM-1800. L'analyse en temps réel avec consoles de visualisation pour l'interrogation interactive de fichiers-échantillons donne immédiatement l'état d'avancement de l'expérience et un contrôle automatique périodique assure une efficacité optimale. La disposition du champ magnétique et le positionnement mécanique précis d'aimants de grandes dimensions ainsi que de détecteurs de particules sont effectués sur la base de conditions expérimentales fondamentales mises en machine. Le software choisi prouve que des sous-programmes simples appelés par FORTRAN permettent aux personnes non spécialisées d'écrire des programmes de contrôle complexes. Les programmes de mise au point aident à la mise en oeuvre des installations nouvelles.

A SELF CHECKING SYSTEM OF DATA ACQUISITION AND CONTROL USING A TWO BRANCH CAMAC HIGHWAY

1. INTRODUCTION

We describe major hardware and software aspects of a sub-nuclear physics experiment at the Intersecting Storage Rings (ISR) at CERN, Geneva (Al.72). The experiment is designed to detect particles produced at small angles from the collision of two high intensity high energy, proton beams continuously circulating within the ISR.

2. GENERAL DESCRIPTION

2.1. THE EXPERIMENT

A spectrometer of some 30 metres in length has been constructed, consisting of three Cerenkov detectors for particle identification, two septum magnets to "shave" the produced particles away from the intersection region into the spectrometer, and three bending magnets for main particle momentum distribution. Further detectors are placed opposite the spectrometer and around the intersection (fig.1). The entire experiment contains fifty phototube scintillation counters and thirty nine wire spark chambers.

The ISR consists of two intersecting rings with a diameter of 300 metres and to avoid health hazards it is located 10 metres underground. The proton beams are stored many hours for a physics run, and hence prevent any adjustment unless these are actuated by remote control over an effective length of 60 metres. This distance, expressed as propagation delays in coaxial cable, necessitated the development of the spark chamber trigger in logic modules in the Intersection Tunnel not accessible during any run.

Specially made units enable the experimenter to change the conditions of this apparatus. Over sixty NIM remote controlled coaxial cable delays are used; two of the Cerenkov detectors and the two septum magnets can be tilted or lifted to different angles with respect to the median plane of the proton beams; a status panel controls and displays the selected combination of Cerenkov and scintillation counters which define the spark chamber trigger.

The signals from these many detectors are further processed in a Counting Room, located directly above the apparatus in the ISR tunnel. Here the computer accepts the trigger as an interrupt signal ("event") and accesses the crates in the two branch CAMAC system (fig.2).

2.2. THE ON-LINE COMPUTER

The experiment is supervised by an IBM 1800 computer, which has a 24K core of 16 bit words, two discs and two magnetic tape units (fig.3). Under the surveillance of the IBM Time Sharing Executive (TSX), the computer performs four main tasks:

1. DATA ACQUISITION
   For each event several hundred words of physics data are read, preprocessed and written directly onto magnetic tape for later analysis on large computers.
2. ON-LINE ANALYSIS
   The incoming data is sampled and fully analysed in real time. This produces various CRT displays and typewriter summaries showing the status of the experiment.
3. MONITORING
   The many parameters of the experiment are periodically measured and compared with previously stored values. The computer interrogates the gas pressure and the gas quality transducers, temperature sensors, hall plate magnetic field detectors, and checks voltages and currents of the magnet supplies, phototube, spark chamber, CAMAC and NIM crate supplies.
4. SETTING UP
   Many parts of the experiment need setting or adjusting before a data taking can start. The computer can calculate the required settings and effect the necessary changes of magnet currents and mechanical position of the apparatus.

2.3. CAMAC HARDWARE

The spark chambers use "magnetostrictive" readout. A 20 MHz clock is started by the trigger, and stopped by a chamber output pulse, the delay time being proportional to the position of the incident particle. These clock pulses fill one channel of a CAMAC scaler. Multiple spark events are encoded by routing the clock signals to other scaling channels.

The experiment has over sixty 16 bit quad scalers, seventeen 16 bit pattern units, ten quad time of flight digitizers and twenty five input gates and output drivers. Many special units are used, accessing digital voltmeters, keyboard, ISR data link, a controller for the remote controlled delays (RCD's) and of course the 1800 CAMAC Interface and Branch Selector Units, developed by the Electronics II Group at CERN. This group was also responsible for the Interrupt processing modules, the LAM Graders.

The 1800 Interface has a block data transfer mode as well as individual CAMAC module addressing facilities. Data accessing times can be reduced by keeping all words to 16 bits.

At this point, the usefulness of the CAMAC test function F(25) and the Read Module Characteristic F(6) cannot be over-emphasized. We place much value in the self-checking and debugging routines.

2.4. USER FACILITIES AND DISPLAYS

An interactive system, using two storage CRT displays and a CAMAC keyboard, enable the user to communicate with the experiment. The computer's typewriter-keyboard and data entry switches provide back-up when necessary.

The CRT's are coupled directly to the computer, providing a handy fault finding tool when the CAMAC system is inoperative.

Over 50 different displays, examples of which are shown in figs.4 to 7 keep the user well informed as to the condition and status of the experiment. They are formed by programs that make samples and accumulate statistics from the incoming events.

The IBM TSX-system allows simultaneous use of the peripherals including the data links to the large remote computers (FOCUS(Da 73)) and the ISR control computer (ARGUS 600(IS 72.1)), giving important machine parameters.

3. THE CAMAC SOFTWARE

3.1. PHILOSOPHY

The main design criteria of the software were:

1) The CAMAC could be used for many different processes, with parallel access possibilities.
2) For reasons of flexibility it should be possible for each process to have control over all CAMAC hardware and software features.
3) It should be user orientated rather than machine orientated, to enable the non-specialist to rapidly implement tests, modifications or extensions to the experiment.

CAMAC is not only the heart of the data acquisition system, it is also used for process control in a time sharing environment. As a result one of the main problems that had to be solved was the avoidance of interference. Two or more processes are said to interfere if the correct execution of one is disturbed by the other. We can show that suitable structuring combined with the adoption of certain conventions can eliminate interference.

We have complete CAMAC control in the high level language of FORTRAN by a set of special "CAMAC routines", which have been written in IBM 1800 Assembler language.

3.2. INTERFERENCE

The various interference sources in a CAMAC system depend on the properties of its component parts, especially upon the control logic of interfaces, crate controllers, LAM Graders and similar units. It is almost impossible to find a general solution to the interference problem that can be applied to all practical CAMAC systems, and in a more closely defined system care must be taken to avoid restrictive use of CAMAC hence the loss of flexibility.

We list below some possible interference sources and show our solutions to these problems.

3.2.1. SOFTWARE RE-ENTRANCY

This is a well known problem that arises when a software module is shared by several processes which may modify it during execution. It is not specific to CAMAC and standard techniques have been used, such as interrupt masking and the separation of "pure"and "variable" code.

3.2.2. I/O CHANNEL CONFLICTS

Digital Input (DI) and Digital Output (DO) Channels transfer blocks of data between the computer and the CAMAC Interface in a cycle stealing mode (independent of the Central Processing Unit), see fig.3. A busy test is made before a channel is reassigned, and a time-out procedure prevents the lock-out of any requesting process.

Simple, short driver routines have been written for each digital I/O channel. In these routines called DIP and DOP, only two functions are allowed -"initialize transfer" and "channel busy test" - no interrupts are used to signal the end of the transfer. CAMAC can therefore be used on all interrupt levels of even higher priority than the peripherals. A flow diagram is shown in fig.8.

3.2.2. MEMORY EFFECTS

The CAMAC control hardware contains many registers that can be loaded by the computer. Their use is restricted by certain conventions. Some may be freely changed by any process (e.g. the crate address register in the interface, the SNR register in the crate controller), others need saving and restoring (e.g. single/block transfer mode, computer/CAMAC mode, all in the interface). A record of the current status for each interrupt level is maintained by the software. All this is done automatically by the routines of the FORTRAN package.

3.2.4. SEQUENCING

Most CAMAC operations and especially data transfers require several computer cycles, eg. a full read instruction BCNAF requires two cycles to address branch B, one cycle to address crate C and finally one cycle for the NAF portion. Furthermore several cycles are needed to open the computer input channel. All these instructions must be executed sequentially without interruption, otherwise the results are unpredictable. This leads us to the concept of "indivisable action" - well known in the theory of parallel processing.

The following activities (if applicable) are always required to form one indivisable action:

- the complete CAMAC address and command IBCNAF (interface-branch-crate-station-subaddress-function).
- all the CAMAC instructions between "enable non-standard feature" and "restore original status" inclusive.
- all data transfers.

3.3. THE FORTRAN CAMAC PACKAGE

The above techniques have been applied in a subroutine package, enabling the user to write his programs in FORTRAN without having a detailed knowledge of the CAMAC system and yet avoiding interference.

These routines are comprised of sub-tasks, each considered to be one indivisable action.

Examples:

CALL CNAF (ICR,N,IA,IF)

executes the CAMAC command GNAF.

CALL READF(ICR,N,IA,IF,DATA)

executes the CAMAC read command CNAF (with F = any CAMAC read function) and places the result in DATA.

CALL WRITF(ICR,N,IA,IF,DATA)

executes the CAMAC write command CNAF (with F = any CAMAC write function) where DATA is output to CAMAC.

Several hardware or software features can be enabled/disabled. It should be noted that at this level there is no direct correspondence between these modes and the actual hardware features, as seen by the programmer.

Six modes are supported by the software:

QR Q response read-makes all read data transfers Q conditional.
QW Q response write-as for read.
PR Precision read-single(16 bit, computer) or double (24 bit, CAMAC) precision for read instructions. The latter case requires two computer words for storage (a real variable in FORTRAN).
PW Precision write-as for read.
I Interface - selects interface 0 or 1.
B Branch-a maximum of seven branches is supported by the software.

These modes are packed into the CAMAC status word as shown

15															0
QR	QW	PR	PW	X	X	X	X	I	B7	B6	B5	B4	B3	B2	B1

There is one status word for each interrupt level. All modes are preset to values that are used by default if a program does not set a mode explicitly.

Modes can be set by certain subroutines eg.:

CALL QRESP (1) sets all read/write data transfers conditionally on Q,
CALL BSU (2) selects the second CAMAC branch.

These routines change only the status word (modes are "logically" set). All subroutines that perform CAMAC operations for which these modes are relevant examine this status word and take the appropriate action (eg. hardware features may be enabled/disabled, data may be reformatted etc.).

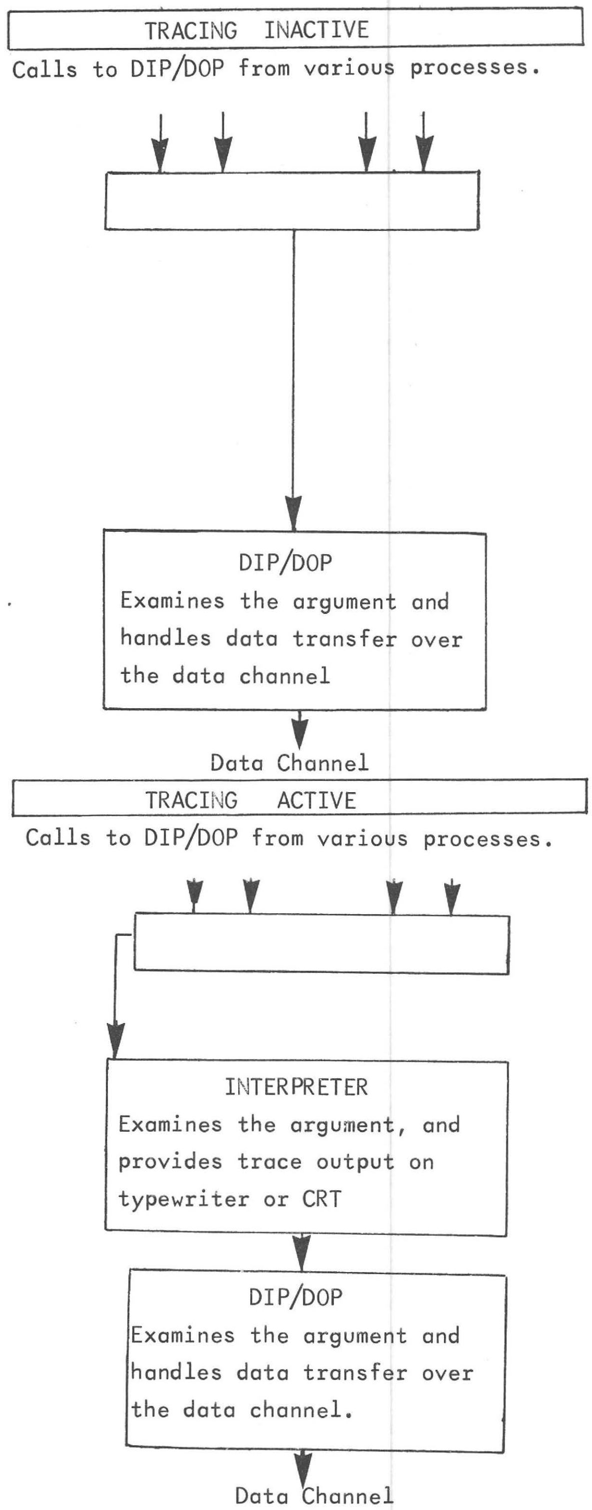
Finally we give the execution time of a few routines:

READF (read one CAMAC word)	2.5 ms
WRITE (write one CAMAC word)	1.5 ms
READ 2 (read 256 CAMAC words, block transfer) and for comparison	5.0 ms
floating divide (standard precision)	0.7 ms
floating square root (standard precision	5.5 ms

A comprehensive list of the CAMAC subroutines is given in figure 9.

3.4. THE CAMAC INTERPRETER

For debugging of both hardware and software an interpreter to trace all CAMAC operations has been implemented. It interacts directly with the two driver routines, DIP and DOP, to trace the CAMAC operations of all processes. To activate tracing these two routines are delinked from the calling chain and the interpreter is inserted. To start CAMAC tracing subroutine SPY should be called. It is stopped by calling SPYOF which re-establishes the original links.



Since all calls to DIP/DOP are caught, all processes including those activated by interrupts can be traced. The interpreter must remain in core during the time that the original links are broken. Trace output may be displayed on a CRT or typewriter. Data entry switches on the computer console enable various options to be selected:

DATA ENTRY
SWITCH NUMBER
0 All tracing temporarily suppressed. The links to the interpreter however remain, and tracing will be resumed when the switch is off.

1 The computer goes into the wait mode and displays the hexadecimal number 1111 on the computer console every time a call to the subroutine DOP is encountered. The computer console START must be pressed to continue.

2 The computer enters the wait mode and displays the hexadecimal number 2222 every time a call to the subroutine DIP is encountered.

4 Abridged output for DOP is provided.

5 Abridged output for DIP is provided.

14 Controls automatic "new page" on the CRT.

15 Selects typewriter/CRT for trace output.
For each call to DIP/DOP the following output is available:
- Address from which the call originated.
- Function selected by the parameter (busy test or input/output table).
- Status of the digital I/O channel at the moment of the call (busy/not busy).

If data transfer is requested the following output will be given in addition:
- Table starting address.
- Control word (this is the first word of the table and contains control information for the data channel).
- Table word count.
- Digital I/O register address.
- Data contents of the table.

For an output table (DOP) each entry is displayed as a hexadecimal value followed by its interpretation. For an input table (DIP) the address of the first data item is displayed in hexadecimal, followed by the data value in hexadecimal and decimal. The display is delayed until the data transfer is completed (busy test).

A short sample program with one read and write operation together with trace print out are shown in figs. 10 and 11.
Notes on figures 9 and 11.
Non-standard commands for the IBM 1800 CAMAC Interface (Ba 71.1).

F(CR=0/1,CW=0/1,T=0/1,M=0/1, $C(C_7,C_6,C_5,C_4,C_3,C_2,C_1)$)
The crate address $C_7C_6C_4C_3C_2C_1$ is loaded into the read or write register or both, depending on the CR (read) and CW (write) bits. Each C_i corresponds with one crate. Multicrate addressing is possible. The T and M bits control the block transfer mode.

F(Z) Initialize the entire CAMAC system.

MS/LS DATA One 24 bit CAMAC output word is coded into two 16 bit computer as shown.

15			0
0	1		12 bits Most Significant data
1	0		12 bits Least Significant data

Pseudo commands for the IBM 1800 CAMAC Interface:

N(0)A(4)F(26) Set double word transfer mode for read operations.

N(0)A(5)F(26) Set all read data transfers conditionally on Q = 1.

N(0)A(7)F(26) Transfer Q response for non-read commands.

N(0)A(10)F(26) Set all write data transfers regardless of the Q state.

Any mode enabled by F(26) may also be disabled by F(24).
Pseudo commands for the Branch Selector Unit (Ba 71.2).

N(31)A(0)F(26) Select BSU (the crate address is used as branch address).

N(31)A(0)F(24) Release BSU (the crate address again is used as branch address).

Pseudo commands for the LAM Grader (Is 71).

N(28)A(0)F(16) Write Graded LAM mask.

Pseudo commands for the Crate Controller type A may be found in (Es 72).

4. THE MONITORING AND CONTROL SYSTEM

4.1. INTRODUCTION

Figure 12 shows the monitoring and control system which is connected to crates in Branch two. Digital data is interfaced directly into CAMAC registers, analogue information is multiplexed into a CAMAC controlled digital voltmeter (DVM) via the analogue scanner.

4.2. SETTING THE SPECTROMETER

The two septum magnets of 2 and 20 tons respectively, and two of the Cerenkov counters, may be moved in the vertical plane and rotated about their axes. The changes in height and angle are about one metre and twenty degrees respectively.

Movement is effected by means of eight standard three phase motors with integral electromagnetic breaks driving screw jacks. The motors can be stopped within half a turn. High reliability is ensured by considerably underating the control components.

Each motor is shaft encoded into four decade Gray code. Positional repeatability is better than ± 0.5 mm. The parallel output of the encoder is multiplexed into a Gray to BCD to Decimal converter and displayed upon selection of an appropriate address. Manual up/down control may then be applied to the selected motor (Sc 70).

All eight motors may be positioned simultaneously from the computer via CAMAC. The required encoder positions are either calculated from the basic parameters-particle production angle and momentum-or the settings may be entered via the CAMAC keyboard. The position control program is then executed, see figure 13.

Notes on figure 13
(a) The program exits when all the encoders stop moving.
(b) Motor movement is inhibited unless a "give me control" bit is raised by the computer and a "computer control" bit is set simultaneously. This bit may only be set if the "give me control"

bit is already present. At the end of the program the computer resets both bits. Manual control is also inhibited during this period.
These safety features have been used elsewhere on the experiment.

(c) Parallel motor addressing gives shorter setting times than manual operation.

(d) Harware limits define the maximum excursion of any unit. They are designed to be "Fail Safe", and provide 100 percent cover, i.e. of height, depth and tilt. The presence of a limit condition is displayed at the end of the program.

4.3. MAGNET CURRENT CONTROL

The magnet currents (0 to 20 kA for the septum magnets and 0 to 800 A for the bending magnets), may also be controlled from CAMAC (Is 72.2). The operation is as follows:

A magnet address may be raised in a CAMAC output register, together with polarity, up/down and fast/slow bits. Two input registers are used to read the magnet currents and status information with power supply Read and master control veto.

A CAMAC controlled parallel setting system is now under development, that will set to the required magnetic fields using Hall plates mounted inside the magnets.

A flow diagram of the present system is shown in figure 14.

Notes on figure 14.

(a) A polarity change takes about 20 seconds during which time the supply is "Not Ready".

(b) The program aims at a higher current than requested to avoid magnet hysterisis errors. A test is made that the maximum supply current is not exceeded.

(c) Two setting speeds are used for the bending magnets and three for the septum magnets.

(d) Ten passes are requested to give sufficient "settling time" for the supplies. The large overshoot characteristics of the supplies required careful optimization of setting speeds and time delays.

4.4. REMOTE CONTROLLED DELAYS

A CAMAC interface (Ru 71) exists, enabling the remote controlled delays (RCD's) to be read or written into, under program control. Optimum delay values are determined manually and are latched into each delay station. Since this information is fairly volatile, it is backed-up by a permanent disc file. The control program has three operating modes:

(1) Read all RCD stations, compare with disc file and display.

(2) Update all RCD stations from the disc file and display the old and new values.

(3) Update the disc file from the RCD stations and display the old and new values.

A flow diagram of the control program is shown in figure 15.

4.5. THE CAMAC SELF CHECKING PROGRAM

This provides a "grass roots" check of the CAMAC hardware. The program is designed to show up any faults which could give false data during normal operation, and is executed periodically during running.

A general initialise is performed, then the Q-response and Module Characteristic of each station and sub-address in all branches is read and compared against a permanent disc file. This not only checks the NAF decoding of each module, but will show if a module has been stolen or misplaced.

An increment, read, and clear test is then made on all scaler sub-addresses and pattern units. At the end of the program the original status of the CAMAC hardware is restored.

The total execution time is about 30 seconds, all interrupts being masked.

4.6. THE ON-LINE CHECKING SYSTEM

The CAMAC system is also used to monitor the operating conditions of the experiment during running. This system checks most of the main experiment parameters thus simplifying the shift crew's task. Our system philosophy is based on the use of permanent and temporary disc files. As mentioned previously, the permanent files contain the RCD values and a record of the CAMAC hardware.

At the start of each run, a temporary file is set up by reading the current values of the experiment parameters, i.e. magnet currents, magnet positions, gas pressure etc.

Periodically during running, these values are checked by a program executed in parallel with data taking. A flow diagram of the program is shown in figure 16.

The program interleaves DVM reading with RCD checking to minimise execution time.

4.7. SCINTILLATION COUNTER EFFICIENCY TESTS

An assembly of eighteen counters is checked in the following manner:

The counter EHT, about 2.0 kV, is monitored with a simple potentiometer divider chain, multiplexed into the DVM under CAMAC control, and its EHT value displayed. A more sophisticated test is then made using a CAMAC controlled fast pulse multiplexer (JP 70). The nanosecond counter signals are fed into a CAMAC pulse height analyser system and displayed as a simple pulse height histogram in real time, see figures 17, 18.

Normal operation may be simulated by using nanosecond light pulses, timing and amplitude of which have previously been preset.

ACKNOWLEDGEMENTS

This work has been supported by the European Organization for Nuclear Research (CERN), the U.K. Science Research Council through the Daresbury Laboratory and the Foundation for Fundamental Research on Matter (F.O.M.) Holland.

The authors are grateful for facilities provided by these organizations, which includes the secretarial services provided by Mme M. Keller.

LIST OF REFERENCES

Al 72 M.G. Albrow et al., Physics Letters, Vol 42B,No.2 (Nov. 1972) 279 - 282.

Ba 71.1 CAMAC Interfaces and Their Users, CERN NP Type 057.
F, Bal, F. Iselin, A. Long, B.A. Löfstedt, A. Maurer, O. Ph. Nicolaysen, A. Orêve, Ph. Ponting, J.P. Vanuxem, Electronics II, CERN.

Ba 71.2 Branch Selector, Type 079.
F. Bal et al., Electronics II, CERN.

Da 73 Focus Project Notes 1-92 (1967-1973).
Data Handling Division, CERN.

Es 72 CAMAC Organization of Multicrate Systems-EUR 4600e
ESONE committee 1972.

Is 71 LAM Grader ("Look-at-me Grader"), Type 064
F. Iselin, A. Lang, B.A. Löfstedt, A. Maurer, O.Ph. Nicolaysen, A. Orêve, Ph. Ponting.

Is 72.1 Message Processor ISR-8101 (1972), ISR, CERN.

Is 72.2 Power Supply Control System for Experiments,
ISR 1525, ISR, CERN.

Jp 70 Remote Controlled Switch, Type NM 660
J and P Engineering Ltd., Reading, Berkshire, UK.

Ru 71 RCD Controller Type 152.
A. Rudge, NP Division, CERN.

Sc 70 Codeur de Position à Commande Electronique.
A. van der Schuren, P. Têtu, J.J. Aebi, L. Berrard, M. Sartori.
Electronique Industrielle No. 131, Mars 1970, Editions Radio Paris.

Figures are available on request to the authors.

GENERAL DISCUSSION

Q - N. Sakkinen

The testing and debugging of software for critical applications has not been discussed. Most speakers have mentioned an "easy programming facility"(FORTRAN, BASIC, etc.) in their implementations. But it really does not seem feasible to let everybody (or perhaps anybody) run their programs right away with the real peripheral apparatus. I would like to hear a comment on this problem, though it is not specific to CAMAC.

C - R. Rausch

In our system we effectively have a high level language that is used for scheduling of different tasks and also for messages between all the computers. At the real time level we have CAMAC drivers operating effectively in real time (I mean in the μsec domain range with the autonomous controller, if so required). If short drivers are used, then operations can be performed directly under control of the Sintran monitor. Obviously in a big system like ours there are problems in allowing everybody to write his own programs. People have to follow several rules fixed by the programmer of the language. If the linking of the different application programs is performed in a reasonable way the overall structure of the program might very well reduce a lot of the work for software development.

C - R. Campbell

At St. Andrews, the testing and debugging of operations is carried out by means of our extended BASIC interpreter. Once an operation has been debugged in this way it can then be included in FORTRAN or assembler routines.

IV

MESSEN, STEUERN, REGELN IN DER INDUSTRIE
MEASUREMENT AND CONTROL IN INDUSTRY
MESURE ET CONTRÔLE INDUSTRIELS

Präsident - Chairman - Président:

R. PATZELT, Technische Hochschule Wien, Austria

INDUSTRIAL COMPUTER CONTROL

K. G. Hilton
GEC-Elliott Process Automation Ltd, Leicester, England

ABSTRACT

Industrial computer control is an established market with a wide spread of problems and requirements. Input/output systems used in this market area are not normally able to cover all the requirements which the market demands. CAMAC has some features which could be of value and some which are not advantageous.

ZUSAMMENFASSUNG

Für Rechnersteuerungen in der Industrie hat sich bereits ein Markt entwickelt, der durch ein breites Spektrum von Problemen und Erfordernissen gekennzeichnet ist. Die verfügbaren Ein- und Ausgabesysteme sind nicht durchweg in der Lage, allen Marktanforderungen zu entsprechen. CAMAC-Systeme haben bestimmte Eigenschaften, welche hier von Nutzen sein könnten, und andere, welche keine Vorteile bieten.

RESUME

Le contrôle par ordinateurs utilisé dans l'industrie constitue un marché bien établi présentant une grande diversité de problèmes et d'exigences. Les sytèmes d'entrée/sortie utilisés sur ce marché ne peuvent pas, dans la plupart des cas, satisfaire toutes les exigences de ce marché. Certaines caractéristiques de CAMAC sont intéressantes, d'autres ne présentent pas d'avantages particuliers dans ce domaine.

There is nothing new about the industrial computer control market, it has been in existence for more than ten years and many companies, particularly the large electrical companies have been active for that time. In terms of CAMAC therefore, moving into this area is not as much a case of breaking new ground but rather moving into a field of activity which is established already.

In order to see how CAMAC can fit into this field, what advantages it can offer and what its shortcomings are it is important to see clearly what the industrial field looks like, what equipment it utilises and what design constraints apply to it.

The industrial computer control market is not, in fact a single entity with a common requirement. It covers a range of application areas from simple plant logging systems where the role of the computer is quite limited to massive systems applied in nuclear power stations and chemical complexes. The standards of construction required and environmental factors influencing the design of the plant communication hardware vary very widely.

It is useful at this point to consider some examples of these in various situations.

The examples are:

1. A logging system installed in a clean control room atmosphere

2. A chemical plant control with corrosive dust or vapour

3. Logging or control of mechanical plant e.g. turbines or steel mills.

These examples cover 3 common situations, the benign environment, the chemically hostile environment and the mechanically and electrically hostile environment.

Taking these examples individually.

1. Typically such a system will have a life of 5-10 years. The standards of mechanical design and construction are in general not arduous, the main purpose of enclosures being for protection against chance damage and for appearance. No great problems exist with temperature and humidity. Electrical design problems may be more variable. Many inputs will be obtained from "standard" transducers giving outputs in the commonly found ranges i.e. thermocouple outputs. 0-5 mA or 0-20 mV as examples, together with on-off inputs from contact or electronic sources.

   The equipment as supplied and installed is not normally subject to substantial change. The equipment or plant to which it is attached may be extended or modified at intervals, but if the extension is considerable, the user will in most cases buy a new system. Field extension flexibility whilst useful and perhaps important in some cases, is not a major design consideration.

2. The non-mechanically hostile environment is a common factor in many applications chiefly in the chemical industry. A great number of chemical processes have a corrosive or toxic product or use corrosive or toxic feedstocks at some part of the process. In cases of this kind, electrical design is relatively unaffected by the problems but packaging, enclosures, mechanical design, materials and manufacturing standards are all greatly affected by the environment. In some cases it is necessary to hermetically seal a system using recirculating cooling air, in other cases dust proof enclosures with good air filtration and corrosion resistant construction may be adequate. System inputs will be similar to those issued in 1.

   Such systems tend to be less subject to field modification particularly where extreme environmental protection is needed. Their lifetime is, of course, closely related to plant lifetime.

3. The mechanically hostile environments are those whose problems relate to vibration, humidity or extremes of temperature. These occur most forcibly in marine application where strong low frequency vibration together with shock loadings, humidity and in certain instances salt contamination calls for a very solid manner of construction. In terms of utilising equipment primarily designed to industrial standards, the main problem is prevention of resonances in the structure and wiring which can cause damage.

   In power plants and steel works, for example mechanical shock and vibration can still cause trouble although the problems are less severe than aboard ship. Well designed industrial equipment is adequate in such cases. Nevertheless in any situation where vibration is present attention must be paid to resonances and to anchoring components, equipment and cable forms adequately.

   In this kind of application, typically in industries using electrical power in quantity, severe problems can and frequently do arise from electrical interference. This can be mains borne, introduced in cable runs or caused by magnetic or electric fields present in a plant. Good layout of input/output equipment, particularly in relation to wiring harnesses, earthing practice and plant cabling, together with well designed equipment able to withstand appreciable common mode interference are essential in keeping this problem under control.

These general requirements do not, of course give rise to a single equipment specification, far less a single equipment design. It is, of course, possible to apply a single range of equipment across the spectrum of application but not economically. Manufacturers have designed equipment to suit their business pattern and it is found that there is a penalty in configuration size and in cost in applying equipment outside its design area.

Companies with a substantial business in, for example, power system automation will have designed input-output equipment with the capability of handling large numbers of points economically. To apply such equipment to a 150 input logging system or a 60 loop DDC is not likely to be a satisfactory economic solution. Equally systems designed with the small system as the prime business target will be likely to be cumbersome on large systems.

Compatibility as between different manufacturers in general does not happen, where it does it is coincidental and not a design consideration.

CAMAC, of course, is not in the same category as proprietary manufacturers equipment and by its nature avoids certain of the difficulties outlined above. The pitfall it avoids is that of compatibility. It sets out to be a machine independent standard interface and to this extent it clearly succeeds.

From the point of view of the system vendor, concerned with selling complete systems and carrying responsibility for the successful implementation of the system, this feature of CAMAC is not a first consideration. Unless the end user has a strong view, which may be based on such consideration as commonality of spares holdings, training of maintenance personnel or a wish to implement undefined system extensions himself, the system vendors decision on whether or not to offer CAMAC as process input/output equipment must be based on other considerations.

The principal other general consideration which he must bear in mind is that CAMAC suffers from the same generic problem as all input-output systems, proprietary or otherwise. There are some applications for which it is suitable and others for which it is not so appropriate. Just as other, proprietary, systems can be used in areas for which they are not entirely suitable, so can CAMAC but there is a penalty in complexity and in cost in so doing.

The facilities which CAMAC has, resulting from its use in experimental equipment, for setting up a configuration quickly and in having front access to modules to facilitate connection for example to nucleonic instrumentation present particular problems in industrial systems where quick changes to configuration are not necessary. In order to deal with the fact that modules carry connections both at front and rear, actual wires or ribbon must be used for one set of connections, whereas another board construction would be preferable. This intermodule wiring which in general will be considerable must be organised properly to prevent electrical interference problems and to give maintenance access to the modules.

Where equipment is being supplied for the end user to implement the system, the constraints are different. The fact that CAMAC is a standard system is more relevant. A user may have more knowledge of CAMAC than of proprietary systems and therefore feel better able to configure and program systems in CAMAC. He will therefore spend less on system implementation than he otherwise might.

This brings us to the centre of the question - costs. If a user believes that to have a CAMAC implemented control system in his plant will save him money either in the short term or in the long term then this is an important factor for him to use in comparing bids for such a control system.

The comparison between CAMAC implemented systems and proprietary systems in price as bid is not easy to make in the general case. Indications to me are that in many industrial control situations CAMAC is more costly.

Taking these two factors together, in some cases, the economic solution will be CAMAC, in other cases it will not be CAMAC.

If these economics are generally favourable to industrial CAMAC, then it will undoubtedly prosper, if the economics are not generally favourable then its application will not be as widespread and will be based chiefly on user preference, processor independence, where this is needed and on special cases.

The author wishes to thank the directors of GEC-Elliott Process Automation Limited, a GEC-Elliott Automation Company for permission to publish this paper.

DISCUSSION

Q - J. Robin

Concerning your third example: How do you think it would be possible to overcome the problem of interference?

A - K. Hilton

This is a problem which is by no means unique to CAMAC systems and to which CAMAC is no more susceptible than proprietary systems. It is a problem which can be very difficult to solve and there is not a single solution. Actions which must be taken to avoid or minimize interference are -

1) Careful separation of signal, power and control cables.
2) Twisting or screening or even using coaxial signal leads.
3) Good and careful earthing.

In short, good engineering practice and good housekeeping of the electronics is the proper course to take.

C - R. Patzelt

This is a general problem, having equal significance to CAMAC and conventional computer I/0-systems.

Q - J. Chmielewski

Is it possible that CAMAC will be helpful to bridge the gap existing between laboratory analytical instruments (e. g. nuclear magnetic resonance spectrometers) and on-line process computer controlled systems?

A - K. Hilton

CAMAC would certainly be helpful in situations where on-line analytical or other complex instruments of a similar type are required for use in a computer control system.
This area has tended to be costly to implement on individual systems using proprietary equipment and a system using CAMAC would certainly be possible. It would ease the interfacing problems in such a case, provided only that there is no reason elsewhere in the system why CAMAC could not be used. In the sense in which an on-line system using analytical instruments bridges two areas of activity, and CAMAC can be used in such a situation, it will be helpful in bridging the gap.

AN EVALUATION OF CAMAC EQUIPMENT IN AN INDUSTRIAL ENVIRONMENT

W. T. Lyon
Aluminium Company of America, Pittsburgh, USA

ABSTRACT

The Aluminium Company of America has been looking at the CAMAC standard as the basis for a standard process control interface to be used within Alcoa. Preliminary studies have given favorable answers but some questions remain. A pilot CAMAC installation was made on an existing process with the objective of answering the following questions:

1. Is CAMAC suitable for Alcoa's use ?
2. What compromises, if any, must be made to make it usable ?
3. Is it really computer independent ?
4. What are the software problems associated with it's use ?
5. Should Alcoa's concept of the computer/process interface change ?

This paper is a report on the results of that evaluation.

ZUSAMMENFASSUNG

Die Aluminium Company of America untersuchte, ob CAMAC die Grundlage für eine Standard-Prozessteuerungsschnittstelle zur Verwendung bei Alcoa darstellen kann. Vorläufige Untersuchungen zeigten günstige Ergebnisse, einige Fragen sind aber noch offen. An einem bestehenden Prozess wurde eine CAMAC Pilotanlage installiert, um Antworten auf folgende Fragen zu erhalten:

1. Ist CAMAC geeignet für den Einsatz bei Alcoa ?
2. Was für Kompromisse müssen gegebenenfalls getroffen werden, um es brauchbar zu machen ?
3. Ist es wirklich Rechner-unabhängig ?
4. Welche Softwareprobleme sind mit seinem Einsatz verbunden ?
5. Sollte Alcoa's Konzept der Rechner/Prozess-Schnittstelle geändert werden ?

Der Vortrag berichtet über die Ergebnisse dieser Untersuchung.

RESUME

L'Aluminium Company of America étudie la possibilité d'utiliser, pour ses propres besoins, le standard CAMAC comme base pour une interface normalisée dans le contrôle de processus. Des études préliminaires ont donné des réponses favorables, mais quelques questions restent en suspens. Une installation pilote CAMAC a été faite dans un processus réel pour permettre de répondre aux questions suivantes :

1. CAMAC correspond-t-il aux besoins d'Alcoa ?
2. Si besoin est, quels compromis sont à faire pour pouvoir l'utiliser ?
3. Est-il réellement indépendant du type de calculateur ?
4. Quels problèmes de programmation se posent lors de son utilisation ?
5. Alcoa doit-elle changer son concept d'interface calculateur/processus ?

Cet article fait le point des résultats de cette évaluation.

1. Why is Alcoa Interested in CAMAC?

1.1 Alcoa has long been an active supporter of Industrial Standards. We have become interested in CAMAC, a standard created for use in the high-energy physics laboratory, because we think it has something to offer us.

1.2 Alcoa is one of the larger industrial users of computers, with over seventy process computer installations. These computer installations, representing more than 30 different types of CPUs, give us some of the larger computer-generated headaches. This proliferation of computer types brings with it a multitude of training and maintenance problems. The use of high-level languages such as Fortran helps ease the software problem, but performing process I/O is a stumbling block. Even when FORTRAN calls to process I/O subroutines are standardized (such as the work outlined in ISA Standard S61.1), a person still needs to have a fairly intimate knowledge of the process I/O equipment on his particular computer.

1.3 On the maintenance side, one of our plants, with seven different CPU types, relies heavily on its own maintenance department and has stocked spare parts to keep its computers running. Having seven different computer types in the plant presents quite a challenge for the maintenance technicians, but this challenge often becomes a nightmare. Some of our plants rely on *electricians* to perform first-line maintenance on process I/O equipment. This would be desirable everywhere, but is impossible because of the training problems.

1.4 In short, Alcoa finds itself faced with a large number of computer mainframe types, a desire to have highly mobile computer personnel, and a gigantic training and spare parts problem. We are unwilling, on a company-wide basis, to become committed to a single computer source (which is a different form of standardization), so we are looking for alternatives to achieve a majority of the benefits that could be achieved by standardizing on a single computer vendor. We feel that use of a standard process interface, supported by high-level languages, is one of these alternatives. We are seriously looking at CAMAC as a standard for our use.

2. What Is To Be Evaluated?

2.1 It would be easy to say, "CAMAC is the only internationally-recognized standard, so let's adopt it". We cannot afford to blindly take this approach for fear it really might not fit our needs. We feel the following areas need to be looked into and compared with our existing and projected requirements:

- Software problems
- Noise susceptibility
- Hardware suitability
- Hardware and software portability
- Vendor and user acceptance
- Cost to use
- Training and spare parts requirements

2.2 All of these evaluation areas need to be looked at with the following questions in mind:

1. Will CAMAC allow us to do our projects more economically?
2. Will it allow us to do projects with fewer people, or conversely, to do more projects with the same number of people?
3. Will it be a long-lived standard?
4. Is the standard adequately administered and maintained?

2.3 Affirmative answers to all of these questions would nudge us in the direction of committing ourselves to broadscale use of CAMAC equipment throughout Alcoa.

3. Our Approach

3.1 Since a multifaceted evaluation was to be conducted, a parallel approach was adopted to perform the evaluation. Some of the areas of consideration, such as vendor and user acceptance, are quite nebulous and have to be approached in a unique fashion. The overall evaluation consists of the following elements:

Performing a preliminary evaluation of the standard.

Writing specifications for modules needed for our requirements.

Communicating with existing CAMAC equipment vendors.

Communicating with CPU manufacturers.

Communicating with instrumentation and transducer manufacturers.

Defining software techniques and writing software drivers.

Trying CAMAC equipment and software in a real, on-line, industrial application.

4. Results

4.1 Preliminary Evaluation:

4.1.1 The preliminary evaluation has mainly consisted of closely examining the standards, EUR-4100e and EUR-4600e, for content to see if there are any glaring deficiencies which we feel cannot be tolerated. There appear to be two vulnerable areas which are quite basic to CAMAC: (1) The speed of the Dataway; and (2) The gyrations required to do programmed I/O. Since a Dataway cycle requires at least one microsecond, much slower than core memory access times of modern day computers (a commonly-used benchmark), there were some serious doubts as to the ability of CAMAC instrumentation to keep up with processes running a high number of control loops at millisecond rates. In looking more closely at some of the processes in question, we found that *we haven't a single installation in which the control computer is process-I/O bound*, i.e., one in which the computer can't acquire data fast enough to satisfy the needs of the process. Even if control of higher-speed processes were attempted, the speed of the computer is more of a problem than the speed of the CAMAC dataway. Today's computer would require a minimum speed increase of 10-fold, to be able to do any meaningful manipulation of process information acquired at maximum Dataway speed.

4.1.2 Programmed I/O can take as many as three times as many instructions per data transfer in a CAMAC system compared with conventional I/O. If this presents a problem, it can be circumvented by using direct memory transfer or more computer device codes per crate.

4.2 Hardware Suitability, Availability & Specs.

4.2.1 A quick look at what was available in the market place convinced us that virtually no existing modules were suited to our applications. Vendors' catalogues talked of such items as scalars, blind scalars, word generators, etc., which meant nothing to us. In addition, most modules already on the market, which were *functional equivalents, were not interchangeable* because of small hardware and software differences.

4.2.2 A closer examination convinced us that nowhere did there exist anything that would talk directly to our big, ugly motor starters and dirty, corroded relay contacts or anything else usually found in industrial environments. They couldn't talk to our devices because they were not designed to do so, but here was where the revelation came.

4.2.3 Our first thought had been to look for general-purpose equipment as we know it today, but packaged in the CAMAC format. General purpose equipment such as digital inputs and outputs and analog inputs and outputs almost always requires additional signal conditioning interposed between it and the process transducers. Instead, it was found that traditional CAMAC users attack the entire *instrumentation* problem and design special purpose modules to talk directly to their measurement devices, thus leaving out the interposing signal conditioners. This approach hadn't occurred to us in the past because of the cost of the special engineering required of the computer manufacturer.

4.2.4 With this information we began preparing specifications for the types of modules needed to control and monitor our processes, keeping as our objectives the requirements that, whenever possible, *the CAMAC module should talk directly to the process transducer, and that functionally equivalent modules should be both hardware and software compatible.* Some of the specifications were sent to vendors for quotations. Whenever a manufacturer showed a willingness to develop a module at his expense, one or two were purchased for evaluation. To date the following modules have been purchased:

KineticSystems

(1) 24-Input Contact Sense Module
(2) Pulse-Duration Demodulator
(3) 8-Output Triac Module
(4) Crate Controller for Modcomp III Computer
(5) Manual Crate Controller*
(6) DataWay Display Module*
(7) Standard CAMAC Crate, Powered

Joerger Enterprises

(8) A-1 Crate Controller*
(9) Synchro Receiver Module
(10) Operator's Meter Driver

Bi Ra Systems

(11) Operator's Meter Driver
(12) Manual Input/Output Module

*Items 5, 6 and 8 were not developed as industrial CAMAC modules.

4.2.5 Potential problems found with the modules we have received center around the use of the recommended front panel connector and with the commonly-used practice of tying logic common to the chassis. Because of conductor spacings, maximum conductor sizes, signal levels, cable assembly time and power densities, the 52-pin double-density connector is not suitable for industrial control. In addition, while very high frequency circuits, such as found in the front-ends of NIM modules, may require circuit common tied directly to the module housing, this is not necessary at dataway speeds. Although not part of the CAMAC standard, most CAMAC manufacturers to date have integrally tied circuit common to the housing in all of their modules. For most industrial applications the circuit common *cannot* be tied to the CAMAC housing in every module. This is a serious problem in our applications and, as a result, we cannot use existing CAMAC modules (e.g., A-1 controllers) without modification.

4.3 Communication With Vendors and Users

4.3.1 We have had extensive communication with vendors and other potential users. We wanted to become familiar with existing CAMAC vendors and to see how well CAMAC is known among those not currently manufacturing or using it. We also wanted to inform as many as possible about the existence of CAMAC, hoping that if we were to adopt it there would be many others who would also use it. In addition, these contacts give us information about the availability or potential availability of CAMAC equipment and an idea as to its acceptance by other users.

4.3.2 Our activities have consisted primarily of:

(1) Writing specifications for modules and sending our requests for quotation to existing CAMAC vendors
(2) Sending explanatory letters and copies of module specifications to our traditional computer equipment suppliers
(3) Presenting technical papers and appearing on panel discussions at technical conferences
(4) Making personal contacts with vendors, prospective vendors and users
(5) Participating in industry standards efforts

4.3.3 From existing CAMAC vendors the response has been a willingness to work with us in the definition and development of modules. This can be seen by the above list of industrial modules which have been purchased. All of these modules were developed by the vendors at their expense. This willingness to invest their own time and money was done in anticipation of a developing market. To Alcoa, this translates into equipment availability. Our recent projects have had short lead time and have required off-the-shelf availability of equipment. Most could not wait for a long development period. Therefore, vendors' willingness to bring industrial modules on the market ahead of in-hand orders is a necessary and encouraging sign.

4.3.4 Response from many *computer* vendors, as might be expected, has ranged from indifferent to negative. This is true primarily of computer manufacturers who build their own process Input/Output equipment. Even in this area, though, we see some large computer manufacturers and several small ones giving CAMAC a very hard look for possible integration into their product lines.

4.3.5 Contact with instrumentation and transducer manufacturers has sparked some interest on their part in packaging their electronics packages on CAMAC modules. For example, a manufacturer of magnetic speed pickups might mount several conditioning circuits, consisting of amplifiers, shaping and counting circuits. Traditionally, this electronics package would produce an analog signal which would then have to be converted to digital by an ADC for subsequent input to the computer. As a CAMAC module, the counting register would be a CAMAC address and be input directly to the computer without the conversion to analog and back. For the most part, the transducer manufacturers have been cool to the idea of repackaging their equipment *until they understand that we are talking about a standard format*. Then they see a potential market, *one they would like to be in*, opening up to them. We consider this an encouraging sign because it allows a CAMAC vendor, in essence, to enlarge his engineering staff.

4.4 Software Considerations

4.4.1 Keep in mind that one of our objectives in adopting CAMAC, or any other standard, will be to minimize the people problem associated with process computer projects. The personnel training and the special engineering content of each project must be minimized. This means that a maximum amount of vendor-independence must be achieved. Among other things, this dictates the use of a high-level language for applications programming.

4.4.2 The only suitable language available today that offers enough standardization to allow portability of programs between CPU types is Fortran. Fortran is not a particularly good language for process control, but since it is the only thing available, we have based much of our software evaluation of CAMAC on its compatibility with Fortran. Most of the conclusions drawn will also apply to other procedural languages.

4.4.3 To put things more in perspective, a better understanding of basic hardware/software dependencies is required. Figure 1a illustrates the relationship that usually exists between hardware and software. This simplified view can imply that computer control *must* be computer-dependent. If these relationships are analyzed more carefully, and in the context of a standard I/O interface, the result is Figure 1b. This figure suggests that the only software element which must be computer-dependent is the computer-I/O bus driver. This is somewhat idealistic but we believe that, if care is taken in structuring the functional software handlers, this ideal can be very closely realized.

4.4.4 The bus driver routine shown in Figure 1b is CMCBSC, the NIM-CAMAC Working Group's proposal. Although the functional module drivers could talk to the I/O bus directly, they would then be computer-dependent. This is not true if they are written in Fortran and communicate with the I/O bus through CMCBSC. CMCBSC is used in our pilot evaluation, but the long argument list accompanying its call requires too much overhead when used in more demanding control applications.

4.4.5 In defining the functional software handlers, we were tempted to use a one-to-one relationship with the functional hardware modules. However, it was found that modules have overlapping requirements that allow the definition of generic functions which relate to the *ultimate use* of the data being transferred. A Fortran applications program needs to transfer the following types of data: (1) single bits of digital information; (2) groups of bits and (3) integer data (an arithmetically related group of bits).

4.4.6 An example of a functional handler is:

ICHINT (loc)

This Fortran *function* subprogram is an Input CAMAC Handler for an INTeger value, where loc is a pseudo address number for a hardware register in a CAMAC module. The actual address information (Branch, Crate, Station Number, and Sub-address) would be found by the handler/driver in a look-up table. The look-up table would be created during system generation or another system initialization procedure.

4.4.7 Another functional handler example is:

CALL OCHGRP (loc, msb, lsb, istats)

This Fortran Subroutine outputs a group of bits to a CAMAC register at location loc. The bits are consecutive from msb to lsb, and their new states are represented as an integer in the value of istats. A complete description of the functional handlers and other software considerations will be the subject of a later paper.

4.4.8 An alternative method of communicating with the CAMAC modules employs the use of an "I/O scan package". Such an approach usually has one program (the scan package) running autonomously, which inputs and outputs process data to and from an area in memory. The scan package then becomes the I/O bus driver of Figure 1b and the Functional software handlers become subroutines which access the data acquired by the scan package. With these two layers of software considered as a unit, their implementation method becomes transparent to the application program. Therefore, the engineer/programmer needn't concern himself with the actual means by which he will get his process data, as long as he knows the proper calls to use in his program.

4.4.9 To summarize our findings with CAMAC associated software, we have found most potentail problems not to be CAMAC problems, but problems associated with the use of high-level languages. The true CAMAC problems could be termed more as minor irritations than problems. These have mainly to do with the addressability of CAMAC equipment. Because of the multiplexing possible on a branch, the multitude of registers that can be addressed in a single crate and the numerous functions that can apply to a single module, more computer I/O cycles are usually required to access CAMAC hardware than hardware which is less flexible. On the surface, this looks as if the time required to do I/O could double or triple even if driver overhead is excluded. We found this to be true, but believe that it is tolerable. More than enough time-reduction should be gained by functionalizing the modules and letting hardware do some of the work usually done by software, to offset any increase in actual I/O time.

4.5 Pilot Evaluation of CAMAC

4.5.1 At this writing, our pilot evaluation has not been completed. Its delay is due primarily to late deliveries in equipment. This is probably not too surprising because of the developmental nature of the project. Three new modules had to be developed concurrently with the development of the computer controller.

4.5.2 The computer chosen to test the CAMAC equipment is a Modular Computer Systems Modcomp III. It is currently being used in an energy-management application. The CAMAC equipment will be used in conjunction with a program to parallel one function of this system, that of monitoring natural gas usage, projecting total usage for the 24-hour demand period, and alarming if the projected limit exceeds the allowed limit.

4.5.3 The purpose of this project is primarily to provide a meaningful vehicle by which to verify our analysis of how to use CAMAC. The concepts that will be tested are:

1. Use of functional modules
2. Proposed software techniques
3. Termination methods
4. Computer independence
5. Cost

In addition we will be using the hardware in an actual industrial environment. This may not produce any conclusive evidence, but will provide an opportunity to get a better feel for its susceptibility to electrical noise.

4.5.4 Results to date are limited, but two items of interest can be reported: (1) One special-purpose module was developed for this project. It is a pulse-duration demodulator and was designed to look at a contact-closure off a telemetering signal from a differential-pressure transducer. The length of time the contact is closed during a 15-second interval corresponds to natural gas flow rate. Deriving this information from the contact closure with software requires a rapid scan of the contact for resolution and a great deal of software logic to obtain synchronization, error checking and time-duration of the closure, all of which requires about 2,000 words of core memory. (The program was coded in Fortran.) Using the pulse-duration demodulator to perform these functions has reduced the memory requirement by an order of magnitude and the scan rate was reduced from once every 50ms to once every 15 seconds. This example provides a beginning for verifying the usefulness of functional modules. (2) Preliminary cost data shows that general-purpose modules such as the 8-bit Triac Module and the 24-Bit Contact-Sense Module cost virtually the same as their traditional counterparts. These comparisons are made on a per-bit basis and take into account the cost of the crate, power supply and the controller, as well as the cost of the module itself.

4.5.5 When a more specialized module such as the pulse-duration demodulator is examined on a per-bit basis, its cost seems exorbitant, but it becomes a bargain when the savings in memory and software effort are considered.

5. Summary

5.1 Most of the results reported were from our preliminary investigations. These indicate that there is nothing basically wrong with the CAMAC standard itself. Common implementations of the standard don't apply too well to industrial applications in some cases, but this can be solved by specifying more appropriate implementations. CAMAC vendors have been very responsive to this activity and we are encouraged by what they have accomplished to date.

5.2 Our cost estimates indicate that general-purpose CAMAC equipment shouldn't be any more expensive than comparable conventional I/O equipment. Moreover, the overall costs, compared on a function-for-function basis, in most cases favor the use of functional modules. This type of module is not economically practical unless it is used with a standard I/O interface such as CAMAC.

5.3 Nothing can be reported at this time about noise susceptibility, but we see no reason to believe that CAMAC equipment will be any different from other computer equipment.

5.4 Software investigations have probably done more to point up the problems associated with using a high-level language for applications programs than of the deficiencies of CAMAC. We have found that if a standard I/O interface is used, the design of high-level-language-oriented software is a much simpler task.

5.5 Contacts with vendors and other industrial users give us an indication that there is some interest in CAMAC, but our contacts haven't revealed a non-CAMAC vendor or non-CAMAC user that has made a firm committment to either produce or use CAMAC equipment. Before Alcoa would commit to using this equipment, we would need stronger indications than what we are now receiving that others will be doing the same. However, judging by the contacts that have been made, this could be just around the corner.

Hardware - Software Relationships

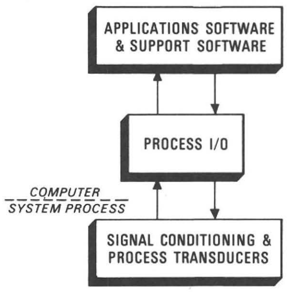


Figure 1a

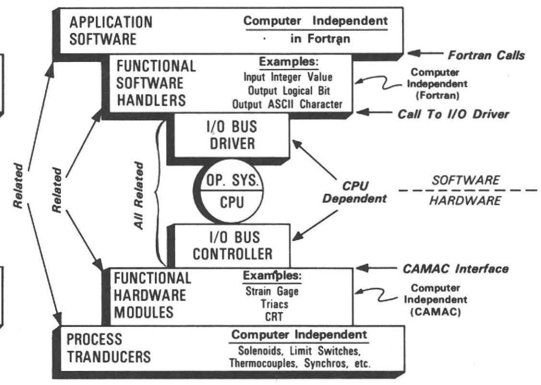


Figure 1b

DISCUSSION

Q - H. Klessmann

You mentioned that for industrial applications one of the recommended front-panel connector types (Cannon double density) is a potential problem.
Has ALCOA made a systematic investigation on multipin-connectors and what are your suggestions for a reliable alternative in process control applications?

A - W. Lyon

ALCOA has not made an exhaustive study of connectors. We have done some exploratory investigations but don't have any recommendations to make at this moment in time.

Q - M. Kinder

It is probably easier to achieve full interchangeability and multiple sources of supply with "general purpose" input-output modules than with more "functional units".
The more detailed the function - the less likely the module is to be exactly duplicated by another manufacturer.

A - W. Lyon

I have to agree that the more specialized the function, the less the availability of functionally equivalent modules. The extreme, of course, is the case where only one vendor supplies a particular module. With CAMAC I believe I can feel comfortable with the knowledge that I can have that module built by another vendor if my prime source fails.

Q - I. Lilja

We are paper machine builders. The problems facing us in process control of paper mills are not related to the lack of methods of interfacing a computer to our measurement devices. There is only one big problem; that is the lack of exact theoretical process models. However, we are going to use CAMAC in order to get a direct link between a test paper machine and our computing centre. So, CAMAC will help us in developing the process models, which must be exact enough before a computer can be introduced in total process control.
I would like to ask if ALCOA already knows all the process models which are needed in industrial computer control systems?

A - W. Lyon

We have developed a number of process models describing metal deformation processes, smelting and refining processes. I can't comment generally on the quality or accuracy of these models, because I don't know for sure. It has been our experience that an exact process model is not required if some form of adaptive control is employed.
Therefore, about all I can say for our process models is they are good enough to achieve adequate results as long as we help them out with adaptive control.

C - K. Hilton

I recommend strongly applying adaptive control rather than predictive control.

C - P. Wolstenholme

I should like to comment on the connection of industrial cabling to CAMAC module front panels.
At the CERN intersecting storage ring project, we have to meet an unusual combination of laboratory-style flexibility with industrial-type reliability. A false move in our control system can cost 100,000 - Swiss Francs!
We have chosen to take up a whole 19" rack for each CAMAC crate, and we mount "patch panels" above and below the crate. The Cannon, or preferably Hughes, front panel connectors are linked by short, factory-wired links to the front of the "patch" chassis, and heavy cabling is then brought to the back. Although there are more connections to go wrong, we gain by having all the delicate work done in well-controlled conditions.
Our CAMAC racks have doors on the front, for improved security, and we use a separate power unit in its own ventilated crate, which can be seen at the exhibition outside.

CAMAC ALS GRUNDLAGE EINES SYSTEMS FÜR DEN EINSATZ ALS RECHNERHERSTELLER UNABHÄNGIGE PROZESSPERIPHERIE

L. Magyary
Gesellschaft für Nuklearmesstechnik und Industrieelektronik (GNI)
M. Knüfelmann & Co. mbH, Leverkusen, Germany

ZUSAMMENFASSUNG

Der Aufsatz beschäftigt sich mit einer Interpretationsvariante des CAMAC-Systems. Basis des System RUPI (Rechnerhersteller unabhängige Prozessperipherie für Industrieausrüstungen) ist die Empfehlung EUR 4100 im Bezug auf seinen Dataway mit seinem Timing und der Mechanik. Bei den Modulen beginnt die Modifikation dort, wo wissenschaftliche und industrielle Nutzung ihre charakteristischen Forderungen stellen.

ABSTRACT

The paper deals with a variation in how the CAMAC-system is interpreted. The basis of the RUPI system (process peripherals for industrial equipment that are independent of computer manufacturers) is recommendation EUR 4100 as far as its dataway, timing and mechanics are concerned. The modification to the modules comes when scientific and industrial users make their usual characteristic demands.

RESUME

Le présent exposé porte sur une variante d'interprétation du système CAMAC. Le système RUPI (Rechnerhersteller unabhängige Prozessperipherie für Industrieausrüstungen - Unité périphérique de traitement pour équipement industriel, indépendante de la marque de fabrication) a pour principe fondamental le rapport EUR 4100 concernant l'interconnexion avec son système de synchronisation et avec l'équipement mécanique. Dans les modules, la modification est effectuée aux points où l'exploitation scientifique et industrielle impose ses exigences.

1. Allgemeines

Die industrielle Rechneranwendung im Zusammenhang mit der Produktionsautomatisierung weist wesentliche Unterschiede sowohl gegenüber der kommerziellen- als auch der wissenschaftlich experimentellen Datenverarbeitung auf.

Oft ist der Einsatz eines Rechners eine politische Entscheidung, hingegen ist der Einsatz von Kleinrechnern eine Frage des Preis/Leistungsverhältnisses.

Die Entwicklung des Kleinrechner- bzw. Peripheriemarktes einerseits und das Verlangen der Industrie nach modernen leistungsfähigen Produktionsmitteln andererseits, führten zu einem System, das im wesentlichen als eine Rechnerhersteller unabhängige Prozeßperipherie (für Industrieausrüstungen) eingesetzt werden kann. Dieses System wird im weiteren kurz RUPI genannt.

2. Aufbau des Systems

Werden mehrere gleiche oder ihren Funktionen nach ungleiche Module durch einen Kleinrechner gesteuert bzw. kontrolliert, so entstehen Anlagen, die entsprechend ihren Einsatzphilosophien als Einzweckelektroniken oder gekoppelt mit einem Rechner der höheren Hirarchiestufe, als autarke Untersysteme angesprochen werden können.

Über Einsatzphilosophie und Organisationsprobleme soll in diesem Aufsatz nicht gesprochen werden.

Die neutrale Auslegung der CAMAC-Empfehlungen EUR 4100 bot sich als Grundlage für die Entwicklung des Systems RUPI an. Dies betrifft:

- die Anordnung und Timing des Dataways
- die Empfehlungen über mechanische Größen
- und die Signal-Standards.

Diese Richtung wurde weiterhin durch die bestehenden Verbindungen mit der Kernforschungsanlage Jülich, Zentrallabor für Elektronik, wesentlich unterstützt.

Auf dieser Grundlage wurde eine Modulserie erarbeitet, welche im wesentlichen durch komplexe Funktionen gekennzeichnet ist. Diese Modulserie stellt ein geschlossenes System im Sinne der EDV-Anwender dar. In dem Aufbau wurden Gesichtspunkte der professionellen Anlagenfertigung berücksichtigt.

Die Mechanik wurde in zwei Ebenen eingesetzt. In der Zentrale als CAMAC-Crate mit Dataway und im Prozeß als Trägermechanik für die nicht Dataway-kompatiblen Anpassungselektroniken.

Bild 1 zeigt den grundsätzlichen Aufbau einer solchen Anlage. Es sind drei Ebenen zu unterscheiden. Die Zentrale, bestehend aus einem Klein- bzw. Prozeßrechner, aus der Zentrale des Systems RUPI, das im wesentlichen ein CAMAC-Crate mit den entsprechenden Modulen beinhaltet. Diese Module sind in der Lage, ohne zusätzliche Modems auf eine Entfernung von max. ca. 10 km angeschlossen zu werden. Diese Maßnahme gewährleistet der Zentrale eine gewisse Ortsunabhängigkeit, was bei großräumigen Betrieben ein wesentlicher Gesichtspunkt ist. Direkt am Prozeß werden sogenannte externe Elektroniken eingesetzt, die mit der Zentrale kompatibel sind und eine direkte Anpassung an den Prozeß darstellen.

3. Die Instrumentierung

Die Instrumentierung entspricht der Aufgabenstellung. Folgende Module finden Einsatz:

3.1 Universelle Kommunikations-Zweiweginterface
3.2 FS Ein- bzw. Ausgabemodul,
3.3 Multiplexer für den Anschluß von externen Zählern, Digitaleingänge (Matrizes, statische- und dynamische Eingänge), Digitalausgänge, Sonderelektroniken.
3.4 Standardsystem für die Gestaltung von Betriebstastaturen,
3.5 eine Reihe externer Anpassungselektroniken für unterschiedliche Ein- bzw. Ausgabegeräte, wie z. B. Lochstreifenleser/Stanzer - Lochkartenleser/Stanzer usw.
3.6 externe Speicher für Anzeigen bzw. Sollwertvorgaben,
3.7 Universalmodul für die Gestaltung von Sonderelektroniken.

In der Zentrale sind einige Exoten vorgesehen, wie z. B. DMA-Modul für einen direkten Kernspeicherzugriff, Puffer-Modul, für Anpassungs- und Umformatierungsfragen in Zusammenarbeit mit dem Siemens-Prozeßelement P4KS.

Die Komponenten des Systems RUPI finden ihren Einsatz nicht nur in der horizontalen Daten-Ebene, sondern auch in der vertikalen Übertragungsebene. Bild 2 zeigt ein redundantes Mehrrechnersystem.

4. Einsatzmöglichkeiten

Es sind im wesentlichen drei Einsatzgebiete zu unterscheiden.

- kleinrechnergesteuerte bzw. kontrollierte Einzweckelektronik
- vorhin genannte Systeme gekoppelt mit einem Rechner der höheren Hirarchiestufe als Prozeßperipherie mit selbständigem Entscheidungsvermögen
- und als Koppelelektronik. Diese Ein-

satzmöglichkeit wird durch die universelle Auslegung der einzelnen Module gewährt.

Da die Vielseitigkeit eines solchen Systems anhand eines konkreten Anwendungsbeispiels schlecht bzw. nur ungenügend dargestellt werden kann, wird hier eine Anlagenkonfiguration vorgestellt, die im Bereich der Materialflußverfolgung typisch ist.

Im Bild 3 sind Komponenten aller drei Einsatzgebiete aufgezeichnet. Die einzelnen Knotenpunkte in der horizontalen Ebene bilden einzelne autarke Untersysteme. Würden diese Systeme keinen Anschluß an einen Rechner der höheren Hirarchiestufe besitzen, so könnten sie als Einzweckelektroniken betrachtet werden.

Die Ankopplung der einzelnen Untersysteme erfolgt ebenfalls über Systemkomponenten des Systems RUPI. Aus dieser groben jedoch typischen Aufgabenstellung im Bereich der Prozeßsteuerung bzw. Prozeßleitsysteme ist das Einsatzgebiet des Systems RUPI ersichtlich.

Es soll zum Ausdruck gebracht werden, daß beim industriellen Einsatz nicht nur die Grundphilosophie, sondern deren Einsatz und Anwendung dominiert.

Das System stellt also eine andere Interpretation der CAMAC-Empfehlungen dar.

5. Preis / Leistungs - Betrachtung

Ein Vergleich kann wegen der Vielzahl der Anwendungsmöglichkeiten generell nicht gezogen werden.

5.1 Einzweckelektroniken

Die notwendige Preisaufwendung für die Grundausrüstung bei Einzweckelektroniken rechtfertigt sich nur bei komplexeren Aufgaben. Hierbei ist zu beachten, daß bei Anhäufung von preiswerten Einzelschaltungen nach einer sinnvollen Gruppierung der einzelnen Funktionen der Aufwand in vielen Anwendungsbereichen gerechtfertigt ist. Als typisches Beispiel sei hierzu die Meß- und Rechnertechnik genannt.

Erfahrungsgemäß sind Anlagen unter DM 80.000,- nicht optimal zu realisieren.

Durch die Anwendung solcher Anlagen wird ein Maximum an Flexibilität und Anpassungsfähigkeit sowohl in der Planungs- als auch in der Betriebsphase erreicht.

5.2 Prozeßperipherie

In der Anwendung als Prozeßperipherie im Zusammenhang mit Großrechnersystemen hat die Erfahrung gezeigt, daß sich durchaus konkurrenzfähige Konfigurationen zu herkömmlichen Rechnerperipherien nach hausinternen Normen der Rechnerhersteller gestalten lassen.

Die Leistungsgrenze solcher Peripherie-Einrichtungen liegt unumstritten günstiger als bei Standardsystemen der einzelnen Großcomputerhersteller. Es ergeben sich neue Organisationsmöglichkeiten, da es sich um selbständige Einrichtungen handelt, die in der Lage sind, die Rechnerperipherie sowohl logisch als auch funktionell autark zu bedienen.

5.3 Rechnerkopplung

Über Preise im Bereich der Rechnerkopplung kann nichts gesagt werden.

Die Vorteile der auf dieser Basis realisierten Rechnerkopplungen liegen im wesentlichen in der Neutralität der verwendeten Komponenten. Hierdurch können Probleme optimal gelöst werden, die durch die unterschiedlichen Rechnerarchitekturen entstehen.

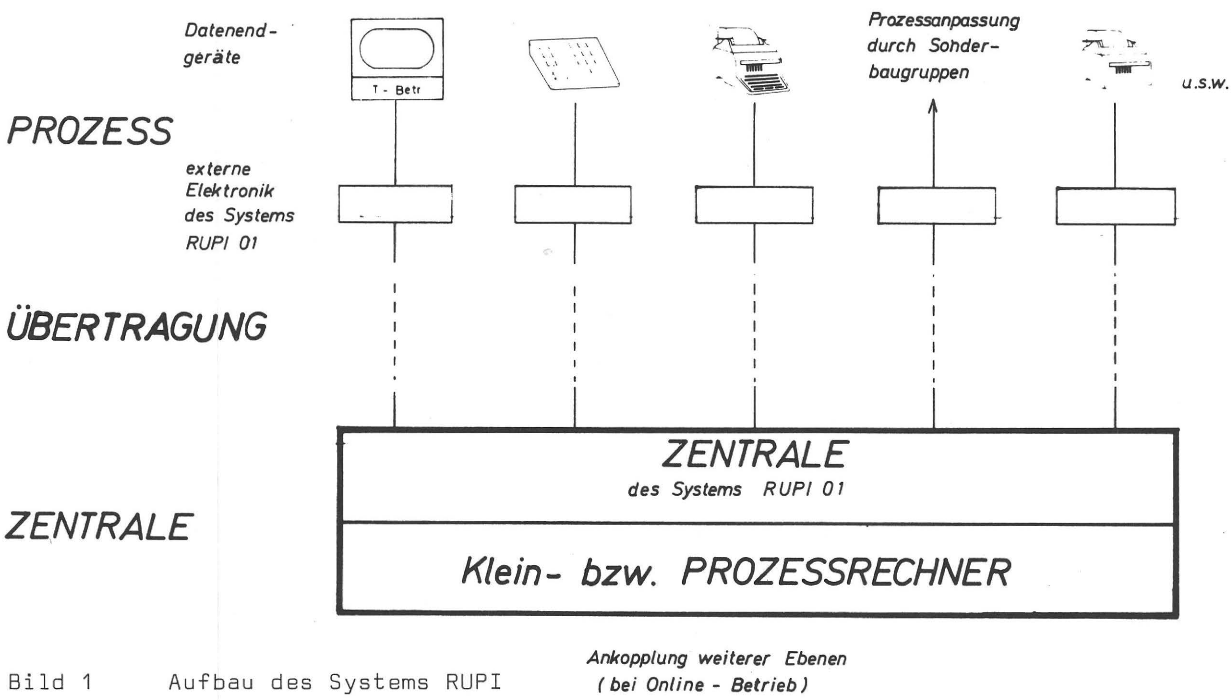


Bild 1 Aufbau des Systems RUPI

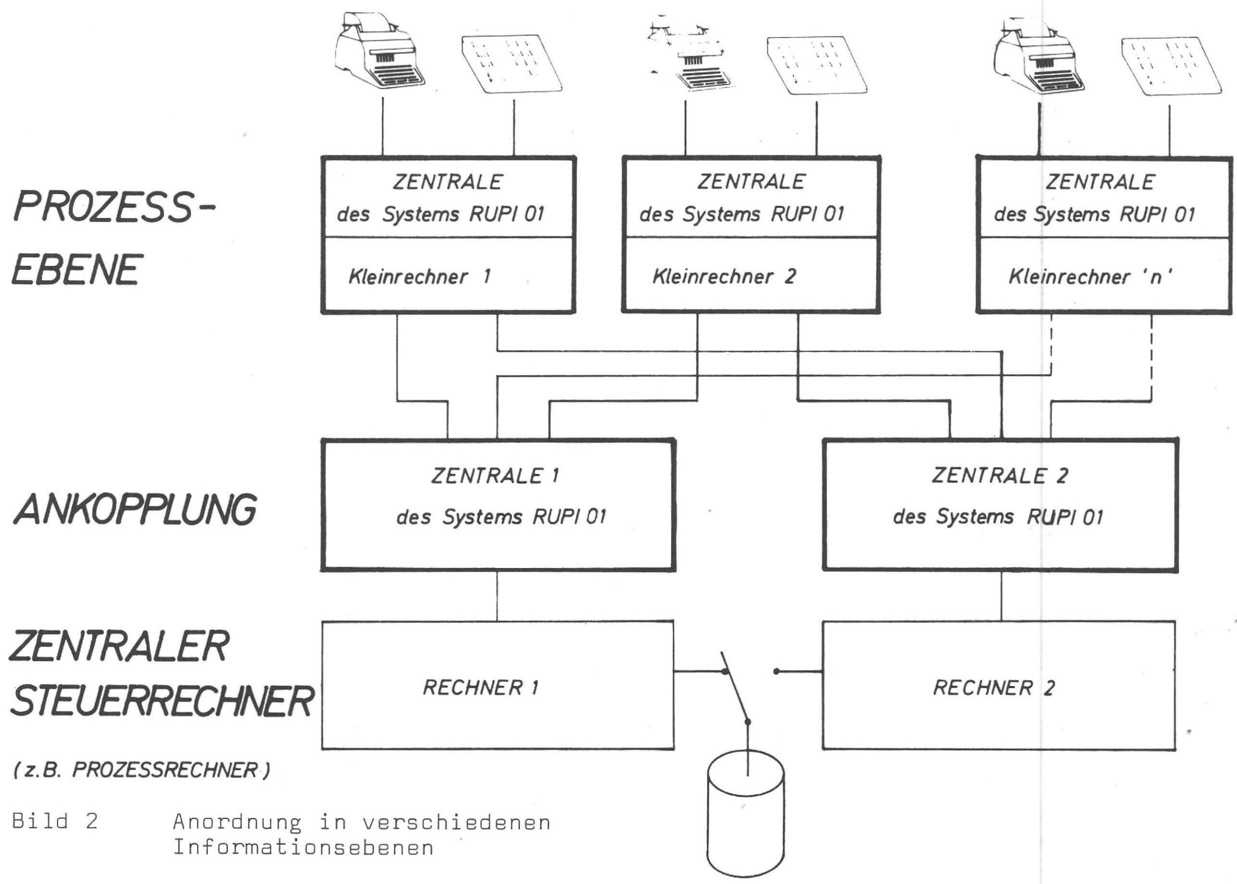


Bild 2 Anordnung in verschiedenen Informationsebenen

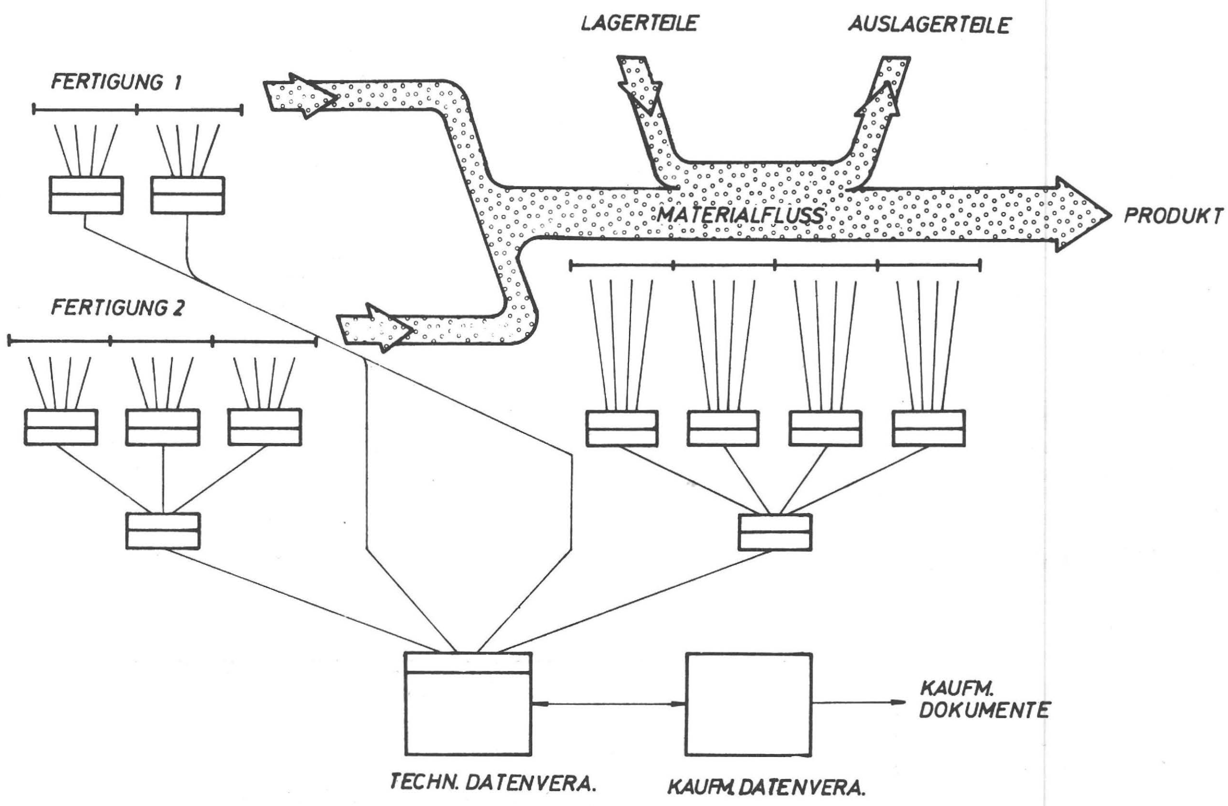


Bild 3 Blockschema für die Anwendung von Rechnersystemen im Bereich der Materialflußverfolgung

SMALL SYSTEMS FOR MEASUREMENT AND CONTROL

D. Reimer
Dornier GmbH, Friedrichshafen, Germany

ABSTRACT

The requirements to be met by an industrially applicable system for measurement and control tasks are delineated. Advantages and disadvantages as a consequence of realizing such a system according to CAMAC standards show that the CAMAC instrumentation represents a flexible solution at reasonable cost. Small autonomous systems require a programmable control unit of easy operation. An example of a monitoring system for temperatures again shows the advantages of a CAMAC instrumentation.

ZUSAMMENFASSUNG

Die Forderungen an ein in der Industrie einsetzbares System für Mess- und Steueraufgaben werden skizziert. Die Vorteile und Nachteile, welche bei der Realisierung eines derartigen Systems nach CAMAC-Standards entstehen, zeigen, dass eine CAMAC-Instrumentierung eine flexible und kostengünstige Lösung darstellt. Kleine autonome Systeme benötigen eine programmierbare und leicht zu bedienende Steuereinheit. Ein Beispiel einer Überwachungsanlage für Temperaturen zeigt noch einmal die Vorzüge einer CAMAC-Instrumentierung.

RESUME

Bref exposé des conditions auxquelles doit répondre un système de mesure et de commande utilisable dans l'industrie. L'analyse des avantages et des inconvénients liés à la réalisation de ce système d'après les normes CAMAC prouve que l'instrumentation CAMAC apporte une solution plus souple et plus économqiue. Les petits systèmes autonomes nécessitent une unité de commande programmable et facile à manipuler. Un exemple d'installation de contrôle de températures fournit une nouvelle preuve des qualités de l'instrumentation CAMAC.

1. Requirements to Be Met by an Industrially Applicable System for Measurement and Control Tasks

Automation and monitoring systems in the industry have in many cases relatively simple tasks compared to processor systems. Control of machines and production must fulfil the following requirements:

- Connection facility for usual sensors and final control elements: it must be possible to connect analogue and digital sensors, transducers, and final control elements to the system in any combination.

- Possibility to adapt the system to any monitor and control tasks; this demand is normally fulfilled by hardware programming.

- Operation; simple operation is essential, e.g. by means of operating keys.

- Facility to attach recorders, printers, loggers, and storage units.

- Attachment facility for superior central units, in order to render possible central accounting, statistics or reference input.

2. Advantages and Disadvantages of CAMAC

These requirements can, as a total, be fulfilled by a modular system only. They can surely be met by a CAMAC automation system.

Advantages:

- Standardized hardware.

- Availability of broad spectrum of modules of a large number of producers.

- Attachment facility for almost any sensors and final control elements as well as many loggers, storage and recording devices.

- Various possibilities for building up hierarchical systems (branch highway, single crate controller for computers, communication interface unit, serial branch highway etc.).

- The requirement for easy operation can be met in many ways by programming the employed computer, which for these simple tasks will be a small computer.

- Due to the possibility to exchange the modules, maintenance causes no difficulty at all.

Disadvantages:

- CAMAC is designed as a laboratory automation system and can therefore be employed under normal room conditions only; furthermore some VDE instructions can only be fulfilled by slight modifications.

- As a process periphery system CAMAC is designed for the solution of complex problems. Universality and flexibility automatically lead to high costs.

  The second item, however, is a prejudice and is refuted in the following such that a wide field of applications in the field of non-nuclear physics and even of applications outside the laboratory are possible.

3. CAMAC Data Processor Systems

The high basic costs of a CAMAC system are only necessary if the system is built up according to the "official ESONE solution" with central unit (computer) - system controller - branch highway - crate controller - dataway - module (function unit in the periphery). Elements without logical importance may be left out if the central unit directly effects the modules via the dataway.

Fig. 1 - CAMAC System Structure

An autonomous controller uses, as a CAMAC module, the control station of a crate. It comprises a program (hardware or software) which, e.g., controls the cyclic interrogation of many temperature measuring points via multiplexers and analogue to digital converters and performs a comparison to limits.

In order to optimally meet the demand for adaption to any surveillance and control tasks the autonomous controller must be programmable. This means that it represents a very small computer with typical functional units, as arithmetic element, register block, program storage, and timing circuitry, and with standardized interface to the CAMAC dataway.

Fig. 2 - CAMAC Data Processor Block Diagram

The instruction set of the data processor may be kept relatively simple; it is, however, possible to realize without difficulties extensive logical and arithmetical functions with integrated circuits being today available. Transfer instructions, jump instructions, and some constant instructions are necessary.

Table I - Data Processor Instructions

To program a data processor it is obviously necessary to know the assembler and its handling. As such an assembler can only run on a large computer this would be contradictory to easy programming and operation. In order to overcome this difficulty the method of the macro instructions is used.

This means that certain instruction sequences (e.g. instructions for switching the multiplexer over to a selected channel for reading the value for comparing it to its limit value in the temperature monitoring system mentioned above) are programmed and numbered. The sequences are filed in the read-only storage of the data processor together with an interpretative program calling and performing the macros in the desired sequence. The calls, i.e. a list of numbers, can be introduced to the data processor via a plugboard, via switches (BCD input) or key boards and can therefore be modified easily. That way all basic requirements are met, and the high costs for a computer and its programming are eliminiated.

4. Example

The monitoring system for temperatures or other analogue measuring values, already cited as an example, can be realized with one 19 inch plug-in unit of 7 height units.

Fig. 3 - CAMAC Data Acquisition System

256 measuring points may be connected via relay multiplexers, the measured values may be preamplified and digitized subsequently. They can be stored (magnetic tape cassette or paper tape); or after a limit survey and a recognized exceeding an alarm signal can be given.

Further output units are feasable including the transfer of measured values or calculated results via telephone lines (modems according to V 24) or special lines.

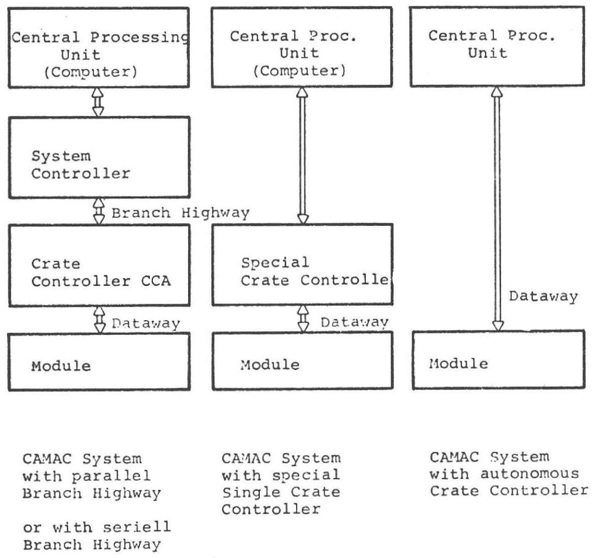


Fig. 1: CAMAC System Structure

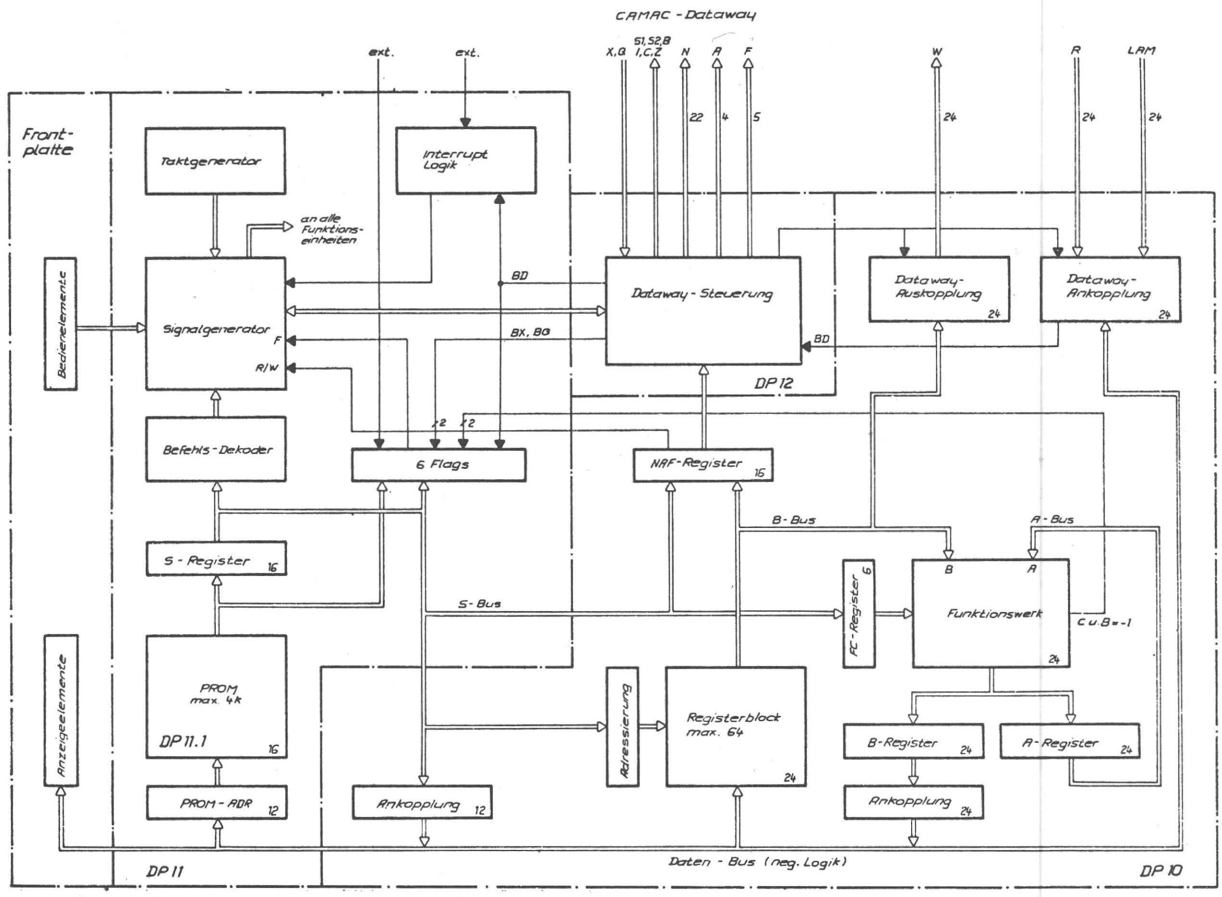


Fig. 2: CAMAC Data Processor Block Diagram

Table I: Data Processor Instructions

Code instruction word:

O P	M D	B_{12} ... B_1

Transfer instructions OP = 00

TRF	MD = 00	Transfer to reg. 1 of DP	NAF in B_1 to B_{12} and MD
	01	Status	N = B_5 ... B_1
	10	Transfer from reg. 1 of DP	A = B_9 ... B_6
	11	Command	F = MD, B_{12} ... B_{10}

Jump instructions OP = 01

BNC	MD = 00	Mandatory jump	Absolute jump address in B_{12} to B_1
BNF 1	10	Jump if no carry	
BNF 2	01	Jump if no BD	
BNF 3	11	Jump if result +-1	

Register instructions OP = 10

REX	MD = 00	Exchange reg. 1 and reg. 2	Reg. 1 = B_{12} ... B_7
EFC	01	Combine reg. 1 with reg. 2 *)	Reg. 2 = B_6 ... B_1
RCP	10	Transfer reg. 1 to reg. 2	
RAI	11	Add up reg. 1 and reg. 2 and increment	

*) 48 logical and arithmetical functions are feasable. The function to be performed is selected by SFC and carried out by EFC.

Set instructions

SAR	MD = 00	Set bits 1 to 12 of reg. 1	Data bits in $B_{12} \ldots B_1$
SAL	= 01	Set bits 13 to 24 of reg. 1	
SKS	= 10	Skip, if flag is not set	($B_{12} \ldots B_7$ = 000 000)
SFC		Set kind of combination	($B_{12} \ldots B_7$ = 000 001)
SFL		Set flags	($B_{12} \ldots B_7$ = 000 010)
CFL		Cancel flags	($B_{12} \ldots B_7$ = 000 011)
EXE	= 11	CAMAC transfer with data in reg. 2 and NAF in reg. 1	(Reg. 1 = $B_{12} \ldots B_7$, reg. 2 = $B_6 \ldots B_1$)

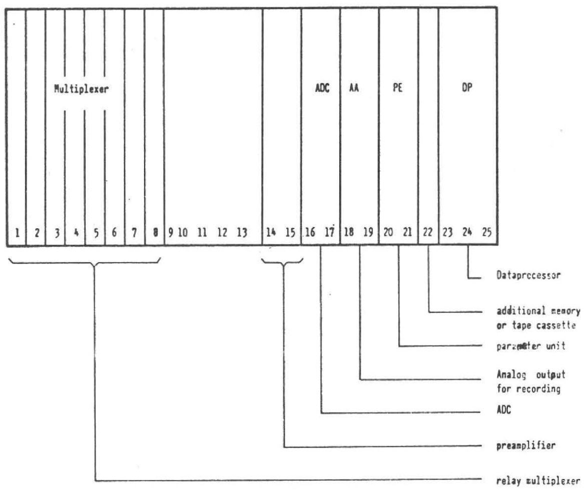


Fig. 3: CAMAC Data Acquisition System

DISCUSSION

Q - D. Abbott

What is the instruction cycle time of your crate controller?

A - D. Reimer

An internal instruction (ADD, JUMP) needs 0,5 to 1,0 µsecs, for external instructions (TRF) additional 0,5µsecs are necessary and also the Dataway cycle time for CAMAC instructions (1,0 to 1,4 µsecs).

Q - D. Abbott

How long (how much time, how many instructions) does it take to set up a CAMAC instruction?

A - D. Reimer

It takes one instruction, that is 1,0 to 1,4 µsecs.

Q - D. Abbott

How much memory (RAM and/or PROM) is incorporated?

A - D. Reimer

Up to 4K words of 16 bits (RAM or/and PROM).

COMPUTER CAMAC BASED CONTROL SYSTEM FOR A 1500 L/S WATERPUMP STATION

A. Schindler
SEN Electronique, Genève, Switzerland

ABSTRACT

This survey and control system is an important landmark in the progressive automation of a particular public service. The use of standard production CAMAC modules greatly simplified the design and construction of the system, and provided a modular conception which was essential in view of the established expansion program.

ZUSAMMENFASSUNG

Das beschriebene Überwachungs- und Steuersystem ist als wichtiger Markstein bei der fortschreitenden Automatisierung eines bestimmten öffentlichen Versorgungsbetriebes zu werten. Durch die Verwendung von standardisierten CAMAC Einheiten aus lieferbaren Typenreihen sind die Planung und die Herstellung des Systems beträchtlich erleichtert worden. Durch CAMAC war auch ein modularer Aufbau des Systems möglich, welcher für die vorgesehenen Erweiterungspläne wesentliche Bedeutung hat.

RESUME

Ce système de surveillance et de contrôle fait époque dans l'automatisation progressive d'un service public déterminé. L'utilisation de modules CAMAC dont la production est normalisée a permis de simplifier dans une large mesure le projet et la construction du système; elle a fourni en outre une conception modulaire indispensable au programme d'expansion arrêté.

1.INTRODUCTION

Les Services Industriels de Genève ont été mandatés par la FIPOI pour construire une station de pompage et de traitement d'eau pour l'alimentation des installations de refroidissement et la fourniture en eau potable de CERN II.

Nous-mêmes, SEN ELECTRONIQUE GENEVE, avons fourni l'instrumentation et le calculateur industriel ainsi qu'une assistance pour les problèmes d'acquisition de données et de contrôle.

Pour bien saisir la portée de cette application, il faut se placer dans une perspective historique :

Jusqu'en	CONTROLE
1950	MANUEL
1960	" + RELAIS
1970	" + TRANSISTORS (I.C)
Période transitoire 1975	" + LOGIQUE CABLEE + CALCULATEUR INDUSTRIEL
....	MANUEL + CALCULATEUR INDUST.+ TELECOMMANDE

2. LE POURQUOI DE CETTE EVOLUTION

- Normalisation entre les stations;
- Simplification de l'entretien;
  " de la maintenance;
- Adaptabilité de l'ensemble : à un nouveau problème similaire (au nombre d'entrées/sorties près) ou à une évolution des paramètres contrôlés;
- L'avantage d'un calculateur industriel programmé ressort clairement si l'on considère que le programme peut être testé très loin hors du système en exploitation;
- Une fiabilité plus grande puisque pour un même ordre de grandeur du nombre de composants on réduit le nombre de connections.

3. LES DONNEES DU PROBLEME

Les deux points fondamentaux sont par ordre d'importance :

1) La qualité de l'eau
2) La production

En effet, une station de pompage doit satisfaire généralement ces deux exigences. La qualité de l'eau est définie par un ensemble de critères chimiques et physiques :

- La turbidité : c'est-à-dire un coefficient d'absorbtion de la lumière par le liquide de 4 à 18° C. (max.), en moyenne 8 à 9° C.
- La température
- Le taux de matière en suspension
- Le taux d'oxygène dissout
- L'oxydabilité : la consommation de $KMNO_4$
- La concentration de différents sels en suspension
- La concentration en germes aérobies.

Hormis pour le dernier point, ces critères sont mesurés en ligne et continûment à chaque étape du traitement. Quant au débit, celui-ci doit rester compris entre une valeur maximale définie au moment de la conception de l'usine et une valeur minimale telle que le courant empêche la fixation d'organismes dans les canalisations; en particulier il faut tenir compte de phénomènes saisonniers tels que l'éclosion des larves d'un coquillage qui,une fois adultes et fixées dans une conduite, peuvent entraîner sa complète obturation.

4. QUELQUES VALEURS NUMERIQUES

Sans entrer dans les détails et juste à titre de comparaison, la station qui alimente principalement la ville de Genève (la station du Prieuré) débite à un régime de l'ordre de 2500 l/s, alors que celle-ci est prévue pour un régime de l'ordre de 1500 l/s. Pour cerner le problème, il a fallu :

- 50 entrées analogiques
- 7000 " digitales
- 60 sorties

5. LA PERIODE TRANSITOIRE

Celle-ci correspond essentiellement aux besoins suivants :

- assistance aux opérateurs humains;
- enregistrement complet de la marche de l'installation;
- connaissance à chaque instant et immédiatement de l'état de l'usine;
- en cas de panique sur le réseau, l'historique de l'événement peut être connu avec précision. La cadence de scrutation est telle que les événements arrivés en cascade peuvent être remis dans leur succession logique.

Le point fondamental, car on ne travaille pas encore en boucle fermée, reste celui de l'acquisition de données. Il permettra, grâce à la création d'une banque de données, une analyse à des fins de :

- maintenance préventive
- recherche d'optimum
- création de modèles pour de la stimulation.

Cette étape est indispensable pour pouvoir envisager avec toutes les garanties requises le passage au contrôle entièrement automatique.

La photo (Fig. 1) montre le système avant d'être livré aux Services Industriels.

6. LE MATERIEL

- Le schéma bloc - fig. 2

  Deux circuits sont à considérer :

  - un circuit normal
  - un circuit auxiliaire

Le circuit normal :

- pompage d'eau brute
- stérilisation
- floculation
- filtration mécanique
- réglage du pH
- ozonisation
- destruction d'ozone résiduel
- chloration

Le circuit auxiliaire :

- mesure de la perte de charge (encrassement)
- normalisation de l'encrassement au débit maximum.

Ces calculs sont effectués par le calculateur.

- commande à l'opérateur de déclencher le processus de lavage des filtres en fonction des résultats obtenus et du test d'un certain nombre d'états logiques de la station
- récupération et traitement des eaux usées.

Tous les capteurs ont été normalisés de 0 à 20 mA. A la sortie de la station une dernière mesure de la qualité est effectuée sur les 5 points suivants :

- chlore résiduel
- turbidité
- pression
- température
- débit

7. L'INSTRUMENTATION CAMAC

L'usage industriel de cette instrumentation est un exemple typique de retombées scientifiques dans le monde de tous les jours.

En effet, les physiciens de la physique des hautes énergies se sont heurtés vers la fin des années 60 à des problèmes d'acquisition de données en particulier et plus généralement à des problèmes de contrôle de leurs expériences. Ceci les a amenés à définir des techniques et un type d'instrumentation dont je vais très rapidement vous exposer les normes. Mais auparavant je vous donnerai quelques raisons qui ont présidé au choix de cet appareillage dans cette application :

- son prix
- **sa grande souplesse d'utilisation**
- ses performances élevées pour une fiabilité comparable
- sa modularité
- son indépendance face à un constructeur
- ses références : milieu hautement perturbé CERN

  applications industrielles antérieures

8. LE STANDARD

Pour une description détaillée des normes CAMAC, il faut se référer à la publication officielle EUR 4100.

9. LES MODULES UTILISES

La figure 3 montre un des chassis CAMAC PC 2006C. La réalisation de cette application nécessite principalement l'emploi des modules suivants :

- deux chassis avec leurs alimentations	PC 2006/C
- terminaison de BUS	BT 2022
- un multiplexeur destiné à sélectionner par l'intermédiaire d'un ordre CAMAC l'une de ses 12 voies d'entrée	MX 2025
- double voltmètre digital destiné à travailler dans un environnement hostile	2DVM 2013
- double registre de sortie 16-BIT	2OR 2008
- double convertisseur digital à Analogue	2DAC 2011
- surveillant de 64 lignes	64LS 2052

Je m'arrêterai quelques instants sur le module 64 LS 2052, car il présente plusieurs aspects intéressants. Ce module permet la surveillance de 64 lignes portant des niveaux TTL. Par surveillance, il faut comprendre le choix parmi trois modes :

- 1 entrée passe à l'état 0
- 1 entrée passe à l'état 1
- 1 entrée change d'état par rapport à l'état précédent.

Lorsque une alarme est détectée :

- 1 ordre de lecture identifie l'adresse de la ligne concernée ainsi que son état
- 1 commande de masquage permet de cacher l'alarme une fois qu'elle a été reconnue, afin d'éviter la saturation
- l'état du registre de masque peut être, lui-même, lu.

Par ailleurs, la possibilité existe de tester "on line" le fonctionnement de tiroir proprement dit en raccordant une ligne de test présente sur chaque prise à l'une des entrées, ce qui produit nécessairement une alarme à l'adresse sélectionnée.

Ces modules travaillent sous la directive du contrôleur de chassis dont je vais maintenant vous parler.

10. LE CONTROLEUR - L'INTERFACE

Le contrôleur de baie CC 2023 a la double fonction dans notre philosophie:

- il génère les opérations sur le dataway (purement CAMAC)
- il lie l'ordinateur aux modules (l'interface)

Il est dédicacé aux ordinateurs Nova et peut être considéré comme périphérique.

La liaison calculateur-contrôleur est faite de la manière suivante: nous disposons essentiellement des registres suivants :

registre de commande génère A,N,F

" " données R,W,L

" d'adresse en mode d'accès direct à la mémoire.

On peut travailler en transfert programmé ou en accès direct à la mémoire. Dans ce dernier cas :

-accès "à gachette" (triggered access) où le transfert est commandé par un signal externe

-accès autonome à un bloc d'information : une suite de mots serait transférée avec incrémentation automatique des sous-adresses et des adresses.

11. LE CALCULATEUR INDUSTRIEL

La NOVA DGC a un ensemble d'instructions relativement simples (29 instructions de base) mais qui est démultiplié par les possibilités de la micro-programmation sur les instructions arithmétiques et logiques et sur les instructions d'entrées/sorties (soit 1071 instructions). Chaque instruction peut adresser 1024 mots-mémoire directement et indirectement toute la mémoire disponible.
16 niveaux d'interruption software assignables dynamiquement sont disponibles.
Un MTBF de 6000 heures est typique en tenant compte de la durée de vie des lampes de la console, bien que cette panne ne soit pas fatale.
Le NOVA modèle 1200 était choisi pour cette application - la figure 4 montre l'installation.

12. LES PERIPHERIQUES

Hormis le CAMAC, on dispose essentiellement de :

- 1 disque à tête fixe : temps d'accès moyen : 84 mus
- 1 lecteur de bande papier
- 1 teletype

13. LE PROGRAMME

<u>Le système en temps réel</u>

<u>Déf.</u> Un système en temps réel est un ensemble Hardware/Software capable de contrôler (surveiller/agir sur) un ou plusieurs processus, c'est-à-dire :

- d'acquérir des informations sur ces processus
- de traiter ces informations en un temps tel que l'obtention des résultats permette de modifier le déroulement du processus.

Ses responsabilités :

- inventorier, classer et donner le contrôle aux tâches composant le programme
- traiter les interruptions
- gérer les entrées/sorties

Ses moyens :

- distribue le temps du CPU selon certains critères accessibles à l'utilisateur
- donne la possibilité d'avoir plusieurs tâches en parallèle
- assure la communication et la synchronisation entre les tâches
- offre au programme d'exploitation la possibilité de communiquer avec lui
- supporte l'adjonction de périphériques non standards
- est réentrant.

Le programme

Les tâches

Déf. : une portion logiquement complète du programme d'application pouvant être exécutée :

1) indépendamment, et
2) simultanément avec les autres tâches.

C'est l'unité de base supportée par le système d'exploitation.

Les états d'une tâche

Puisqu'une fois initialisée une tâche peut être exécutée à n'importe quel instant, le programme d'exploitation retient sur chaque tâche (hormis celle en cours d'exécution) un bloc de contrôle.

4 états sont possibles :

- Exécutant : a le contrôle du CPU
- Pendant : prête à être exécutée; le sera lorsque toutes les tâches de priorités plus grandes auront été servies
- Suspendu : attend l'occurrence ou la fin d'une opération "temps réel"
- Dormant : n'a jamais été initialisée ou est terminée.

Les priorités :

A chaque tâche est attribuée une priorité distribuée sur une échelle de 256_{10} **niveaux** (de Ø - la plus haute - à 377_8 - la plus basse). Le système lui-même est à la priorité Ø - la plus haute - et peut entrer en compétition avec les tâches que l'utilisateur a assignées à ce niveau.

Pour communiquer avec le système d'exploitation deux types d'appels ont été définis : ceux qui s'adressent au système proprement dit et ceux qui s'adressent aux tâches gérées par le système.

Les appels au système allouent les canaux d'entrée/sortie (de 1 à 63/10); formattent les transferts selon qu'ils sont par bloc, par ligne ou séquentiels; communiquent avec l'opérateur; gèrent l'horloge et le calendrier.

Les appels aux tâches lancent leur exécution, définissent (ou redéfinissent) leur priorité, contrôlent les changements d'état (suspension, réactivation) et assument les communications entre tâches ainsi que leur synchronisation.

Le programme d'exploitation compte environ 3000/10 mots et les "drivers" CAMAC inclus dans le système environ 2000. Le temps pendant lequel le système est inapte au service d'une interruption "Interrupt Inhibit Time" est en moyenne de 50 µs.

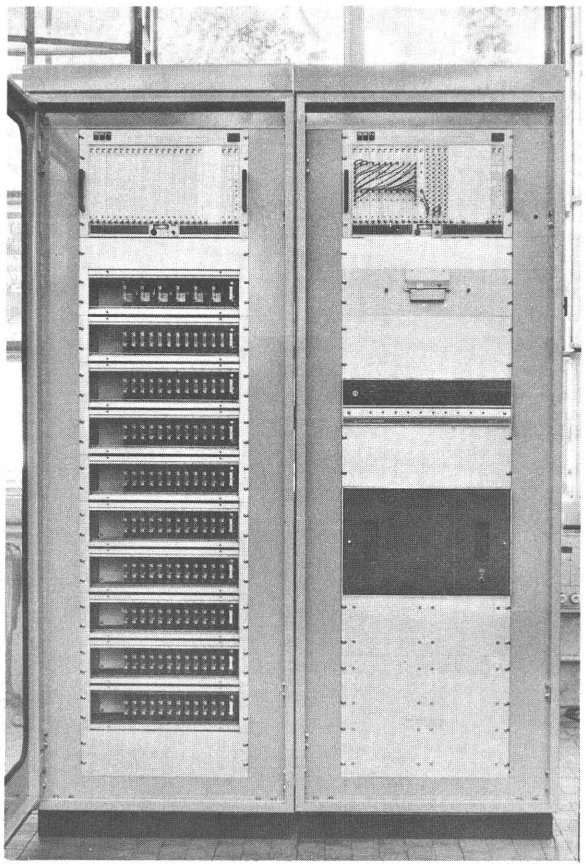


Fig. 1 Système complet

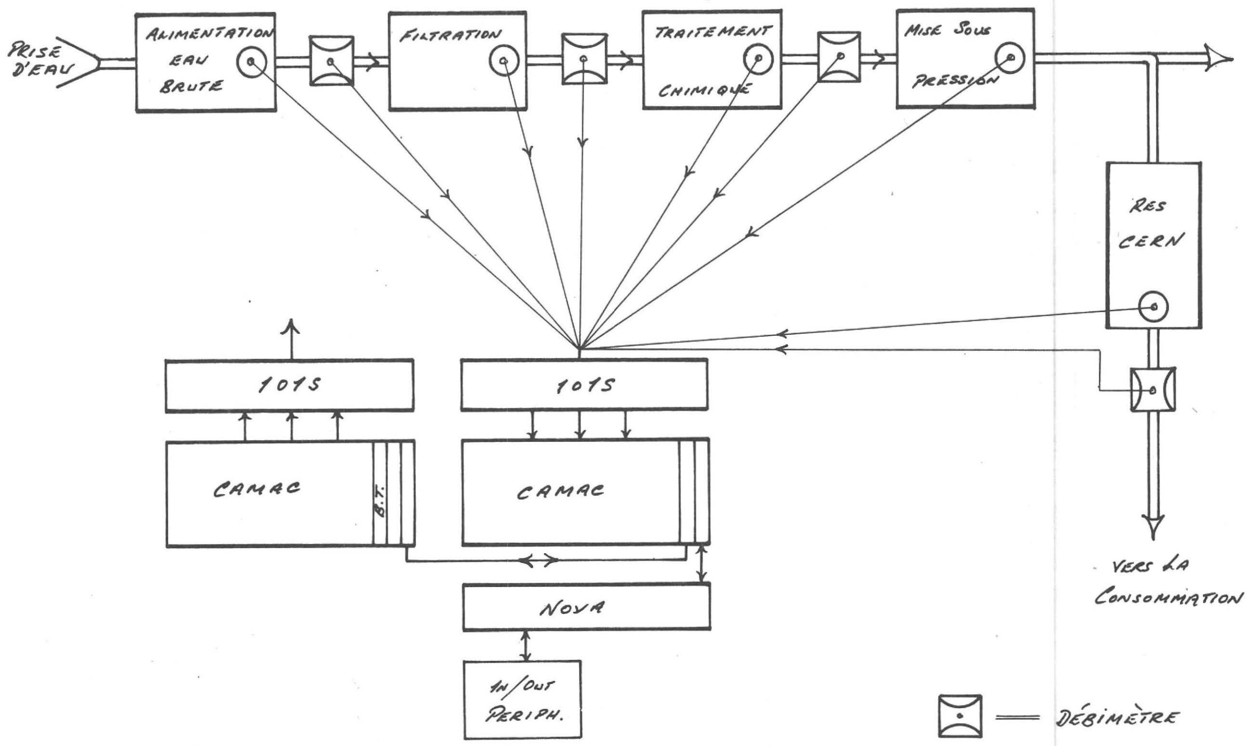


Fig. 2 Situation des points de prise des données

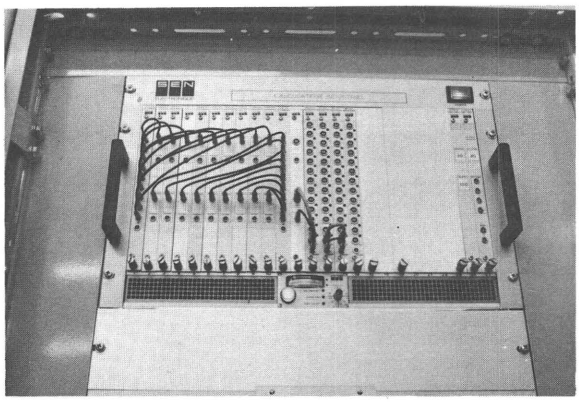


Fig. 3 Un des chassis CAMAC

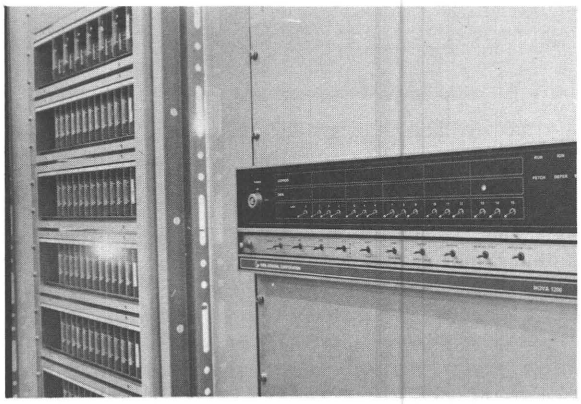


Fig. 4 L'installation du NOVA.

CONTROLE D'EPAISSEUR DANS UNE INSTALLATION INDUSTRIELLE

J. M. Servent
Schlumberger Instruments et Systèmes, Centre d'Instrumentation
Nucléaire, Bagneux, France

RESUME

Un système CAMAC-Calculateur est décrit qui assure les mesures en continu de l'épaisseur de films plastiques produits par 5 machines indépendantes. Ce système gère les signaux digitaux et analogiques d'entrée et les périphériques de sortie des informations. Il a été prévu pour une extension ultérieure à 15 machines.

ZUSAMMENFASSUNG

Ein rechnergesteuertes CAMAC-System führt kontinuierlich Dickenmessungen an Kunststoffstreifen durch, die von fünf Maschinen hergestellt werden. Das System verarbeitet die digitalen und analogen Eingabesignale und steuert die Peripheriegeräte für die Datenausgabe. Das System ist so ausgelegt, dass eine spätere Erweiterung zur Steuerung von 15 Maschinen möglich ist.

ABSTRACT

A computerized CAMAC system is described which makes continuous thickness measurements of plastic ribbons produced by 5 independent machines. This system handles the input digital and analog signals and peripherals for data output. It has been designed to be further easily extended to 15 machines.

1 - BUT

Le système CAMAC-Calculateur qui va être décrit, est destiné à la mesure en continu de l'épaisseur des films plastiques produits par 5 machines indépendantes. Il a été conçu pour une extension ultérieure à 15 machines.

Le système assure :

- Le dialogue avec l'opérateur
- La mesure des épaisseurs
- La gestion des appels machines
- L'émission de certaines alarmes
- Le traitement des données acquises
- L'édition des résultats demandés par l'utilisateur.

2 - DESCRIPTION DU SYSTEME.

Le système comprend :

2.1 - Partie calculateur.

Un calculateur PDP.11/05 de Digital Equipement avec 8 K mémoire, extension arithmétique et télétype ASR 33.

2.2 - Partie CAMAC.

- Un châssis alimentation CJAL 41.
- Un contrôleur de châssis-interface PDP.11 type JCC.11.
- Un module terminaison d'Unibus JUT.11.
- Une horloge à quartz JHQ.10.
- Un multiplexeur 16 voies JMX.10.
- Un convertisseur JCTF.10.
- Une échelle fréquencemètre JEF.10
- Un registre d'entrée JRE.10.
- Deux modules commande de relais JRD.10.
- Trois modules de commande d'enregistreurs JXY.10.

2.3 - Organes de sortie de données.

- Cinq enregistreurs X(t) Elliot.
- Une imprimante rapide Logabax.

3 - FONCTIONNEMENT.

3.1 - Acquisition des mesures.

Sur chaque machine, une jauge à absorption radio-active délivre une tension de 0 à 10 volts mesurant l'épaisseur du film. La tête de mesure est animée d'un mouvement de va-et-vient, pour effectuer les mesures d'un bord à l'autre du film, et obtenir ainsi un profil.
Pour chaque profil, 100 points de mesure sont relevés et, en fonctionnement normal, 5 profils sont réalisés pour chaque rouleau de film.

La vitesse du travelling est de 1cm/s mais la largeur des films pouvant varier d'un rouleau à l'autre, la fréquence d'échantillonnage au cours d'un travelling sera définie par le calculateur pour avoir les 100 points par profil.

3.2 - Exploitation.

Chaque machine envoie au système les signaux suivants :

- Signal analogique de mesure d'épaisseur
- Signal d'occupation de la jauge en travelling.
- Signal de changement de film.
- Signal impulsionnel d'avance de film.

Chaque machine reçoit du système les signaux suivants :

- Contact de lancement d'un travelling.
- Contact de libération de la jauge.
- Contact alarme de discordance entre enregistreur et paramètres entrés à la télétype.
- Contact de mise en service de l'enregistreur.

L'horloge à quartz envoie une interruption au calculateur à la fréquence de 100 Hz. Le calculateur incrémente le registre mémoire constituant l'horloge temps réel et vient lire le registre d'entrée JRE.10. Il détermine si un profil doit être réalisé sur une machine. (Longueur de film produite égale à 1/5 de la longueur totale) ou si l'opérateur a lancé un profil manuel (signal d'occupation à 1 avec signal du compteur à prédétermination à 0).

Dans le premier cas, l'ordinateur envoie sur la voie considérée le contact de verrouillage et, dès réception du signal d'occupation effectuera les 100 mesures nécessaires à la cadence voulue déterminée à partir de l'horloge et de la largeur de film donnée à la télétype. A la fin du cycle de mesure le calculateur libère le contact de verrouillage et effectue les calculs nécessaires à partir des données (épaisseur voulue, gamme) fournies en début de rouleau à la télétype par l'opérateur. A la fin des 5 profils spéciaux l'ordinateur sort les résultats de la machine concernée (contact fin de rouleau à 1).

Dans le deuxième cas, l'ordinateur ne pourra lancer un cycle de mesures que lorsque le profil manuel sera terminé (occupation à 0) et que la longueur voulue a été atteinte (bit profil à 1) la mise à 1 du Bit profil a été mémorisée dès son apparition pendant le Profil manuel éventuellement.

3.3 - Sortie des résultats.

Pour chaque machine, et de façon indépendante, le calculateur imprimera les résultats suivants :

- Date, numéro de machine, numéro de rouleau.
- Profils moyens des rouleaux.
- Longueurs des rouleaux.
- Profils spéciaux à la demande.
- Poids théoriques des rouleaux après indication par l'opérateur de la masse volumique.

- Rendement de surface :

$$r = \frac{\text{Poids réel}}{\text{Poids théorique}}.$$

- Enregistrement X (t) des profils moyens à raison de deux par rouleau.

3.4 - Conception du Software.

Le programme de gestion du système est organisé autour d'un moniteur effectuant une "tache principale", tous les appels sont traités en mode prioritaire, c'est-à-dire, que chaque appel provoque la séquence suivante :

- Arrêt du programme en cours
- Traitement de l'appel
- Retour au programme interrompu

Afin de faciliter les extensions ultérieures à un plus grand nombre de machines, les différentes taches gérées par le moniteur se présentent sous forme de sous routines, ces sous routines sont les suivantes :

- (LCTDLG) : dialogue avec l'opérateur
- (LCTTVL) : lancement d'un travelling
- (LCTMES) : lancement d'une mesure ponctuelle
- (LCTMOY) : calcul de la valeur moyenne
- (LCTRES) : calcul et édition des résultats.

Cette dernière sous routine utilisant le mode d'entrée sortie le plus lent sera la tache principale du moniteur.

4 - CONCLUSION :

Cet exemple d'application du CAMAC couplé à un calculateur, pour réaliser un ensemble de contrôle à usage industriel, a confirmé le grand intérêt qu'offre ce système dans l'élaboration d'équipements spécialement adaptés aux besoins de l'utilisateur. Il est bien rare, en effet, que ce dernier puisse trouver en catalogue chez un constructeur, l'équipement répondant en tous points, à ses exigences.

C'est pourquoi, la grande diversité des fonctions modulaires disponibles en CAMAC, et la facilité avec laquelle elles peuvent être associées, constituent sans aucun doute, un outil puissant pour la réalisation d'équipements spécifiques et tout particulièrement dans l'industrie.

SYSTEM ZUR INDUSTRIELLEN PROZESSTEUERUNG DES HERSTELLVORGANGES VON ASBESTZEMENT-PLATTEN MIT KLEINRECHNERN UND CAMAC IM ON-LINE BETRIEB

P. Liebl
Eternit-Werke L. Hatschek, Vöcklabruck, Austria

ZUSAMMENFASSUNG

Beschreibung eines integrierten Systems zur industriellen Prozessteuerung im On-Line Betrieb von mehreren Asbestzement-Plattenmaschinen nach Hatschek, unter Verwendung von Kleinrechnern und CAMAC. Definition der Aufgabenstellung. Vorgeschichte und Entwicklung der Anlage. Bisherige Erfahrungen.

ABSTRACT

An integrated on-line system for the industrial process control of machines to product asbestos cement plates (method Hatschek) is described. The system uses minicomputers and CAMAC. The needs, the historical evolution, the development of the system and the gained experiences are given.

RESUME

Description d'un système intégré destiné à la commande du processus industriel dans le fonctionnement on-line de plusieurs machines utilisées pour la fabrication de plaques en amiante-ciment suivant le procédé Hatschek, avec utilisation de petits calculateurs et du CAMAC. Définition de la tâche. Historique et développement de l'installation. Expériences acquises jusqu'à présent.

In Zusammenarbeit zwischen den Eternit-Werken Ludwig Hatschek in Vöcklabruck und der Oesterr. Studiengesellschaft für Atomenergie (SGAE) wurde in den vergangenen Jahren ein integriertes System für die Prozessteuerung von 7 Asbestzement-Plattenmaschinen entwickelt und im Hauptwerk Vöcklabruck verwirklicht .

Dieses System führt als Hauptaufgabe die digitale Dickenmessung und automatische Stoffregelung der Plattenmaschinen nach HATSCHEK im On-Line Betrieb aus.

Die Aufgabe der Dickenmessung und -Regelung der Asbestzement-Platten ist die Erzielung einer optimalen Gleichmässigkeit der Dicke der erzeugten Produktion. Hiefür wurde ein eigenes mit Isotopen arbeitendes Dickenmessverfahren entwickelt, nach welchem die frischen Platten unmittelbar nach ihrer Herstellung während des Durchlaufes durch einen Messrahmen mit einer Messzeit von ca. 1 sec gemessen werden. Danach erfolgt die automatische Regelung der Stoffzufuhr zur Plattenmaschine entsprechend einem ebenfalls neu entwickelten speziellen Regelungsverfahren .

Sowohl Dickenmessung wie -Regelung wurden ursprünglich von in Hardware ausgeführten elektronischen Anlagen durchgeführt. Bei Neubau von Geräten bot es sich jedoch an, die verschiedenen Zähl-und Rechenaufgaben nicht mehr von in Hardware ausgeführten Einzweckgeräten durchführen zu lassen, sondern diese nunmehr Kleinrechnern zu übertragen . Es wurden Kleinrechner der Type PDP8/E der Firma DIGITAL ausgewählt .

Die Interface-Aufgaben wurden durchwegs mit CAMAC-Einschüben gelöst, welches System sich optimal für die Behandlung der vielen analogen und digitalen Ein-und Ausgänge (siehe Beilagen Nr.1 und 2) eignet. Damit gelang es auch das Auslangen mit relativ kleiner Rechnerkapazität zu finden und insbesondere verschiedenene Vorbereitungsschritte wie Zählaufgaben bereits im CAMAC auszuführen.

In weiterer Folge ergab sich die Möglichkeit, dieses im On-Line-Betrieb die Plattenmaschinen regelnde System der Prozessteuer- und auch zu anderen Aufgaben heranzuziehen. Diese sind nach zwei Gesichtspunkten zu unterscheiden, und zwar sollten kurzfristig dem Bedienungspersonal der Anlagen soviele Daten als möglich zugängig gemacht werden um den Produktionsablauf zu optimieren, und andererseits waren alle Daten für eine spätere Verarbeitung in Form bon Betriebsprotokollen aufzubewahren .

Zusammenfassend sollen daher die von der Prozessteueranlage übernommenen Aufgaben nochmals wie folgt beschrieben werden :

- Laufende Dickenmessung und entsprechende Regelung der insgesamt 7 Plattenmaschinen im On-Line-Betrieb
- Weiters Uebernahme von zusätzlichen Steuerfunktionen, die bisher in Hardware nicht ausgeführt werden konnten
- Durchführung zusätzlicher Messaufgaben von charakteristischen Maschinendaten, die zur vollständigen Beschreibung des Produktionsablaufes erforderlich sind
- Laufende Ausgabe aller Daten auf optischen Sichtgeräten, sowie zusätzlich der statistischen Auswertungen über den Ablauf der Produktion, die damit dem Anlagenfahrer eine Optimierung seiner Eingriffe erlauben.
- Sammlung aller Fabrikationsdaten und Ausgabe zu gewissen Zeiten, wie Schichtwechsel usw. im Sinne eines kompletten Fabrikations-Protokolles
- Durchführung weiterer Berechnungen zur Charakterisierung des Ablaufes der Produktion, wie Wirkungsgrad, Materialeinsatz usw.
- Genaue Erfassung des Ausschusses nach Ursachen und Zeit
- Erfassung aller Stillstände der Maschinen nach Ursachen und Zeit, sowie statistische Auswertung

Im Sinne eines vollständigen Systems zur Information über alle laufenden Betriebsdaten (Management Information System) wird somit eine laufende Durchleuchtung aller Vorgänge an den Produktionsmaschinen durchgeführt und diese ermöglicht eine Optimierung der Produktion durch Ausschaltung von Ausschuss und Stillständen .

Weiters entsteht ein vollständiges Produktionsprotokoll, dem sämtliche Daten nachher zusammengefasst und statistisch ausgewertet entnommen werden können . Entsprechend konzentriert können Daten ebenfalls dem kommerziellen Grossrechner des Unternehmens zur Verfügung gestellt werden.

Es wurde hier somit CAMAC erfolgreich in einem Industriebetrieb eingesetzt, und die bisherigen Betriebserfahrungen zeigen, dass diese im On-Line-Betrieb stehende Anlage alle Anforderungen hinsichtlich Betriebssicherheit erfüllt.

In der Beilage sind einige Photos der Anlage enthalten .

Beilage 1

CAMAC-Signale für Rechner Dickenmessung und -Regelung
von 2 Plattenmaschinen

Signale für 1 Plattenmaschine :

Printbezeichnung im CAMAC	Eingangssignale		Ausgangssignale	
	statisch	dynamisch	stat.	dyn.
CCR-1 Lochkartenleser		13		5
CIR-1a Eingangsregister	3	21		
709-2 Zähler		4		
COP-3a Ausgangsregister			16	1
CIS Istwert-Sollwert		2		3
CZV-3 Vorwahlzähler		1		1
CTC-1a Zeitzähler Filzg.		2	20	
CTC-1b Zeitzähler Taktz.		3	20	
COP-3b Ausgangssteuerung				13
CDA-1a Digital-Analog-Conv.			3	
CDM-1 Digitaler Multiplexer	8			
Summe für 1 Maschine	11	46	59	23
Summe für 2 Maschinen	22	92	118	46
Einzelsummen	114		164	
Gesamtsumme	278			

Beilage 2

CAMAC-Signale für Rechner Datenerfassung 7 Plattenmaschinen

Printbezeichnung im CAMAC	Eingangssignale		Ausgangssignale	
	statisch	dynam.	stat.	dyn.
CCR-1 Lochkartenleser		13		5
CIR-1b Eingangsregister		24		
CIR-1c Eingangsregister		8		
CDM-1 Dig.Multiplexer	24			
CDM-1 Dig.Multiplexer	32	68		
CTC-1a Zeitzähler Filzg.		8	80	
COP-3c Ausgangssteuerung				24
CTC-1b Zeitzähler Taktz.		12	80	
Summe	56	133	160	29
Einzelsummen	189		189	
Gesamtsumme	378			

Bild 1 Fabrikationshalle mit 4 Asbestzement-Plattenmaschinen nach HATSCHEK

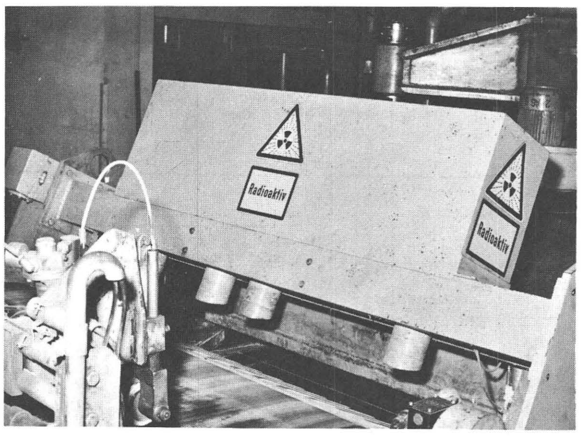


Bild 2 Durchlaufrahmen zur Dickenmessung der Platten

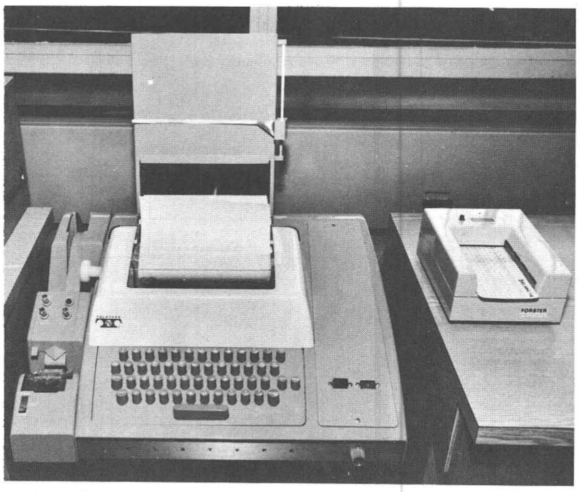


Bild 4 Dateneingabe mit Teletype und Kartenleser

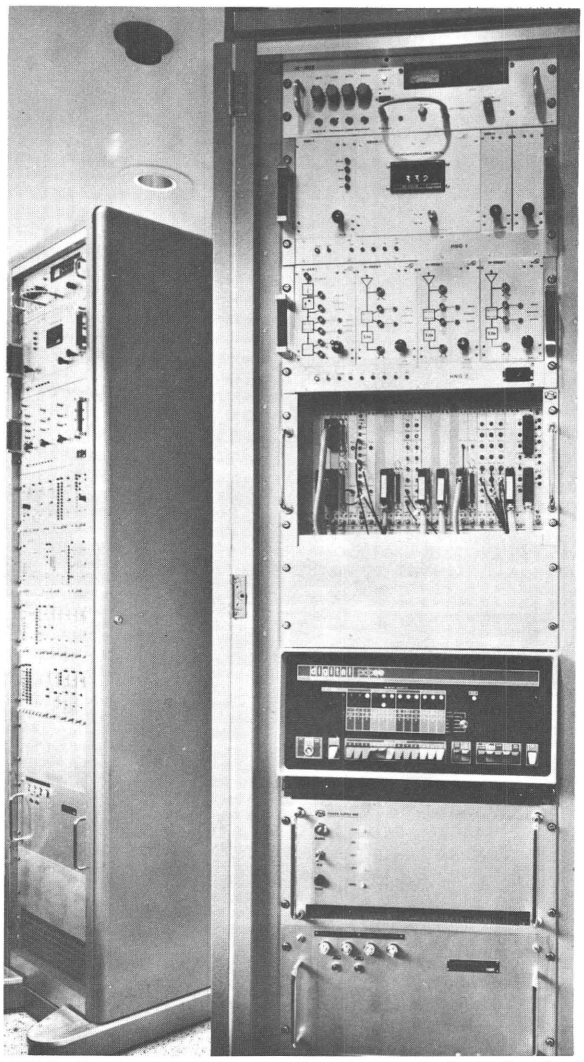


Bild 3 Blick auf einen Schrank der Anlage :
Oben der Analogteil der Dickenmessung, in der Mitte das CAMAC-Crate, unten ein Kleinrechner

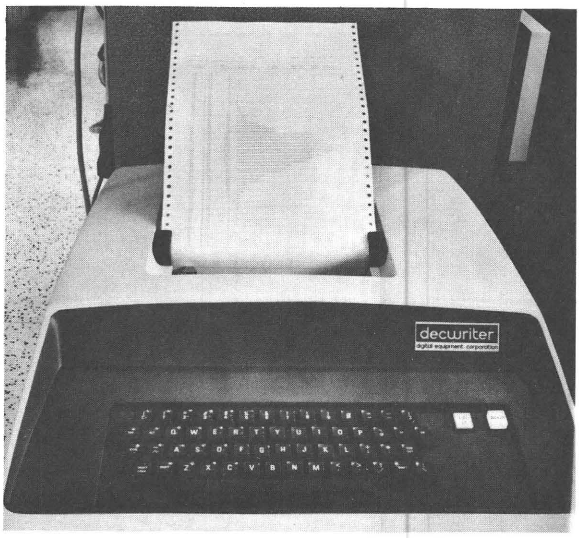


Bild 5 Protokollausgabe am Drucker

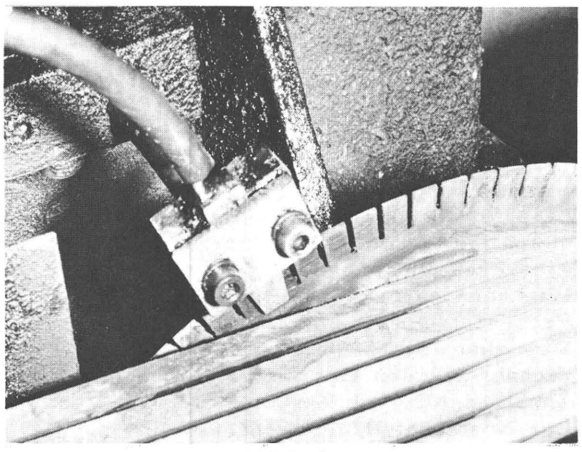


Bild 6
Digitaler Geber für Drehzahlmessung

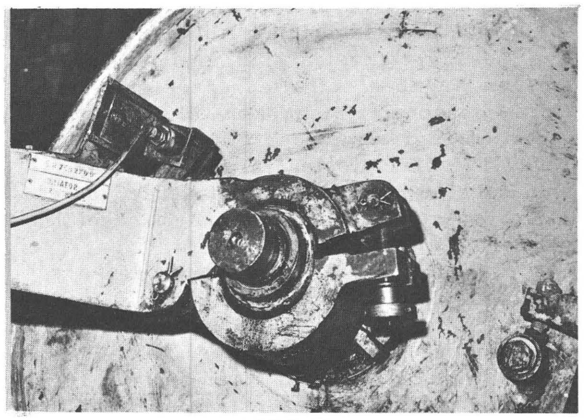


Bild 7
Digitaler Geber für Taktzeitmessung

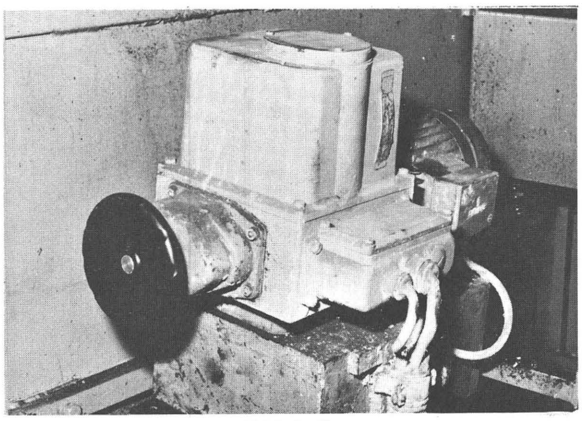


Bild 8
Stellmotor für Regelung der Stoffzufuhr

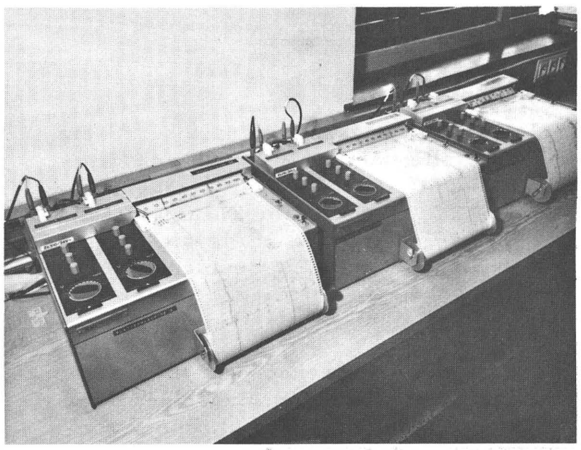


Bild 9
Analogausgabe auf Schreibern

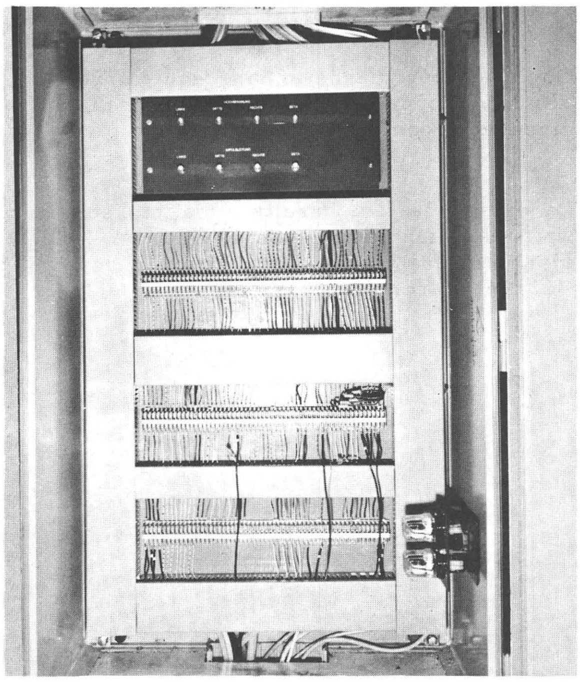


Bild 10
Klemmkasten mit Ein-und Ausgangsleitungen

DISCUSSION

Q - H. Walze

Welche Umgebungsbedingungen herrschen an der Stelle, an der der Schrank mit CAMAC-Crate und Rechner steht? Diese Frage ist interessant im Zusammenhang mit Ihrer Feststellung, dass sich CAMAC hier unter industriellen Bedingungen bewährt habe.

A - P. Liebl

Die Entfernung zwischen dem Steuerraum und den einzelnen Maschinen variiert zwischen 15m und 30m (Luftlinie). Die entsprechenden Kabellängen zu den Maschinen sind 40-100m.
Für den eigentlichen Schaltraum wurde keine Vollklimatisierung vorgesehen, d. h., keine Luftfeuchtigkeitsüberwachung. Es besteht eine Temperaturüberwachung mit Kühlaggregaten, wodurch unter allen Betriebszuständen Temperaturen zwischen 15°C und maximal 30°C auftreten können. CAMAC Geräte und Rechner sind in einem geschlossenen Schrank untergebracht zur Vermeidung von Staubeinflüssen.

C - R. Patzelt

Obviously for magnetic tape equipment you have to provide enclosures for the given environmental conditions.

Q - G. Strasser

Da dieses System bereits arbeitet, sollten weitere Daten bekannt sein wie:
1) Kosten für Hardware (Computer und CAMAC Instrumentierung)
2) Kosten für Messgeräte
3) Kosten für Systemplanung und Programmierung
4) Laufende Wartungskosten

A - P. Liebl

Eine Angabe über Kosten ist etwas problematisch, da die Maschinen zeitlich in grösseren Abständen umgerüstet würden. Man kann sagen, dass die Kosten für Prozésssteuerung und Datenerfassung pro Maschine etwa 200000 DM betragen (errechnet aus den Gesamtkosten und der Anzahl der Maschinen).
Der Kostenanteil der Prozesssteuerung ist natürlich der grösste und sollte sinnvollerweise pro 2 Maschinen angegeben werden.
Die Kosten für die Datenerfassung und Auswertung treten praktisch nur einmal für alle sieben Maschinen auf.
Die Messgerätekosten sind in den vorherigen Angaben enthalten. Eine getrennte Angabe erscheint nicht sinnvoll. Die Kosten für die Programmierung betrugen etwa 50000 DM. Die Kosten für die Systemplanung sind nicht direkt erfassbar, da diese von vorhandenen Teams durchgeführt wurde.
Die Wartung wird von der vorhandenen Elektronikabteilung durchgeführt und Kosten sind nicht so leicht anzugeben.
Es kann aber erwähnt werden, dass beträchtliche Allgemeinkosten für die Schulung des Personals entstehen.
Die Elektroniker benötigen natürlich auch als Ausrüstung Messgeräte, wofür der Aufwand nicht unerheblich ist.

DIREKT NUMERISCH GESTEUERTE WERKZEUGMASCHINEN

T. Pfeifer, K. Zenner
Institut für Werkzeugmaschinen und Betriebslehre,
RWTH Aachen, Aachen, Germany

ZUSAMMENFASSUNG

Der Einsatz von konventionellen numerischen Steuerungen in der Fertigungstechnik reicht heute nicht mehr aus, um den wachsenden Anforderungen nach höherer Produktivität, Genauigkeit und Automatisierung zu genügen. Erst durch den Einsatz von Prozessrechnern zur direkten numerischen Steuerung von Werkzeugmaschinen (DNC), im Zusammenhang mit der Entwicklung entsprechender Steuerungstypen (Rumpfsteuerungen, CNC) sind weitere Verbesserungsmöglichkeiten geschaffen worden. Unter Einbeziehung adaptiver Regelungen und neuartiger Optimierungstechniken gelingt es heute, computer-gesteuerte, selbstoptimierende, vollautomatische, integrierte Fertigungssysteme aufzubauen.

ABSTRACT

The use of conventional numerical controls in manufacturing processes is no longer sufficient to satisfy the growing requirements for increased productivity, precision and automation. It is only by the use of process control computers for the direct numerical control of machine tools (DNC) together with the development of corresponding control types (machine control units, CNC) that further improvements have been made possible. By introducing adaptive controls and new optimization techniques it is now possible to construct computer-controlled, self optimizing, fully optimizing, fully automatic, integrated manufacturing systems.

RESUME

L'application de commandes numériques classiques à la technique de la fabrication ne suffit plus actuellement à satisfaire aux demandes croissantes d'intensification de la productivité, de la précision et de l'automatisation. Seule l'introduction de calculateur de processus, en vue d'assurer la commande numérique directe des machines-outils (DNC), associée au développement de types de commande correspondants (commandes centralisées, CNC) permettra d'étendre dans une large mesure les possibilités de perfectionnement. L'incorporation de dispositifs de régulation auto-adaptables et l'utilisation de nouvelles techniques d'optimalisation permettent maintenant de construire des systèmes intégrés, commandés par ordinateur, à auto-optimalisation et entièrement automatiques.

1. Einleitung

In allen Bereichen der industriellen Produktion wird heute in höherem Maße als jemals zuvor die Forderung sowohl nach steigender Leistungsfähigkeit, höheren Genauigkeiten und Bearbeitungsgüten als auch nach sinkenden Fertigungskosten gestellt. Eine wirkungsvolle Automatisierung und Produktionssteigerung im Bereich der Fertigungstechnik wurde in den letzten Jahren zum großen Teil durch den Einsatz von numerisch gesteuerten Werkzeugmaschinen erreicht.

Die wachsenden Anforderungen an den Automatisierungsgrad und der zunehmende Umfang an auszuführenden Funktionen zur Beherrschung des Prozesses sowie die komplizierte Verflechtung einzelner Kenngrößen nach Optimiermodellen haben dazu geführt, Systeme zu entwickeln, bei denen Digitalrechner (Prozeßrechner) als informationsverarbeitendes Element in den zu steuernden Prozeß integriert sind (Bild 1) (1). Systeme, bei denen NC-Maschinen direkt von einem Rechner gesteuert werden, wurden in Anlehnung an "Direct Numerical Control" als DNC-Systeme bekannt (2).

2. Möglichkeiten der Kopplung von Prozeßrechnern mit Werkzeugmaschinen

2.1 Prozeßrechnersysteme

Der aus dem Digitalrechner entwickelte Prozeßrechner unterscheidet sich von diesem durch seine spezifische, auf den Prozeß zugeschnittene Peripherie. Im Gegensatz zum mathematisch-wissenschaftlichen bzw. kommerziellen Bereich müssen im technischen Bereich die anfallenden Prozeßdaten in den meisten Fällen vom Prozeßrechner in Echtzeit verarbeitet werden. Ein wichtiger Bestandteil der Prozeßperipherie ist ein Koppelelement (Prozeßelement), das Prozeßdaten in digitaler oder analoger Form ein- und ausgeben kann sowie ein Interruptsystem, das einen direkten Zugriff zum Kernspeicher und/oder zur Programmsteuerung vom Prozeß her erlaubt (Bild 2). Rechner, die zur direkten numerischen Steuerung von Werkzeugmaschinen eingesetzt werden sollen, müssen die obengenannten Eigenschaften aufweisen. Direkt numerisch gesteuert bedeutet hier, daß die für die Bearbeitung eines Werkstücks notwendigen Steuerinformationen schritthaltend mit dem Bearbeitungsprozeß von dem Rechner an die entsprechende Werkzeugmaschinensteuerung ausgegeben werden.

Da bei DNC-Systemen im allgemeinen mehrere NC-Maschinen mit einem zentralen Prozeßrechner verbunden sind (s. Bild 1), ist ein leistungsfähiges Interruptsystem erforderlich, um einen reibungslosen Prozeßablauf zu gewährleisten. Gerade laufende Programme müssen häufig zugunsten eines dringlicheren Programms unterbrochen werden. Dies geschieht aufgrund von Signalen (Interruptsignal bzw. Alarmsignal) aus dem Prozeß an den Rechner. Sie gelangen z.B. über das Koppelelement oder das Unterbrechungssystem, das aus zwei oder mehreren Ebenen (Registern) bestehen kann, in den Rechner. Die verschiedenen Ebenen oder Register können hardware- und/oder softwaremäßig aufgebaut sein.

Durch ein zeitlich ineinander verschachteltes Ablaufen der einzelnen Programme läßt sich eine Quasi-Simultanarbeit erreichen (3,4). Dies ist eine notwendige Voraussetzung, um die direkte Füi ung von mehreren numerischen Steuerungen im Echtzeitbetrieb durchzuführen.

Bevor verschiedene Kopplungsmöglichkeiten zwischen Prozeßrechnern und numerisch gesteuerten Werkzeugmaschinen erläutert werden, sollen im folgenden Abschnitt die Aufgaben und Funktionen von NC-Steuerungen kurz umrissen werden.

2.2 Numerische Steuerungen

Bei einer numerischen Steuerung werden die rein numerisch verschlüsselten Angaben der Arbeitsinformation direkt, ohne Zwischenschaltung eines Bedienungsmannes, von der Steuerung einer Anlage verstanden und automatisch verarbeitet. Durch Einlegen eines Informationsträgers, z.B. eines NC-Steuerlochstreifens, in die Steuerung einer Werkzeugmaschine kann ohne die sonst notwendigen und aufwendigen Vorrichtungen ein größeres Teilespektrum bearbeitet werden als mit konventionellen Werkzeugmaschinen. Die einfachste Steuerungsart, die Streckensteuerung, führt gesteuerte Bewegungen aus, die sowohl die nächste Position selbst als auch die Vorschubgeschwindigkeit betreffen. Für kompliziertere Bahnkurven kommen sogenannte Bahnsteuerungen zum Einsatz. Hierunter versteht man ein Steuerungssystem, in dem eine Bewegung in Abhängigkeit von zwei oder mehr Achsen so geführt wird, daß eine beliebige Kontur erzeugt wird. Die momentanen Positions- und Geschwindigkeitswerte müssen dabei von einem Interpolator für jedes Weginkrement neu berechnet werden. Sind die Maschinen nicht mit offener Steuerkette (Schrittmotorantrieb) für die Vorschübe ausgerüstet, so ist ein Meßsystem für jede gesteuerte Achse notwendig. Ein Lageregelkreis, der in der NC geschlossen wird, sorgt für das genaue Anfahren der Soll-Position. In Bild 3 ist eine konventionelle NC-Steuerung nach Funktionsgruppen aufgegliedert dargestellt.

2.3 Ankopplung konventioneller NC-Steuerungen an Fertigungsrechner

Die zur Zeit am häufigsten angewandte Methode der direkten numerischen Steuerung von Werkzeugmaschinen besteht in der Kopplung einer Anzahl konventioneller numerischer Steuerungen an einen Fertigungsrechner (DNC-Rechner) (Bild 4)

(5,6). Unter "konventioneller" NC-Steuerung ist eine übliche autarke numerische Steuerung zu verstehen, die hinter dem Lochstreifenleser mit einer Anschlußnahtstelle zur Rechnerkopplung versehen ist (Behind-Tape-Reader). Alle Funktionen der numerischen Steuerung bleiben dabei erhalten.

Bei einer größeren Zahl von NC-Maschinen nimmt der organisatorische Aufwand für die Verwaltung und Bereitstellung der Steuerlochstreifen umfangreiche Formen an. DNC-Systeme bieten hier durch die zentrale Pflege und Archivierung der Steuerinformationen wesentliche Vorteile. Auf einem Externspeicher hoher Kapazität (Platte oder Trommel) des Fertigungsrechners ist im allgemeinen eine große Anzahl von Werkstückprogrammen abgespeichert, die über den Rechner von jeder der angeschlossenen Steuerungen im Bedarfsfall abgerufen werden können. Der Rechner hat dabei die Aufgabe einer Verteilstation und eines durch den Kernspeicher realisierten Datenzwischenpuffers, von dem aus die einzelnen Bearbeitungssätze auf Anforderung hin über das Koppelelement und eine Datenübertragungsstrecke an die entsprechende Maschine ausgegeben werden. Test, Korrektur und Änderung von Steuerinformationen lassen sich bei DNC-Systemen ebenfalls effektiver gestalten, als dies bei der konventionellen NC-Fertigung der Fall ist. Über geeignete Ein/Ausgabestationen wie Fernschreibterminals oder Datensichtgeräte lassen sich Steuerinformationen direkt auf dem Externspeicher korrigieren und ändern. Das korrigierte Werkstückprogramm steht somit unverzüglich zur Verfügung.

2.4 DNC mit Rumpfsteuerungen

Eine der Möglichkeiten, die universellen Eigenschaften eines Prozeßrechners mit Vorteil einzusetzen, liegt in der Übernahme von Steuerungsfunktionen durch den DNC-Rechner. Auf diese Weise lassen sich bei den einzelnen Steuerungen an den Maschinen Baugruppen einsparen. Der mehrfachen Einsparung einer Baugruppe steht der einmalige Hardware- und Programmieraufwand für den Rechner gegenüber. Daraus resultiert eine Verringerung der Gesamtanlagenkosten. Die im Bild 3 eingezeichneten Nahtstellen deuten an, daß eine Vielzahl von Möglichkeiten besteht, Steuerungsfunktionen in den Rechner zu verlagern.

Bei der Ermittlung derjenigen Funktionen, die vom Rechner ausgeführt werden können, ist besonders die zeitliche Inanspruchnahme des Rechners bei der Ausführung der Funktionen zu berücksichtigen. Ein weiteres Kriterium ist der Arbeitsspeicherplatzbedarf, der bei der Steuerung einer größeren Anzahl von NC-Maschinen eventuell zu groß werden kann.

Als Beispiele für kernspeicherintensive Funktionen seien die große Anzahl logischer Verknüpfungen und der beachtliche Speicherbedarf für die Funktionen der Anpaßsteuerung bei Maschinen mit umfangreichen Werkzeugmagazinen und automatischer Beschickung genannt, z.B. bei Bearbeitungszentren, Interpolationen höherer Ordnung sowie Flächeninterpolationen. Die Berechnung von Bremsvorgängen oder Soll-Ist-Wertvergleiche in Regelkreisen sind Beispiele für Funktionen, die den Rechner selbst oder die Ausgabe von Daten über einen programmgesteuerten Datenkanal überlasten können.

Die im Bild 3 aufgezeigten Steuerungsfunktionen sind im allgemeinen bis zur Nahtstelle 6 hin für eine Verlagerung in den DNC-Rechner geeignet. Die Nahtstelle 6a deutet die Möglichkeit an, unter Verwendung von Stützpunktberechnungen, die Interpolation in eine Grob und Feininterpolation aufzuteilen, um den Rechner zeitlich zu entlasten.

In der letzten Zeit sind entsprechende NC-Steuereinheiten mit relativ geringem Hardwareaufwand entwickelt worden. Da derartige Steuerungen ohne Rechner nicht mehr voll funktionsfähig sind, werden sie als Rumpfsteuerung (bzw. MCU = Machine Control Unit) bezeichnet (2).

Der prinzipielle Aufbau einer Rumpfsteuerung für eine einfache Drehmaschine mit Schrittmotorantrieben ist in dem Blockschaltbild (Bild 5) dargestellt.

Die Steuerinformationen werden parallel in den Eingabeteil der Steuerung übergeben. Die Dekodierung und Aufbereitung der Steuerinformationen findet bereits im Rechner statt. Die Rumpfsteuerung führt lediglich die zeitintensiven Funktionen zur Abarbeitung der Informationen, wie die Ausgabe der Vorschubimpulse an die Verstärker und den Soll-Ist-Wertvergleich des Drehzahlregelkreises aus. Außerdem beinhaltet sie die Schaltbaugruppen zur Betätigung der Starkstromrelais und Schütze.

Der Einsatz von Rumpfsteuerungen wird besonders interessant bei zukünftigen Anlagen untereinander verketteter Werkzeugmaschinen (5). Hier ist die Zentralisierung der Steuerungsaufgaben besonders vorteilhaft, weil nicht mehr jede einzelne Maschine individuell bedient, sondern die Anlage als integriertes Fertigungssystem (s. Bild 6) von einer Warte aus gesteuert und überwacht wird. Man hat dadurch den Vorteil der besseren Überschaubarkeit der Fertigung und erreicht eine wesentliche Einsparung an Bedienungspersonal.

Neben der direkten Steuerung von Maschinen werden dem Rechner dabei noch weitere Aufgaben aus dem Fertigungsbereich übertragen, wie z.B. Betriebsdatenerfassung, Prozeßüberwachung und Fehlerdiagnose sowie die Regelung und

Optimierung des Prozeßablaufes in Verbindung mit Adaptive Control Systemen. Entsprechend müssen auch die Steuereinheiten auf das Zusammenwirken mit AC-Systemen ausgelegt werden. An verketteten Systemen mit Werkstücktransporteinrichtungen (Bild 6,8,9) besteht außerdem die Notwendigkeit, die Paletten-, Rollen- oder Manipulatoren-Transportsysteme einschließlich Be- und Entladeeinrichtungen sowie Pufferstationen zu steuern und zu überwachen. Hierfür eignen sich sowohl Rumpfsteuerungen als auch andere festverdrahtete Steuereinheiten, die direkt mit dem Fertigungsrechner verbunden sind (5).

Bei künftigen DNC-Systemen werden Steuerinformationen nicht vorwiegend einseitig durch den Rechner ausgegeben, sondern verstärkt ein Informationsfluß in beiden Richtungen notwendig. Das bedingt zusätzliche Register (Anzeige- oder Eingaberegister) auf der Steuerungsseite sowie Kopplungsmöglichkeiten für Display- oder andere Ein/Ausgabegeräte (Bild 7).

2.5 DNC-Systeme mit CNC-Steuerungen

Die Realisierung einer Steuerungseinheit mit festverdrahteten Baugruppen führt schnell zu umfangreichen und wenig flexiblen Spezialsteuerungen. Konventionelle NC-Steuerungen mit Lochstreifeneingabe führen sowohl Berechnungen als auch logische Verknüpfungen durch. Es liegt daher nahe, anstelle einer derartigen Steuerung einen kleinen Prozeßrechner einzusetzen. Durch die Verwendung von handelsüblichen Kleinrechnern kann man einerseits zu preisgünstigen Steuereinheiten kommen und andererseits ein hohes Maß an Flexibilität erhalten. Solche Steuerungen mit frei programmierbaren Kleinrechnern werden als CNC-Steuerungen bezeichnet (CNC = Computer Numerical Control) (Bild 8). Die Programm- oder Steuerdateneingabe kann über eingebaute Lochstreifenleser oder innerhalb eines DNC-Systems durch den übergeordneten Fertigungsrechner erfolgen. Betrachtet man die CNC-Steuerung als Sonderform einer NC, so kann man sie wie eine konventionelle NC-Steuerung über einen DNC-Zusatz mit dem Rechner verbinden. Werden den CNC-Steuerungen jedoch weitere Aufgaben (s.u.) übertragen, die einen starken Datenaustausch mit dem Fertigungsrechner erfordern, so ist es sinnvoll, den Kleinrechner über einen entsprechend leistungsfähigen Datenkanal mit dem übergeordneten Rechner zu verbinden. Die CNC-Steuerung und die Werkzeugmaschine stehen im allgemeinen räumlich dicht beieinander. Für den Signalaustausch kann man daher die Standard-Ein-/Ausgabeinterfaces mit bitparalleler Datenübertragung verwenden.

Mit CNC-Steuerungen lassen sich über die reine Maschinensteuerung hinausgehende Funktionen durchführen, deren Realisierung in einer konventionellen NC-Steuerung zu aufwendig wäre und deren Übernahme durch einen DNC-Rechner aus Gründen des Speicherplatzbedarfs oder der zeitlichen Anforderungen nicht in Frage kommt. Adaptive-Control Aufgaben und manche Funktionen bei der Betriebsdatenerfassung lassen sich vorteilhaft von CNC-Steuerungen ausführen. Bearbeitungszentren, die eine umfangreiche Anpaßsteuerung besitzen, um den Werkzeug- und Werkstückwechsel durchzuführen, können durch Kleinrechner kostengünstig gesteuert werden, wenn es gelingt, die zahlreichen logischen Verknüpfungen in einem beschränkten ASP-Bereich durchzuführen.

2.6 Fertigungssysteme

Das Fernziel der gesamten Entwicklung ist die vollautomatische Produktion. Das führt zu der Forderung nach flexiblen integrierten Fertigungssystemen mit automatischem Werkzeugtransport und Werkstückwechsel sowie automatischer Bereitstellung der Steuerinformationen. Derartige Systeme sind vor allem durch eine Verkettung sich in ihren Funktionen ergänzender und/oder ersetzender Einzelmaschinen gekennzeichnet. Die Steuerung derartiger Fertigungsanlagen ist praktisch nur noch mit Hilfe von Prozeßrechnern möglich (Bild 9). Die Tatsache, daß bei Störungen in der Fertigungsanlage, z.B. Ausfall einer Maschine, innerhalb kürzester Zeit der gesamte, dem Fertigungsrechner vorgegebene Fertigungsablauf korrigiert oder neu erstellt werden muß, macht in vielen Fällen die direkte Kopplung des DNC-Rechners mit einem übergeordneten Großrechner erforderlich. Eine derartige Rechnerkonfiguration ermöglicht folgende Aufgabenteilung: die gesamte Fertigungsablaufplanung, unter Einbeziehung der Produktions- und Personalplanung durch die kaufmännische Abteilung, erfolgt in der Großrechenanlage. Diese Informationen werden direkt an den Fertigungsrechner weitergegeben, der sie verwaltet und den Fertigungsablauf entsprechend steuert. In der entgegengesetzten Informationsflußrichtung werden dem Betriebsrechner Daten über Auslastung und Stillstandszeiten der Maschinen übermittelt, so daß die Betriebsführung laufend über den aktuellen Stand der Produktion informiert ist (5).

3. Datenübertragung

Im allgemeinen befindet sich der Fertigungsrechner eines DNC-Systems nicht in unmittelbarer Nähe der Steuerungen. Die Daten müssen über mehr oder weniger große Entfernungen übertragen werden. Von der Art und Geschwindigkeit der Datenübertragung werden Aufbau und Leistungsfähigkeit von on-line Steuerungssystemen, wie dies beim DNC-System der Fall ist, entscheidend mitbestimmt.

Besonders auffällig bei den bisher entwickelten Systemen ist die unterschiedliche Konzeption der Koppelelemente

Rechner - Steuereinheit und der Datenübertragung. Aufgabe der Koppelelemente ist es, als Code- und Pegelumsetzer zu dienen, die Datensicherung zu übernehmen und die zeitlichen Bedingungen des Rechners an die der Datenübertragungsstrecke und der NC-steuerung anzupassen. Daneben müssen die Koppelelemente selbst oder besondere Zusatzgeräte die Funktionen eines Multiplexers übernehmen. Es sind Systeme entwickelt worden, die theoretisch bis zu 250 Maschinen steuern können. Eingesetzt werden heute jedoch fast ausschließlich DNC-Systeme mit 4 bis maximal 20 Maschinen.

Bei DNC-Systemen mit verschiedenartigen Steuereinheiten wie Rumpfsteuerungen, CNC- und konventionellen NC-Steuerungen sowie Transport- und Lagersteuerungen sind besonders hohe Anforderungen an das Übertragungssystem und die Koppelelemente zu stellen, um die unterschiedlichen Steuersysteme optimal an den DNC-Rechner anzupassen (Bild 10). Dabei müssen z.B. auch Probleme der Prioritätssteuerung gelöst werden.

Da es zur Zeit keine einheitlichen Vorstellungen über die Schnittstellen gibt, wird sowohl die parallele als auch die zeichenserielle und bitserielle Datenübertragung im Halb- oder Vollduplexbetrieb angewandt. Dabei überwiegt die formatgebundene Übertragung mit Zeichenlängen zwischen 8 und 24 Bit. Die Zeichenübertragungsrate beträgt bis zu 10.000 Byte/s. Die maximal zulässige Entfernung zwischen Rechner und Steuerungen liegt zwischen 1 und 2 km. Besonders vorteilhaft für den schnellen Datenverkehr, der für den DNC-Betrieb gefordert wird, ist die blockweise Datenausgabe des Rechners, weil der Rechner dadurch zeitlich am geringsten belastet wird. Für die zeichenweise Ausgabe müßte jedesmal eine Ausgaberoutine durchlaufen werden. Der blockweise Datenaustausch eignet sich am besten für den Datenverkehr zwischen DNC-Rechner und CNC-Steuerrechner, weil zwischen beiden Rechnern nicht laufend Daten ausgetauscht werden müssen. Aufgrund der geringeren Speicherhardware an Rumpfsteuerungen und konventionellen NC-Steuerungen müssen bei diesen Systemen häufiger kleinere Informationsmengen (zeichen- oder satzweise) übertragen werden.

Bei der Auslegung eines leistungsfähigen und preisgünstigen Datenübertragungssystems für den DNC-Betrieb muß daher die Anzahl und die unterschiedliche Struktur der verwendeten NC-Steuereinheiten unbedingt berücksichtigt werden.

Schrifttum

(1) AUTORENKOLLEKTIV - Prozeßrechnereinsatz zur Automatisierung der Fertigung. Vortrag zum 14. AWK. Industrie-Anzeiger 93 (1971) 60

(2) Direktführung von numerisch gesteuerten Arbeits- und Prüfmaschinen durch Digitalrechner. VDI-Richtlinie 3424, April 1972

(3) LÜCKE, P. - Möglichkeiten beim Einsatz von Prozeßrechnern zur direkten numerischen Steuerung von Werkzeugmaschinen. Dissertation TH Aachen 1970

(4) REHR, W. - Echtzeitsteuerung von Fertigungseinrichtungen. Dissertation TH Aachen 1972

(5) PFEIFER, T. - Internationale Werkzeugmaschinen-Ausstellung in Tokio. Industrie-Anzeiger 95 (2/1973)

(6) PFEIFER, T., VERHAAG, E. - DNC-Systeme. Stand und Tendenzen rechnergeführter Werkzeugmaschinen. Industrie-Anzeiger 95 (3/1973)

(7) BÄCK, U. - Unveröffentlichte Forschungsarbeiten WZL - TH Aachen

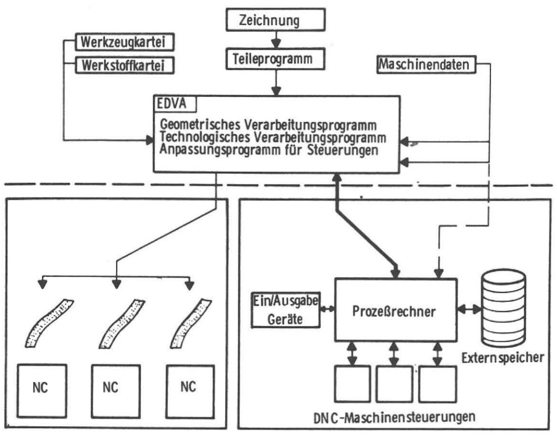


Bild 1:
Prinzipieller Aufbau der NC-Fertigung und der DNC-Fertigung

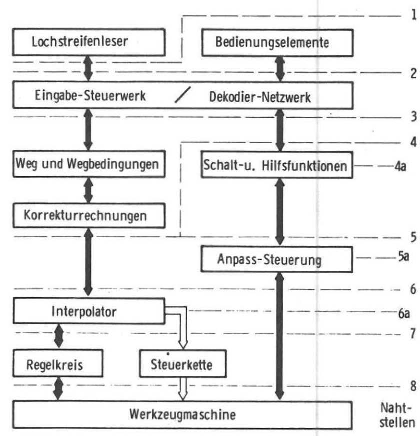


Bild 3:
Funktionsgruppen und Nahtstellen zur Rechnerkopplung bei einer NC-Steuereinheit

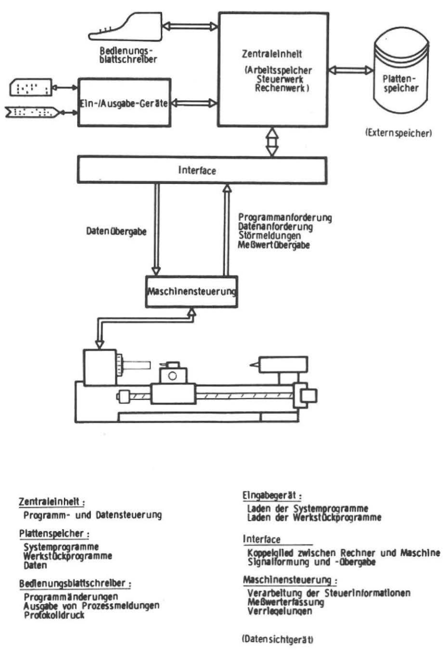


Bild 2:
On-line Steuerung einer Werkzeugmaschine durch einen Prozeßrechner

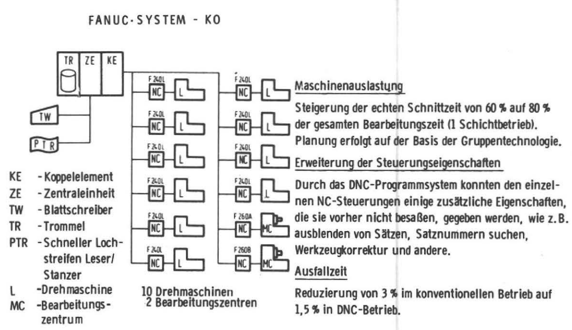


Bild 4:
DNC-System mit konventionellen NC-Steuerungen (IKEGAI IRONWORK)

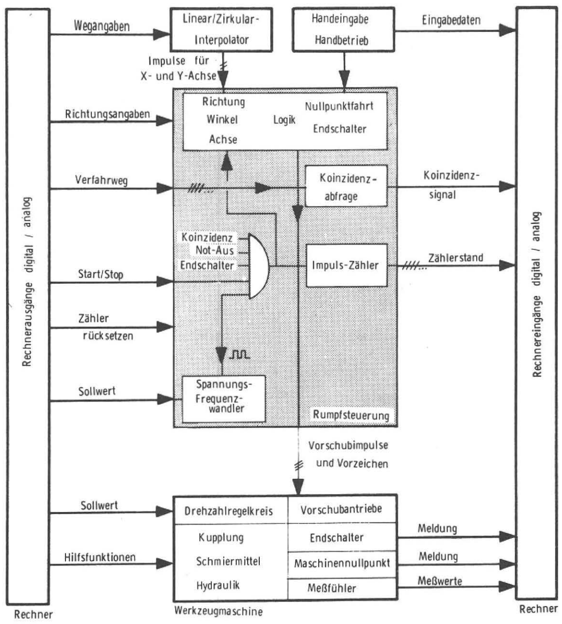


Bild 5:
Rumpfsteuerung für eine Drehmaschine

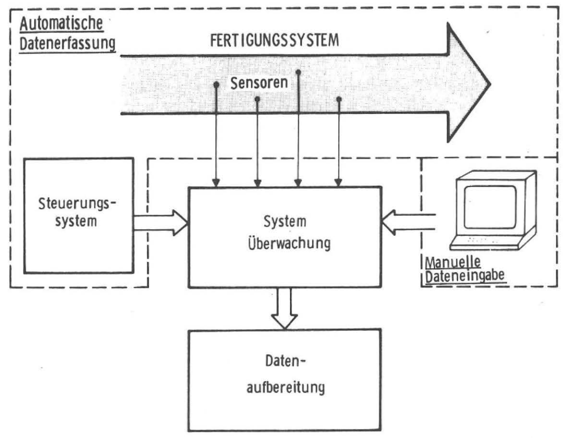


Bild 7:
Datenerfassung innerhalb eines Fertigungssystems

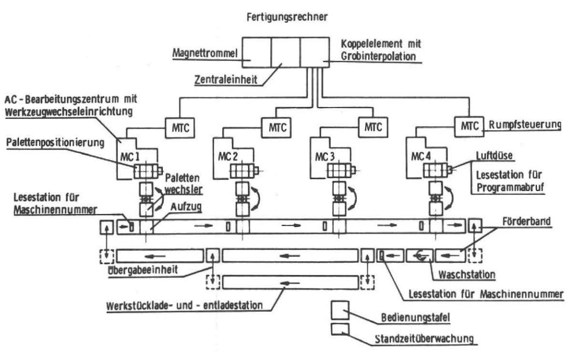


Bild 6:
Layout eines Fertigungssystems
(MAKINO MILLING)

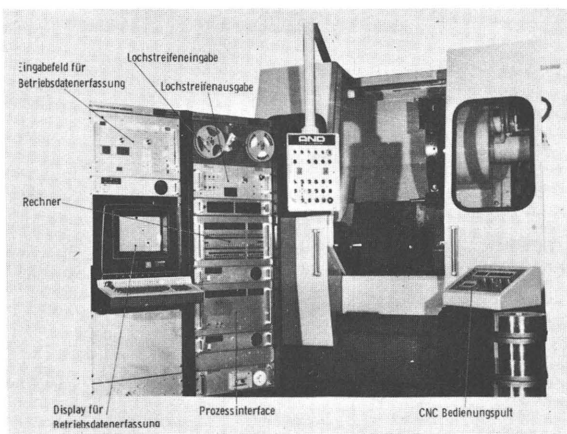


Bild 8:
CNC-Steuerung (WZL, TH Aachen)

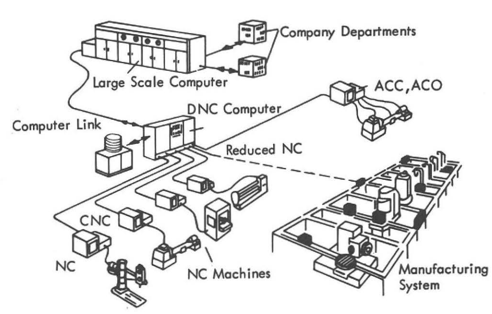


Bild 9:
Konfiguration eines integrierten Informationsverarbeitungssystems in der Fertigung (SIEMENS)

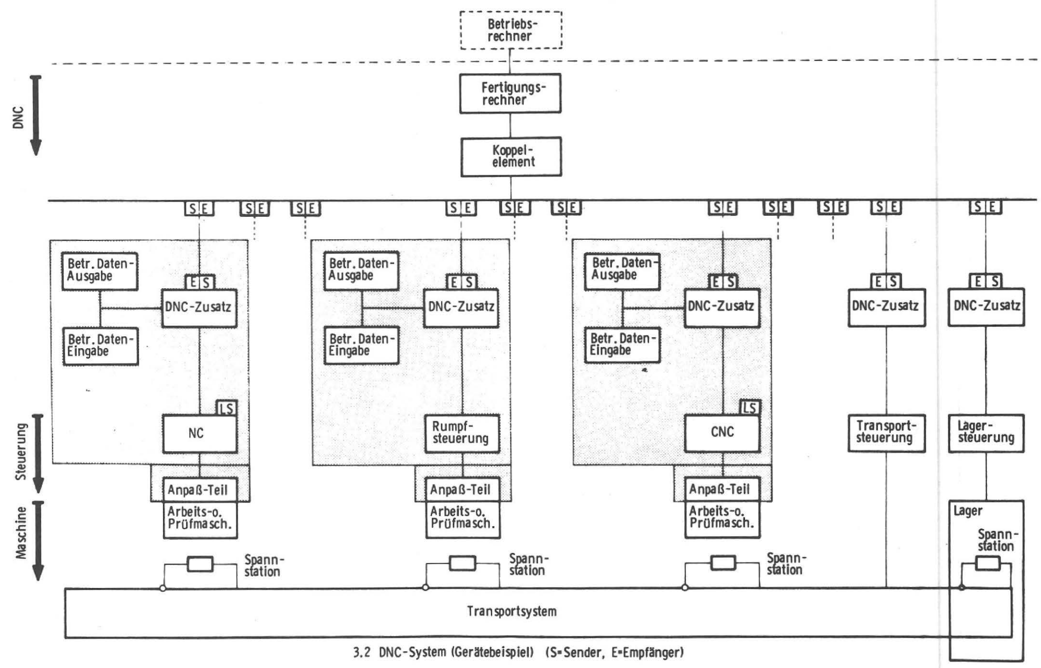


Bild 10: **DNC-Fertigungssystem mit verschiedenartigen Steuereinheiten: NC, Rumpfsteuerung, CNC, Transportsteuerung, Lagersteuerung**

DISCUSSION

Q - A. Pätzold

Das vorgestellte System soll in einer Ausbaustufe neben der Steuerung von Maschinen auch Aufgaben der Betriebsdatenerfassung und -verarbeitung und betr. den Materialfluss übernehmen. Sind bereits Untersuchungen durchgeführt worden, in wieweit der Prozessrechner ausgelastet ist? Können zusätzlich überhaupt eine grössere Anzahl von Rumpfsteuerungen in stark dezimierter Form angeschlossen werden? Diese würden ja erst eine Kostenverminderung einbringen.

A - K. Zenner

In Bezug auf die Bezeichnung 'das vorgestellte System': ich hatte hier nicht die Absicht ein endgültiges geschlossenes System vorzustellen. Ich wollte zunächst einmal Komponenten aufzeigen, die in DNC Systemen mehr oder weniger verwendet werden. Wenn Sie das Fertigungssystem ansprechen wollen, kann ich dazu bemerken, dass dieses System sowohl die direkte Maschinensteuerung als auch den Materialfluss und die Betriebsdatenerfassung in sich vereinigt, allerdings nur in sehr einfacher Form. Bei der Betriebsdatenerfassung wird nur die Standzeit mit einigen Uhren überwacht. Die Auslastung dieses Systems ist mir leider nicht bekannt. Zur Auslastung als Beispiel ein anderes gemischtes DNC System, allerdings ohne Betriebsdatenerfassung: Eine Untersuchung hat ergeben, dass bei 7 Maschinen bestehend aus je einer Rumpfsteuerung, einer CNC Steuerung, einem numerisch gesteuerten Zeichentisch und vier konventionellen NC Steuerungen sich bei dem Rechner eine Auslastung von etwa 5% ergab. Die Schwierigkeit bei der Erörterung der Auslastung ist dadurch gegeben, dass der eigentliche Engpass zwischen dem Massenspeicher und dem Rechner liegt. Dort traten Schwierigkeiten bei der bidirektionellen Datenübertragung auf. Für den zweiten Teil Ihrer Frage möchte ich ein Beispiel anführen. Eine Studie (Japan) ergab, dass ein System mit 12 Rumpfsteuerungen schon bei 4 Rumpfsteuerungen wirtschaftlicher ist als ein konventionelles System.

C - A. Pätzold

Das am Institut für Werkzeugmaschinen der Technischen Universität Berlin entwickelte System zur direkten numerischen Steuerung (DNC-System) von Fertigungseinrichtungen besteht aus einem Prozessrechner (AEG 60-10, 32K-Worte a 12 bit), einem Koppelelement, einem externen Massenspeicher (Magnetband z. Zt.) und Bedienungsperipherie. Zu den angeschlossenen numerisch kontrollierten Werkzeugmaschinen bestehen Datenleitungen. Das realisierte DNC System beinhaltet neben dem Anschluss unterschiedlicher Steuerungen (NC, CNC) auch die Geräte zur Betriebsdatenerfassung und Fertigungsführung. Durch die Einbeziehung des Materialflusses und die Uebergabe von Werkstücken in die Maschine mit Hilfe von Handhabungsgeräten (Industrie-Roboter) können die bislang unabhängig voneinander arbeitenden numerisch kontrollierten Maschinen zu integrierten Fertigungssystemen verknüpft werden.
Bei dem weiteren Ausbau dieser Systeme bzw. bei Neukonzeptionen werden mit Hilfe des CAMAC Systems eine Reihe von Problemen einfacher zu lösen sein.

APPLICATION OF CAMAC TO THE NUMERICAL CONTROL OF MACHINE TOOLS

J. G. Ottes
Gesellschaft für Kernforschung GmbH, Karlsruhe, Germany

ABSTRACT

In the field of numerical control (NC) and especially in direct numerical control (DNC) there is an increasing need for standardization. The numerous advantages of the CAMAC system as a very general interface between any computer and any process should be exploited by this application area. Digital position measuring systems, displays, and digitally controllable drives, clutches, and tool changing devices can most easily make use of the CAMAC standard. If existing problem-oriented programming languages shall be used in NC or DNC installations using CAMAC hardware, certain changes in the output part (post processors) of the software have to be performed.

ZUSAMMENFASSUNG

Auf dem Gebiet der numerischen Steuerung und insbesondere der direkten numerischen Steuerung wird eine Standardisierung immer notwendiger. Die zahlreichen Vorteile des CAMAC-Systems als einer sehr allgemein verwendbaren Schnittstelle zwischen einem beliebigen Rechner und einem beliebigen Prozess sollten für diesen Anwendungsbereich genutzt werden. Für digitale Positionsmessysteme, Sichtgeräte, digital steuerbare Antriebe, Kupplungen und Werkzeugwechselvorrichtungen können CAMAC-Systeme ohne Schwierigkeiten verwendet werden. Wenn bestehende Problem orientierte Programmiersprachen für die numerische Steuerung oder für direkte numerische Steuerung, mit CAMAC-Hardware benutzt werden sollen, so sind gewisse Änderungen am Ausgabeteil der Software erforderlich.

RESUME

Dans le domaine de la commande numérique (NC) et spécialement celui de la commande numérique directe (DNC), il existe un besoin croissant de normalisation. Il est souhaitable que les nombreux avantages du système CAMAC, interface d'usage général entre tout calculateur et tout processus industriel, soient mis à profit dans ce domaine d'application. Des systèmes codeurs numériques de position, des éléments de visualisation, des positionneurs à commande numérique, des mâchoires, et des têtes porte-outils peuvent aisément utiliser le standard CAMAC. Si des langages de programmation orientés existant actuellement sont utilisés dans des installations NC ou DNC utilisant des éléments CAMAC, il sera nécessaire d'effectuer certaines modifications dans les commandes de sorites (post processeurs).

1. General Considerations

In the field of numerical control of machine tools an increasing trend of replacing hard wired numerical controllers by general purpose mini-computers on account of many different reasons can be observed:

1. The major advantage of a computer in a numerical control (NC) system is that it brings much more flexibility into the system.
2. The adaption to a new workpiece can in most cases be done just by changing the software, not the system (1).
3. The price-to-performance ratio is monotonously dropping since years. Most process computers on the market offer adequate arithmetic power, speed and basic software for nearly all NC-tasks. Even a word capacity of 12 bit has been found to be sufficient though 16 or more bit seem to be more handy (2,3,4). Modularity of the computer memory and the input/output system is of great importance.
4. It is theoretically possible to run all NC-tasks with one computer system and this decision reduces the user's spare parts inventory. Besides only one system had to be learned by maintenance and operating personnel.
5. Last not least the use of a process computer opens the possibility to use standard interface hardware like CAMAC. The details of the CAMAC system may be found in the literature (5,6,7). For the purpose at hand some general aspects of the system shall be emphasized.

The CAMAC system is a modern answer to the old problem of interfacing process computers to their peripheral devices. It is a parallel bus system and not the only one in the world (8), but the only one which has received world-wide acceptance and gets continuous care of the ESONE*) and USAEC-NIM**) Committee. The CAMAC system breaks the two-dimensional interface problem into two one-dimensional pieces which are easier to solve. The general structure of the CAMAC system is shown in fig. 1. It is essential that the computer has no bearing whatever on the so called modules and vice versa. Modules are designed to perform a definite process task and interface to the DATAWAY. System Controllers are designed to meet the special requirements of a computer and interface to the BRANCH HIGHWAY. Once a module has been developed conforming to the CAMAC standard it can be used in all CAMAC installations without alteration. This means that for all modules in the CAMAC standard the everlasting, boring, and expensive interface problem does no longer exist. The mutual independency of modules and computers has been achieved by the invariable functional units DATAWAY, BRANCH HIGHWAY and CRATE CONTROLLER A. These units are the same in all CAMAC applications and tend to benefit from large scale production. The combination of computer technology and the CAMAC system looks very promising for NC applications. The modularity of the resulting system gives the greatest possible ease in maintaining, changing or expanding a given NC installation.

The CAMAC system features a 24 bit parallel bus system with an operating frequency ranging from 100 kc to 1 Mc. This is fast enough for machine tool applications which in general don't exceed a maximum feedrate of 100 000 pulses/s, each pulse being correlated to a specific unit-displacement of a slide. In many applications the maximum feedrates are significantly lower.

Another property of the parallel CAMAC Branch Highway is that it can accomodate only a relative short distance of maximum 25 m. This may be insufficient. The final definition of the CAMAC Serial Highway covering largely increased distances at the cost of transmission rate will mitigate the situation (s. fig. 2).

Parallel bus systems tend to be expensive, at least at first glance and in this respect the CAMAC system is no exception. But this drawback is surely ontweighed by the fact that the cabling of the system is uniform and no interfacing problems can arise.

Another drawback of all bus systems is their vulnerability. One defective line may shut down the whole system (s. fig. 3). In this context some built-in and easy to implement precautions may be mentioned: First of all it is possible to use the capacity of 24 bit to implement an error detecting or even error correcting code. Next the function code F(3) was originally intended for error detection in that sense that the exclusive-or-function of a pattern sent by the controller and the complement of that pattern read back from the module must yield all ones, if the system is working properly. The physical layout of the DATAWAY and BRANCH HIGHWAY, the outfit of the important timing lines S1, S2, BTA and BTBi, and the design of the typical DATAWAY timing cycle guarantee a certain immunity against noise and crosstalk. No bad experience has ever been reported since the days of the first CAMAC crates.

This short survey of the main properties of the CAMAC system should have shown that there is at least no serious obstacle in the way of implementing the functional units of a DNC-system with the means of the CAMAC standard. On the contrary a strong impetus should be drawn from it to use the combined merits of general purpose process computers and CAMAC for DNC. From what has been told it should be clear that only digital electronic NC or DNC units can successfully be converted to the CAMAC standard. It is not advisable to have a try with pneumo-hydraulic or fluidic

---

*) ESONE: European Standard on Nuclear Electronics

**) USAEC-NIM: United States Atomic Energie Commission on Nuclear Instrument Modules systems. In the next chapters examples of the conversion of typical NC hardware to the CAMAC standard are given, among them absolute and incremental position measuring systems, numerical displays, drives, clutches etc. A final chapter deals with some software aspects of such systems.

2. Position measuring devices

At present very many different position feedback systems are in use: analogue and digital, absolute and incremental, linear or rotary, each with specific advantages and drawbacks (see table I). It would by far exceed the frame of this essay to deal with the whole lot of them. For DNC purposes the use of (rotary) digital incremental systems or (linear) digital absolute systems seems to be most appropriate.

2.1. Linear incremental position feedback system

In fig. 4 the principle of a linear incremental position feedback is shown. A glass scale of sufficient length carries aequidistant pulse grating with period τ. Its counterpart is a scanning reticle with four groups of similar gratings distant $(n + 1/4)\ \tau$. The sinusoidal output of the single or anti-parallel photo diodes is fed to a preamplifier and a Schmitt-Trigger circuit (see fig. 5). The outgoing square waves are output on a cable. In the following electronics the direction of movement and pulses for an up/down-counter are derived from the two square waves. Thus the net position of the machine slide can be read from the up/down-counter. Preamplifier, Schmitt-Trigger and line drivers must be very near to the photo-diodes, in fact they are integrated in the fixed part of the transducer. Direction discriminator and up/down-counter can easily be designed to meet the CAMAC standard as indicated in the block diagram of fig. 6. The output of the direction discriminating logic controls the up/down-counter. The pulse generating network can be commanded to provide 1, 2 or 4 pulses per displacement unit τ. (A realistic value of τ is 8 µm). The overflow of the 24^{th} bit of the up/down-counter can be regarded as source of a LAM-signal which must be reset by the computer. A set of possible CAMAC commands to the incremental position feedback system is given in table II as an example.

Very often a numerical display of the slide position is desired. To achieve this the 24 bit binary information in the up/down-counter has to be converted to decimal form. This problem has already been solved and some single width binary-to-decimal converters are commercially available. It is recommended to the interested reader who wants a quick and complete survey of all producers of CAMAC items and all available modules to look at the CAMAC bulletin (9).

2.2. Absolute position transducer.

It is possible to eliminate the need for the up/down-counter by coding the etched strokes on the glass scale. Very frequently excess-3-code or pure binary codes are used (see fig. 7), the number of tracks depending on the unit displacement τ and the maximum displacement of the machine slide. It is a property of the binary code, that changes from transparent to non-transparent parts of the scale can often occur in many tracks simultaneously. Therefore all tracks but the first have two solar cells and a so called V-or Kliever-logic is provided to guarantee a safe position readout (s. fig. 7). This system has a big advantage: it is insensible against noise and voltage failure. As soon as the voltage recovers, the position readout is again valid. Incremental systems don't show this valuable feature. The excess-3-code does not demand two solarcells on each track and a V-logic for safe reading, for successive displacement increments effect a change in only one track respectively. But information in this code cannot be processed in the computer immediately but needs conversion, which may be a nuisance. Preamplifiers, Schmitt-Triggers and line drivers are again integral parts of the transducer and need not be altered.

What CAMAC can do in this case is indicated in fig. 8. it should be possible to incorporate the necessary electronics in a single width CAMAC module and this would be a significant reduction in volume.

3. Drives and actuators

During the last ten years a considerable engineering effort has been devoted to the development of incremental actuators like electrical stepping motors, electro-hydraulic stepping motors, and Minertia motors (10,11,12). These actuators are ideally suited for NC and DNC operations. In table III characteristic values like torque, steps per revolution and revolutions per minute for different types of incremental actuators are given. For many applications these values are sufficient. As an example an electric stepping motor is chosen. Each motor needs a special control unit which accepts pulses of variable frequency and converts them into appropriate signal combinations for the windings of the stepping motor.

3.1. Point-to-point and straight cut control

In fig. 9 a possible hardware configuration for a point-to-point control with a stepping motor is given. A register is looded with a number representing the desired displacement by a CAMAC write command if this is admissible at the moment. (It could

happen that the drive is still busy performing the last command). One bit of this register holds the direction information. Another unit of the device generates pulses of variable frequency which decrement the input register and simultaneously advance a ring counter. This procedure continues until the input register contains all zeros. The desired position has been reached. The computer ist notified of this via L-signal.
The variable frequency is thus responsible for the feed rate of the machine slide. It can be adjusted in a wide range from 0 to a maximum according to the needs of the machine tool, workpiece material etc. The frequency variation could be achieved by using a presettable prescaling network. The ring counter accepts the input pulses and produces signal combinations which are power-amplified and then supply the windings of the drive. In table IV the relation between steps and the five windings of the stepping motor type 109 of Fujitsu Limited, Tokio, is shown. This concept unburdens the computer to the greatest possible degree. All electronics except the power part should fit in a single width CAMAC module. Increased demands of torque may be satisfied e.g. by electro-hydraulic stepping motors (see table III), which could be operated by the same electronic unit without any change.

3.2. Continous contour control

Continous contour controls need an appropriate method of interpolation (linear, circular or any other). Essentially interpolators have to perform two tasks: 1. The given contour has to be broken into increments of suitable length. 2. The movement of the drives that generate the contour must be properly coordinated in time. One well known method of interpolating is called digital differential analysing (DDA), (13). Either hard-wired numerical controllers or process computers can do the job (1). As far as drives are concerned there is not much difference to positional or straight cut controls: The hardware interpolator or the program module of the computer has direct access to the control unit of the drives. One possible implementation is outlined as another example in fig. 10.

The idea behind these examples was to show that the control units of drives with the possible exception of the power feeders could easily be implemented in CAMAC.

4. Software Aspects

Since the early days of hard-wired numerical controllers the use of computers was felt necessary. The great number of different machine tools with their individual controllers on one hand and the number of different computers on the other hand caused compatibility problems. Here the need for standardization shows very convincingly. Facing these considerable difficulties the MIT Massachusets Institute of Technology in 1955/56 developed the strategy of subdividing the software solution of the problem generally into two parts. The first part is completely independent of the machine tool and is used to evaluate the cutter location data in a workpiece coordinate system. This part is called processor. The second part is called post processor which performs the adaption of the general cutter location data to the special controller (or computer) in question. It is in the post processor where the compensation of the actual tool diameter and tool length is done, where programmed speeds and feedrates are checked for validity looking at the dynamics of the machine tool slides. The post processor generates the necessary preparatory and miscellaneous functions and takes into account the code and input medium (paper tape or mag tape) of the machine tool control. The first problem oriented programming language implementing this concept is called APT (14) Automatically Programmed Tool). The post processors of APT and all APT - like languages have to be rewritten for each new NC-controller. Generally post processors contain the following functional modules (15): input element, auxiliary element, motion element, control element and output element.
The output element has to be rewritten, if the properties of the CAMAC system are to be taken into account. Here the great advantage of the CAMAC-standard can be realized: The CAMAC system provides virtual independancy of the machine tool! The essence being that only <u>one</u> post processor is needed. This statement applies to all problem oriented languages following the same design philosophy with processor and post processor like EXAPT, (16), MINIAPT (17), QUICK POINT-3 (2), and EASY PROG (3).

There may be good reasons for using assembler-type languages instead of one of the high level programming languages just mentioned. In this case it should be very easy to incorporate device handling routines which accept e.g. commands of the following form:

```
JMS CAMAC        /Jump to Subroutine CAMAC
CR(X,Y)-N(X,Y)/Crate No. Station No.
A(X,Y) F(X,Y)  /Subaddress Function
Wi               /Write Data
```

This is probably the quickest way of getting a CAMAC system into operation for DNC applications, but possibly not the most efficient.

Another promising approach of solving the software problems in conjunction with the use of CAMAC has been made by the definition of the CAMAC language (18). This language "deals with those statements in a program that are concerned with communication between the computer and .... CAMAC" and does not contain any data processing statements.

Thus it must be associated with a conventional programming language (host language). The first implementations of the CAMAC language are expected to show that the use of CAMAC is very much simplified.

5. Conclusion

The numerical control of machine tools (NC) is well within the scope of process computers and the market shows a growing trend of using direct numerical control (DNC). Thus it is only reasonable to exploit the additional merits of a standardized interface system like CAMAC. The advantages of using CAMAC in DNC may be expected in the three areas of position feedback, actuators, and software.

References

(1) KOSEM, MARION: Hard - wired Numerical Controllers yield to efficient Minicomputers. Electronics April 12 (1973)

(2) Quickpoint - 8; PDP-8 Part Program Compiler - User's Guide, Digital Equipment Corporation Maynard, Mass./USA. DEC - 08 - AUAA-D (1969)

(3) EASY PROG. - Das integrierte System für maschinelles Programmieren. Max Müller - Brinker Maschinenfabrik, Hannover (1969)

(4) NC-System Nixdorf 820 - Konturberechnungsprogramm Nixdorf Computer AG, Paderborn/Berlin (1970)

(5) CAMAC - A Modular Instrumentation System for Data Handling EUR 4100 e (1972)

(6) CAMAC - Organisation of Multi-Crate Systems EUR 4600 e

(7) CAMAC - A Modular Instrumentation System for Data Handling Specification of Amplitude Analogue Signals.

(8) U. SCHMIDT: Meßdatenerfassung und Steuerung programmierbarer Meßgeräte nach dem 'Partyline'-System. Regelungstechnische Praxis und Prozeßrechentechnik 5 (1973)

(9) CAMAC Bulletin - A publication of the ESONE Committee

(10) L. BOCK: Electric and Electro - Hydraulic Stepping Motors Steuerungstechnik 1 (1968) No. 1

(11) Stepping Motors and Digital Drive Systems. USM Corporation, Gear Systems Division, Bulch Street, Beverly, Mass.

(12) H. STÜBEN: Der Minertia-Motor als reaktionsschneller Antrieb für Werkzeugmaschinen. Antriebstechnik 7 (1968) No. 6

(13) GOETZ, ELMAR: Digital arbeitende Interpolatoren für numerische Bahnsteuerungen. AEG Mitteilungen 51 (1961) H. 1/2

(14) APT-Part Programming Manual. IIT Research Institute, Chicago (1964)

(15) W.H.P. LESLIE: Numerical Control User's Handbook McGraw Hill (1970)

(16) EXAPT 1/2: Sprachbeschreibung, Mai 1969 ; Verein zur Förderung des EXAPT-Programmierungssystems eV, 51 Aachen.

(17) NC-Software - MINIAPT, 5 Bände G. Horn - NC-Software GmbH, Frankfurt a. M. (1969)

(18) Proposal for a CAMAC Language CAMAC Bulletin, Issue No. 5, Nov. (1972) Supplement.

Table I: Survey of position feedback systems

Digital	incremental	rotary	Mainly photo-electric sensors
		linear	
	absolute	rotary	
		linear	
Analogue	incremental	rotary	magnetique or inductive sensors
		linear	
	absolute	rotary	
		linear	

Table II: Possible CAMAC commands to incremental position feedback system

Command	Effect
A(0) · F(0)	Read binary value of up/down-counter
A(1) · F(0)	Read position in BCD-code
A(0) · F(24)	Disable LAM (Overflow from up/down-counter)
A(0) · F(26)	Enable LAM
A(0) · F(8)	Test LAM
A(0) · F(10)	Clear LAM
A(11) · F(17)+W1	Generate 1 pulse per displacement unit τ
A(11) · F(17)+W2	Generate 2 pulses per displacement unit τ
A(11) · F(17)+W1+W2	Generate 4 pulses per displacement unit τ

Table III: Characteristique values for different types of incremental actuators

Type	Torque kp-cm	Steps per revolution	Revolutions per minute
Electric Stepping Motor e.g. Responsyn	15	2000	90
Electro-hydraulic Stepping Motor e.g. Type 109 of Fujitsu Ltd	200	160	750/3000
Minertia Motor e.g. Type JKMM 200 SR	2000	-	3000

Table IV: Steps and signal combinations for a five winding machine (10)

Step / Winding	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2
a	0	L	L	L	L	L	0	0	0	0	0	L	L	L	L
b	0	0	0	L	L	L	L	L	0	0	0	0	0	L	L
c	0	0	0	0	0	L	L	L	L	L	0	0	0	0	0
d	L	L	0	0	0	0	0	L	L	L	L	L	0	0	0
e	L	L	L	L	0	0	0	0	0	L	L	L	L	L	0

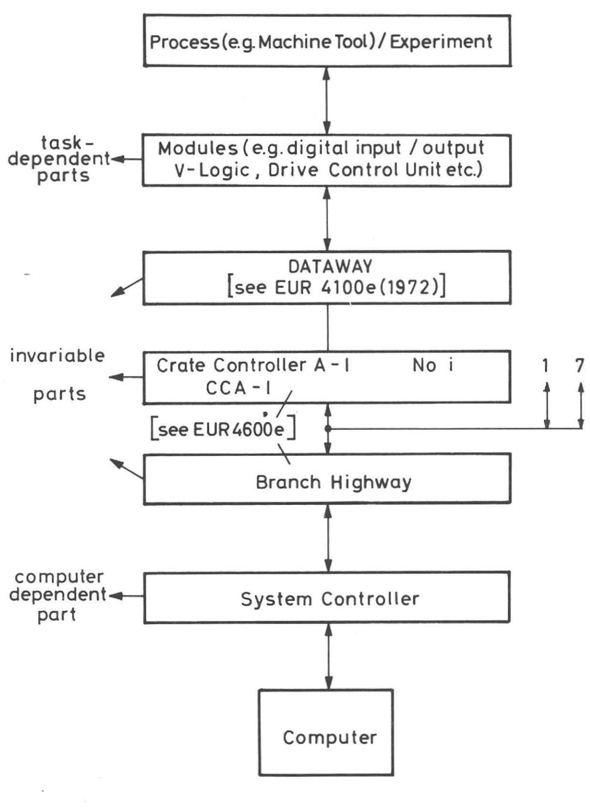


Fig. 1:
General structure of the CAMAC system.

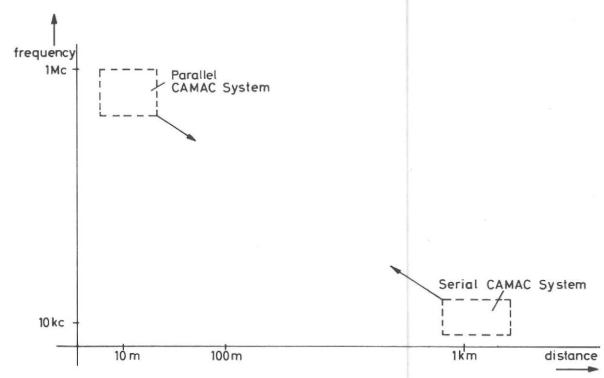


Fig. 2:
Operating frequencies versus distances in the parallel and serial CAMAC systems.

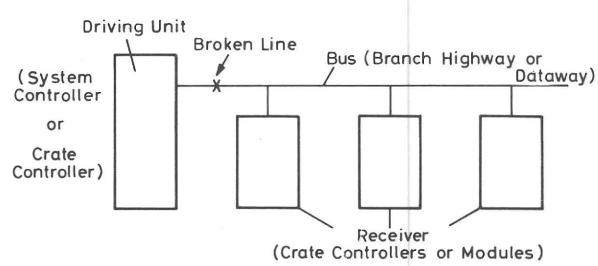


Fig. 3:
Sensitivity of bus systems against defective lines.

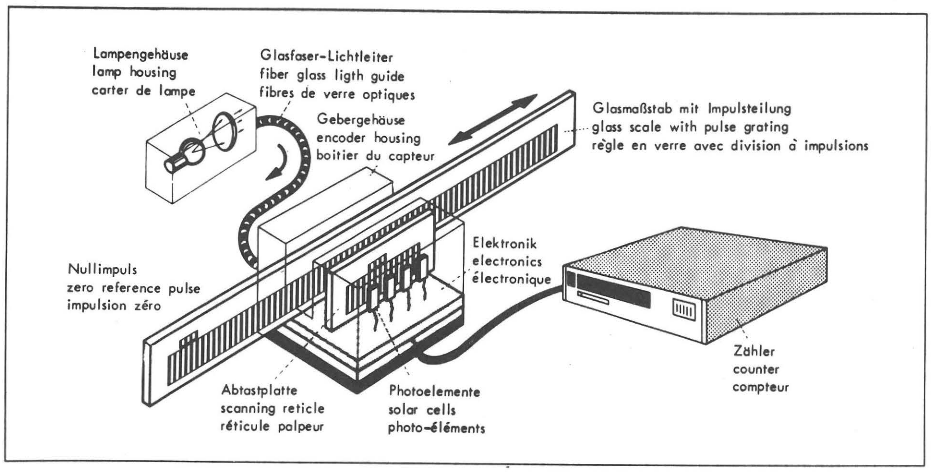


Fig. 4: Principle of a photoelectric incremental position transducer. (Courtesy of Joh. Heidenhain)

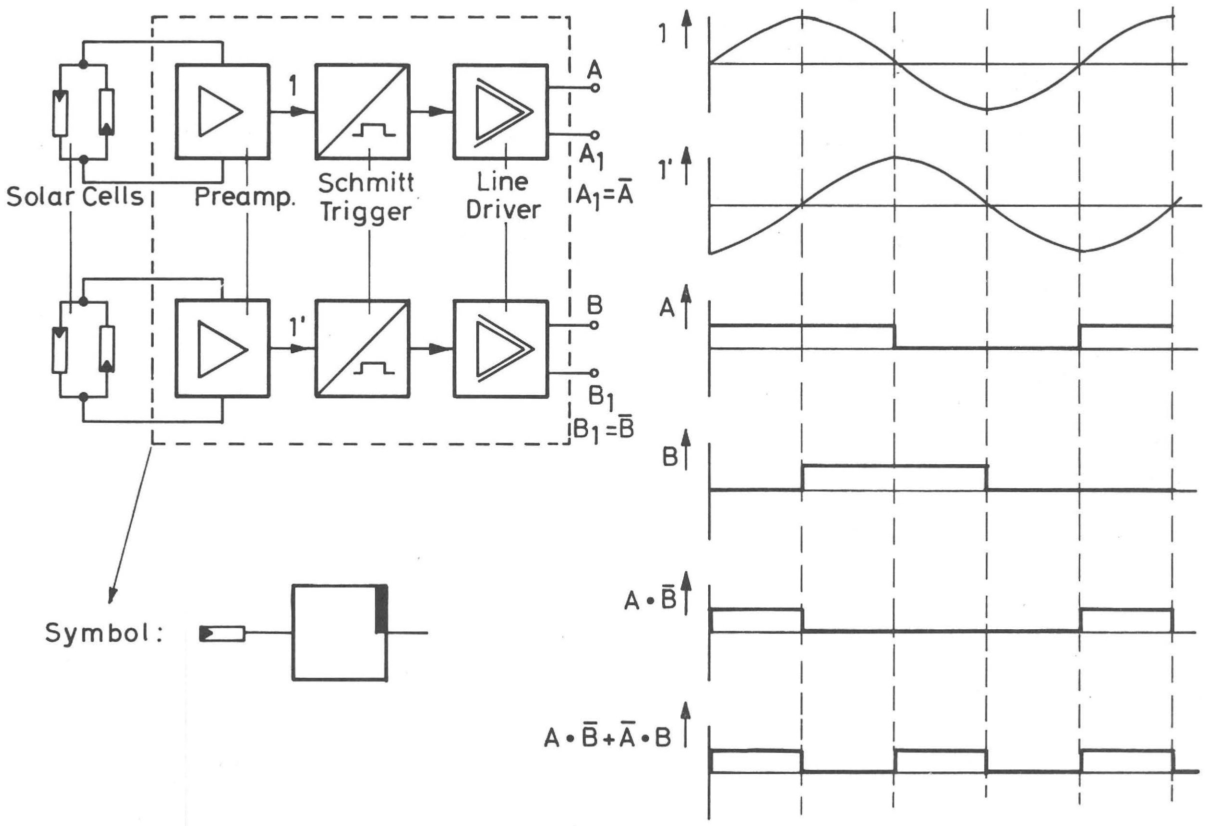


Fig. 5: Transducer electronics and output waveforms.

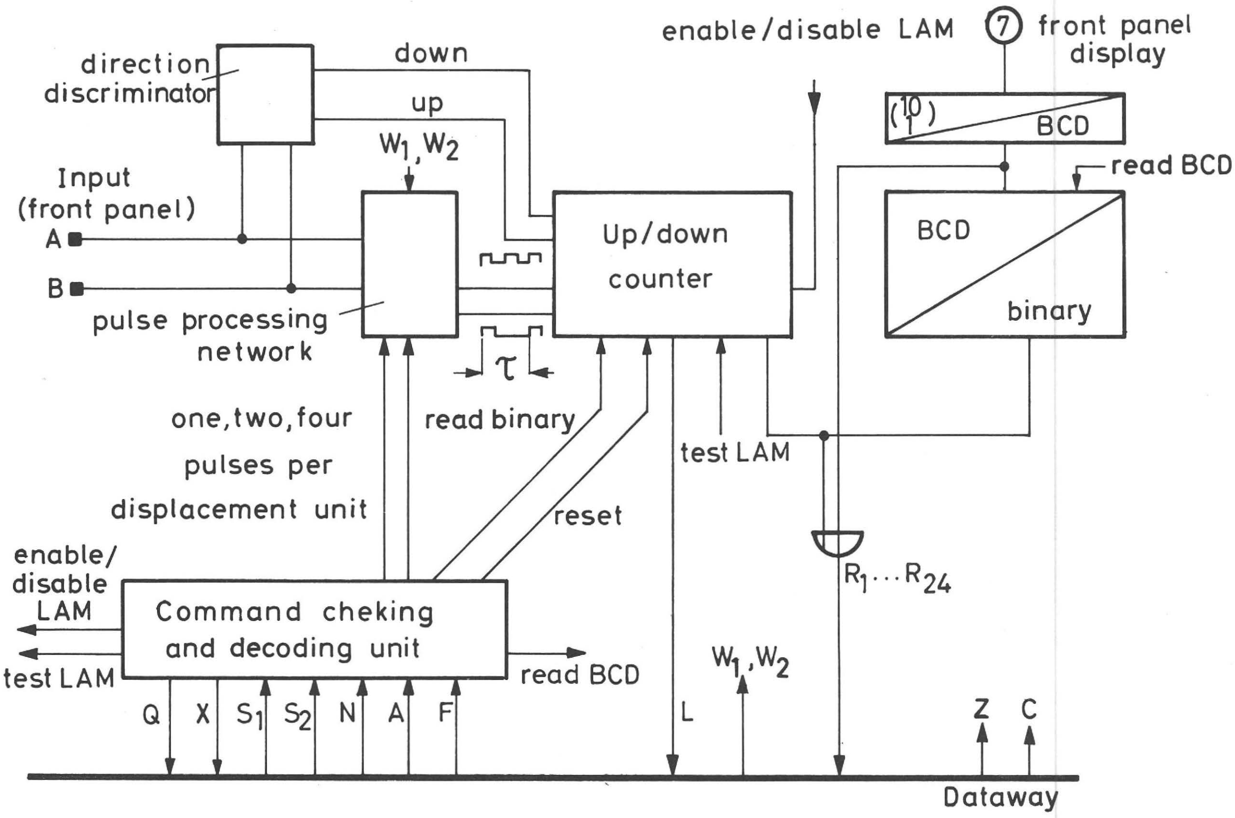


Fig. 6: Block diagram of incremental transducer electronic and display in CAMAC.

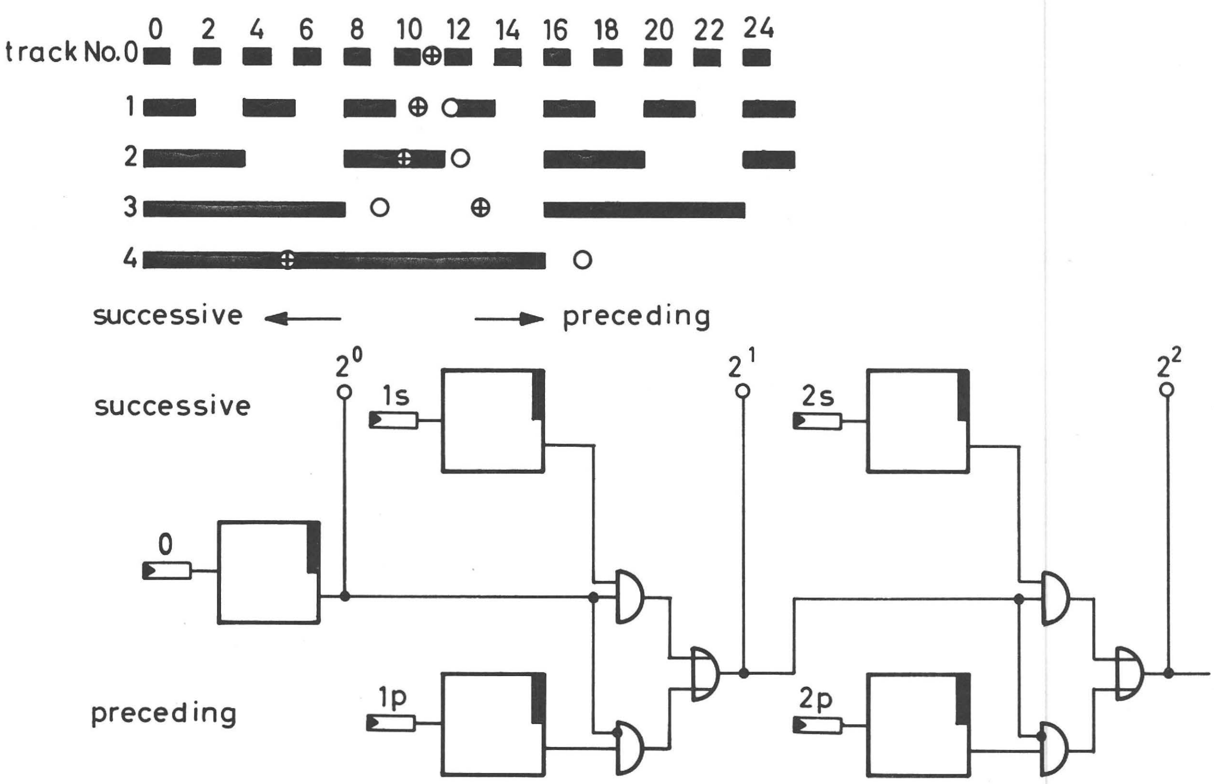


Fig. 7: Binary coded glass scale and V-(Kliever)-Logic

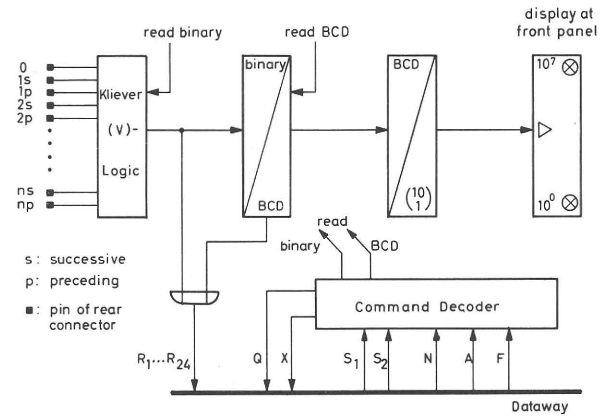


Fig. 8: Kliever-(V)-Logic and display of absolute position transducer in CAMAC.

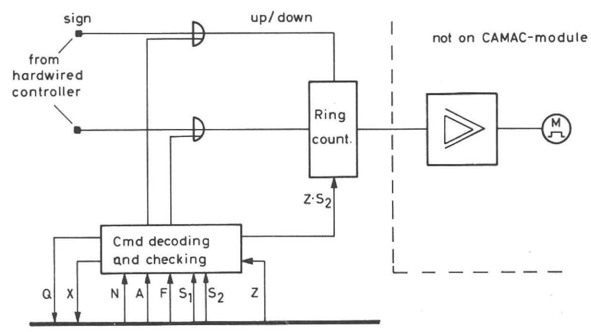


Fig. 10: Block diagram for continuous contour control with stepping motor in CAMAC.

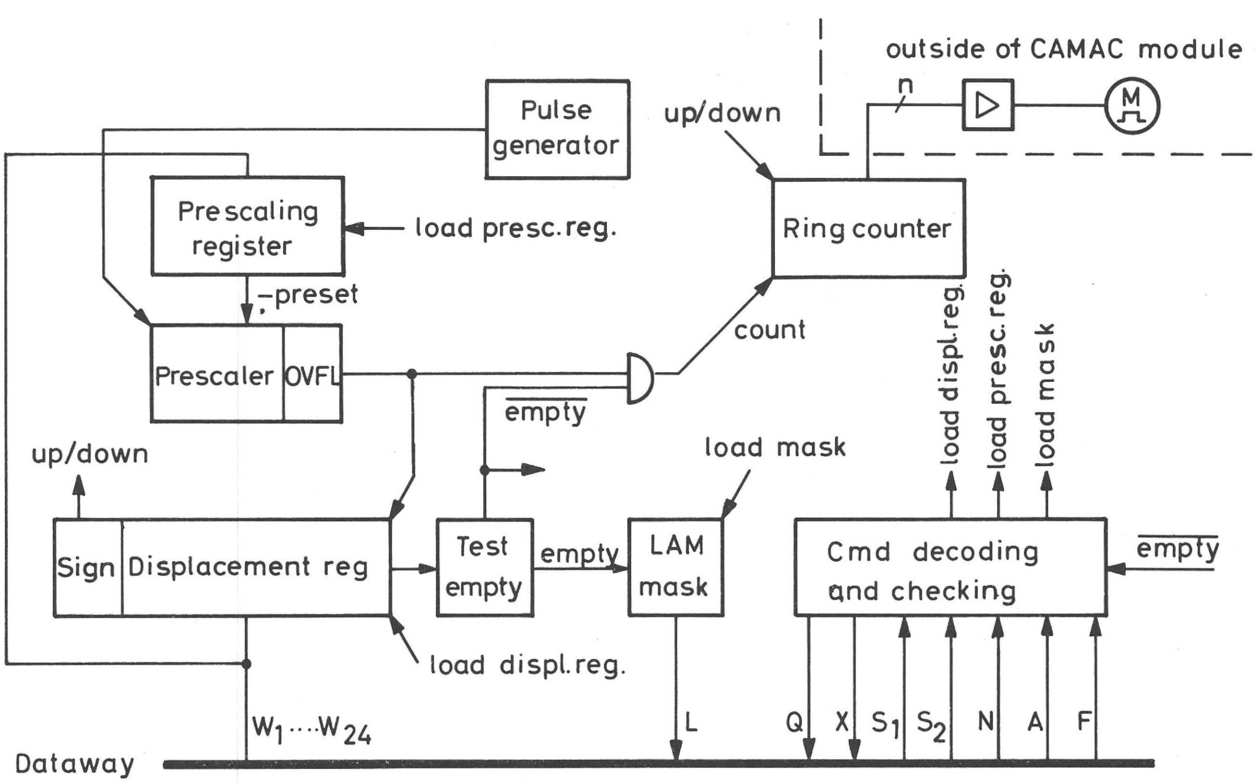


Fig. 9: Block diagram for point-to-point control with stepping motor in CAMAC.

SUPERVISION AND CONTROL IN UTILITY CENTRES

R. Thiel, G. Heinrich
Borer Electronics AG, Solothurn, Switzerland

ABSTRACT

Distribution networks of utility companies are growing fast in size and complexity. Supervision and control of their operation is becoming increasingly difficult. Hardware solutions to their automation do not offer the flexibility required for the continuing growth and structural changes. Using CAMAC as an interface between the network and computer, a solution is shown for the automation of electrical power distribution networks.

ZUSAMMENFASSUNG

Verteilernetze von Energieversorgungsunternehmen sind einem ständigen Wachstum unterworfen. Grösse und Komplexität dieser Netze bedingen den Aufbau von anspruchsvollen Überwachungs- und Steuersystemen. Hardware-Lösungen für die Automatisierung solcher Systemekönnen dem ständigen Wachstum und den laufenden strukturellen Veränderungen nicht gerecht werden. Mit CAMAC, als Anpassung zwischen Verteilernetz und Prozessrechner, wird eine Lösung vorgeschlagen für die Automatisierung von Elektrizitätsverteilernetzen.

RESUME

La supervision et la commande des réseaux de distribution d'énergie deviennent toujours plus problématiques, du fait de leur complexité croissante et de leur continuel agrandissement. Les moyens conventionnels d'automatisation ne peuvent plus s'adapter à ces changements de structure ainsi qu'aux modifications continuelles. En utilisant CAMAC comme interface entre l'installation et l'ordinateur, une solution est proposée pour l'automation des réseaux.

Supervision and Control in Utility Centres

1. Introduction

The efficient distribution of electricity, gas, water, oil, etc., not only requires extensive distribution networks, but also necessitates very flexible, fast-acting supervision and control organizations to operate them. Such organizations can effectively be supported by the employment of computer controlled systems for the preparation of decision data, the suggestion of reaction courses, and/or the automatic control of the networks' operations. Employment of the CAMAC-system as interface between distribution network and computer makes the economic realization of computer supervised and controlled networks practical.

2. Characteristics of Networks

Typical characteristics of energy distribution networks are:

- Wide geographical distribution
- Continual growth
- Complex enmeshed lay-out
- Demand for high operational security
- Unequal loading of the loops and branches
- Variation of the loads with time
- Critically balanced load distribution, and therefore sensitivity toward suddenly implied load changes

Scanning through this list of characteristics it becomes clear that the economic operation of such networks requires careful real-time optimization of supply and demand. In order to arrive at this optimization, a number of basic functions must be fulfilled:

- data acquisition
- data transmission
- data processing
- evaluation of results

2.1 Data acquisition at geographical widely distributed data source points whose number increases continuously with time. Thereby, the following three types of data are to be collected:

- Messages, or Status Reports (Figure 1) (e.g. positions of power switches and circuit breakers, warning- and fault signals, protection signals, condition reports and limit information, etc.)
- Measurements (Figure 2) (e.g. analog and/or digital information on values of voltages, currents, effective and reactive powers, displacement factors, etc.)
- Metering Data (Figure 3) (e.g. positions of counters, counter operations, etc.)

2.2 In order to relate the collected data from various sources to each other it must be brought to a centrally located processing place, which means Data transmission. This, again requires conditionning of the data before transmission, establishment of data transmission paths between data sources and data sink, and retransformation of the received data for easy handling.

2.3 Processing of the collected data may be achieved by means of pure hardware circuits or, preferably, by mini-computers.

2.4 Evaluation of the results and preparation of overall display information and/or control commands may be shared by man and electronic circuitry or taken over entirely by the latter.

While data acquisition and transmission is already automated, data processing and especially the evaluation of the results are still largely manually performed functions.

3. Possible Solutions to Processing and Evaluation

Within the network supervision and control center the received data may be processed, evaluated, and acted upon basically in one of the following two ways:

3.1 Manually Operated (Figure 4)

- The received data is transformed for display and given to individual indicators and instruments.
- The information content of the indicators and instruments is visually read by man.
- Man makes the decision on actions to be taken
- Man executes the actions and supervises the results
- If necessary, man corrects his actions
- A manually written log of the above operations is established

3.2 Computer Assisted or Controlled (Figure 5)

- The received data is transformed for handling and automatically tagged with the exact time of arrival (important for the "post mortem review" of failures).
- The data is then sorted out, evaluated, and condensed by a computer
- The condensed information is displayed while at the same time the data points are continuously supervised and a clear-text log is printed out showing the happenings in chronological order.
- Finally, the suggested course of action and/or direct control orders are issued by the computer. For manual actions, programmed switching may be employed, allowing man to initiate collective

commands for complex switching operations while the computer systematically issues all the necessary orders, considers all the status reports returned and, in the event of a mal-function, generates an alarm after, perhaps, trying to retrieve the situation.

4. CAMAC-Equipment as the Interface

In computer assisted or controlled systems, as described under paragraph 3.2., special attention must be paid to the design of the interface between the computer and the outside world. Taking an electrical power distribution network as an example, we find that messages, measurements, metering data and metering pulses may arrive at the center via data-links, or may be directly available from local data sources. The interface circuits via which all this data is to be transferred to the computer can be considered as part of the entire processing equipment. Whether, or not, this processing equipment can be operated economically depends largely on the capabilities, qualities and flexibility of the interface. Here, CAMAC equipment offers a universal, standardized solution having all the desired characteristics.

Data can be accepted as analog or digital information, as parallel bits (word) or as serial bits (pulse train).

- Switch position indications are received as digital information and accepted via reed relais or opto-isolators to provide the necessary electrical isolation demanded by the authorities. This type of information may be read by means of programmed scanning, or it may itself cause an alarm (LAM) so that immediate reading into the computer memory can be initiated.
- Measurements or counter positions are often forwarded as bit-words (e.g. 24 bit parallel) and can be accepted by CAMAC modules without further conversion.
- Information from remote data sources may have the shape of parallel bit words or pulse trains. Both kinds can be accepted by CAMAC modules that are available on the market.
- Data that is transmitted in serial form is decoded and converted to suit the computer. At the same time it may be checked for errors due to transmission.
- Quite often, pulses are presented which represent a measure of "work done". Here CAMAC counters may be used which will directly forward the metering data to the computer.
- Analog values are converted in integrating ADC's which are built to operate in a noisy environment without being disturbed.
- Since such analog measurement values quite often change very slowly, a single ADC unit may be used for the conversion of a large number of individual measurements simply by adding one or more CAMAC multiplexer units.
- For commands and information issued by the computer, digital output modules can be employed that emit either maintained levels or appropriate coded signals.
- Other output modules forward measurement values to pen-recorders or instruments as either analog or digital signals.

The above few examples indicate the kind of CAMAC modules required for the supervision and control of energy distribution. The large variety of equipment in use for the initiation, transmission, and transformation of such data can be met by the quite impressive line of CAMAC modules that are offered on today's market. Furthermore, CAMAC equipment facilitates the connection of computers of various makes without having to change the prime-interface modules.

5. Comparison

Table I shows a comparison between hardware equipped supervision and control centers and those employing a computer controlled system. From this table it becomes clear that for larger centers a computer controlled system can provide better service than a man-operated center with respect to

- reaction time upon alarms
- initial costs and operating costs
- operational security

TABLE I: Comparison between Hardware and Computer equipped Centers

Criterion: Description	Criterion: affecting	Fully Hardware equipped Center	Fully Computer equipped Center
information clarity	reaction time	large number of instruments and indicator lamps	sorted-out and condensed information; clear-text protocol
decision time		relatively slow	instant decisions
initial costs	initial costs and operating costs	relatively low (large numbers of equivalent instruments and indicator-lamps)	relatively high for processing equipment; savings on instruments
economical distribution		man controlled; not fully optimized	optimized, supply and demand balanced
flexibility		rearrangements or additions of instruments and indicators	highest flexibility; modular interface concept and software
equipment variety		readily available	readily available
project planning		straight forward	simple; modular building blocks
installation and cut-over		time consuming large number of individual connections	fast and simple computer test routine
maintenance-personnel training		expensive	relatively simple routine tests by computer
reliability	operational security	reliable, no protection against man's tiredness	MTBF of equipment
erroneous analysis		possible,	none, "post mortem analysis" possible
erroneous reaction		possible,	none
control commands		may be reduced to one order per case	reduced to one order per case
supervision continuity		not guaranteed	guaranteed within limits of MTBF

6. Conclusion

Distribution networks of utility companies require a highly flexible and fast operating supervision and control organization for security reasons as well as due to economical considerations. As initial costs for computers are decreasing rapidly, computer controlled systems tend to relieve man more and more from routine work and, at the same time, promise to prevent human errors while still reacting much faster to unforeseen conditions than man could do.

Using CAMAC to interface computer and distribution network, a solution was shown for the automation of electrical power distribution networks.

References

1. Heinrich G. — Einsatz von Kleinrechnern mit modularer Peripherie (CAMAC) im Netzbetrieb.
Elektritzitätswirtschaft
Bd. 71 (1972) II, Seiten 313 - 316

2. Heinrich G. — Borer 1000, Mini-Computer System for Energy Distribution Automation
Pub: 902.8.004.9.72.

3. Barnes R.C.M. — A Camac Glossary
Camac bulletin, issue No. 7, July 1973
Supplement

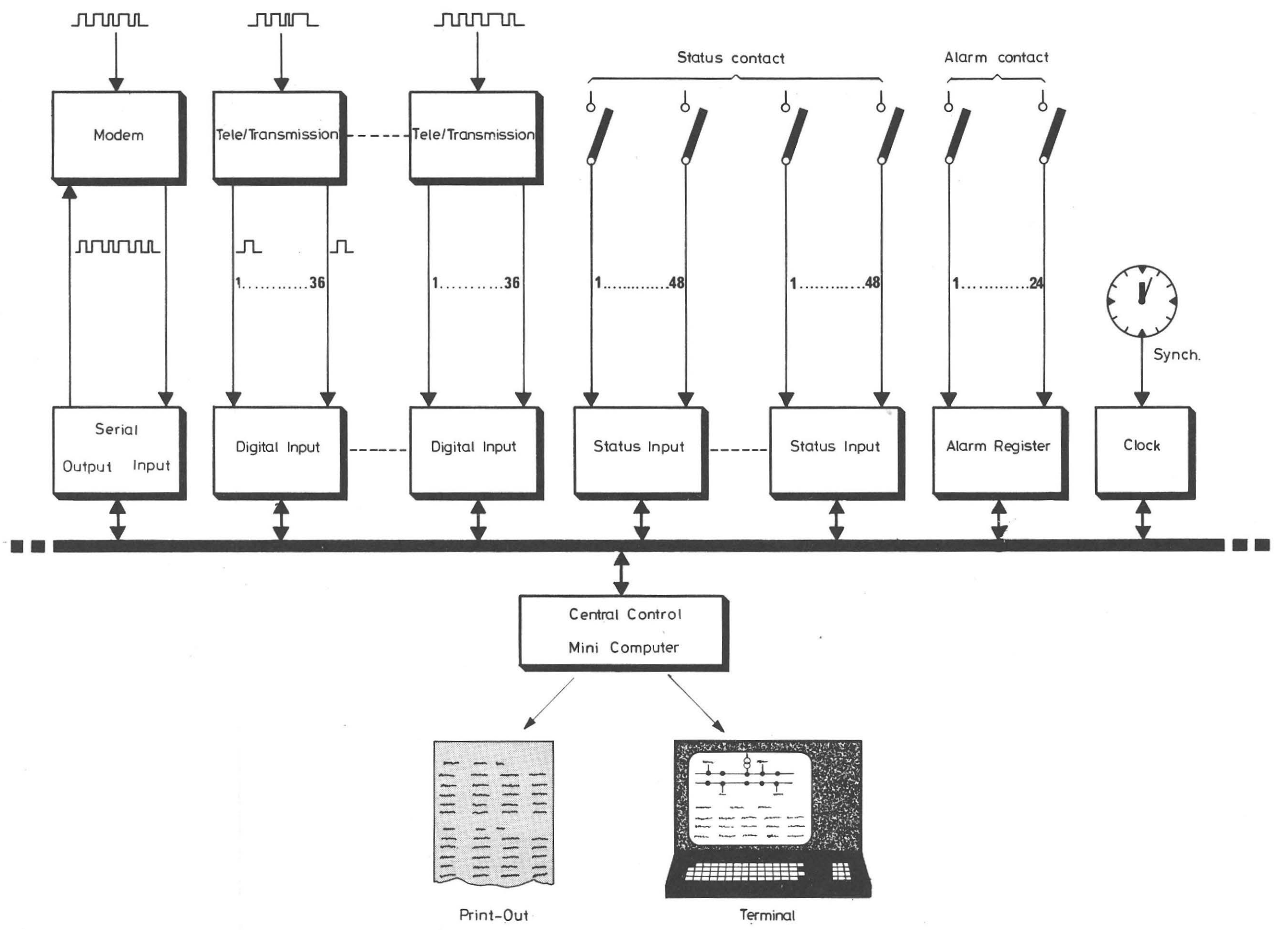


Fig.1 - Data acquisition:
Status reports

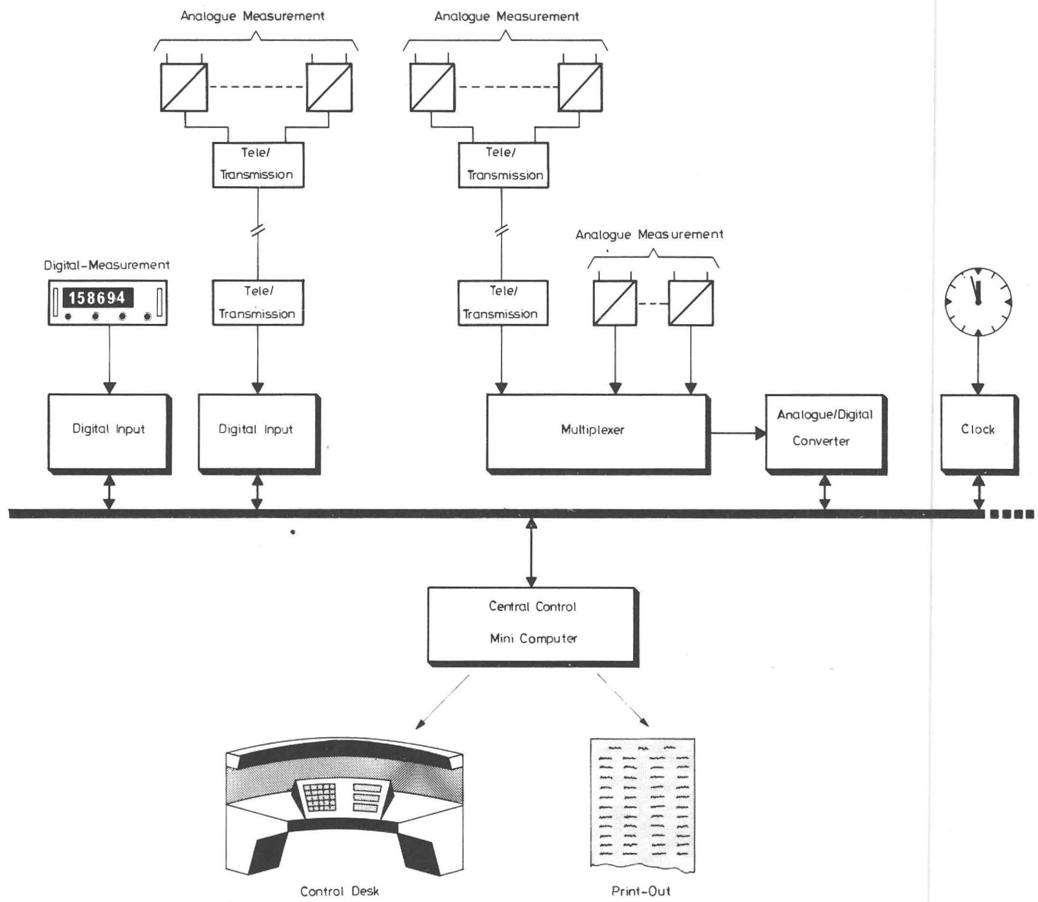


Fig.2 - Data acquisition:
Analogue and digital data

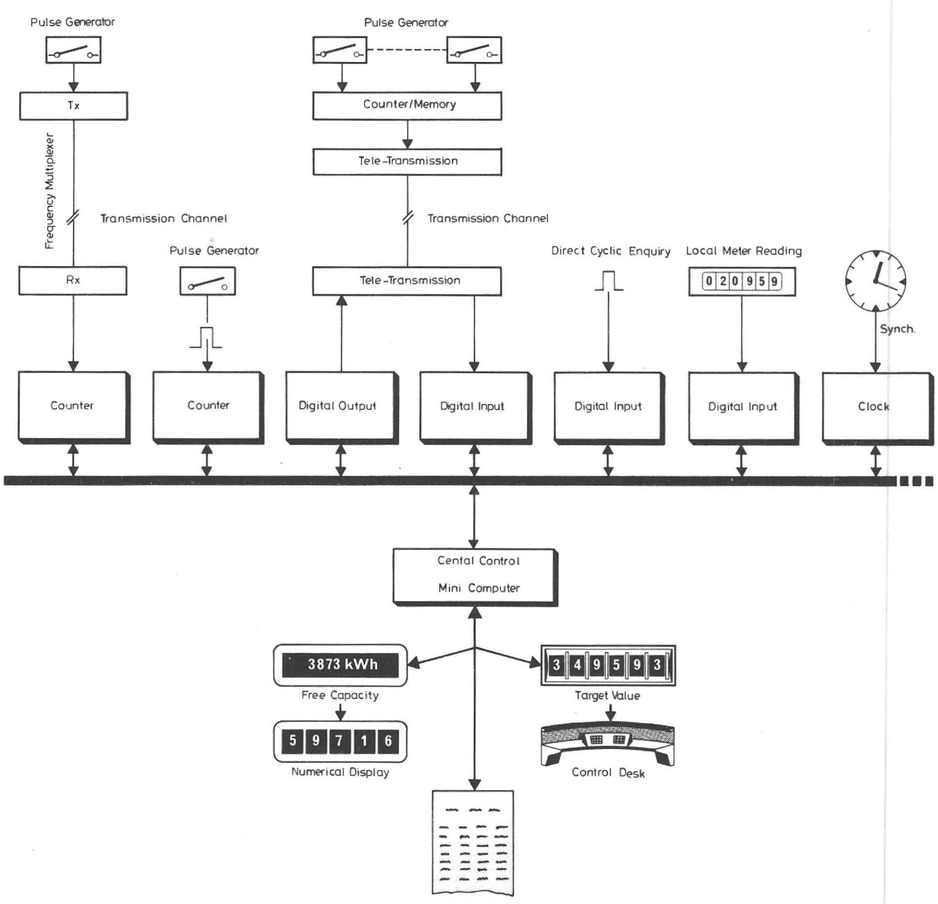


Fig.3 - Data acquisition:
Metering data

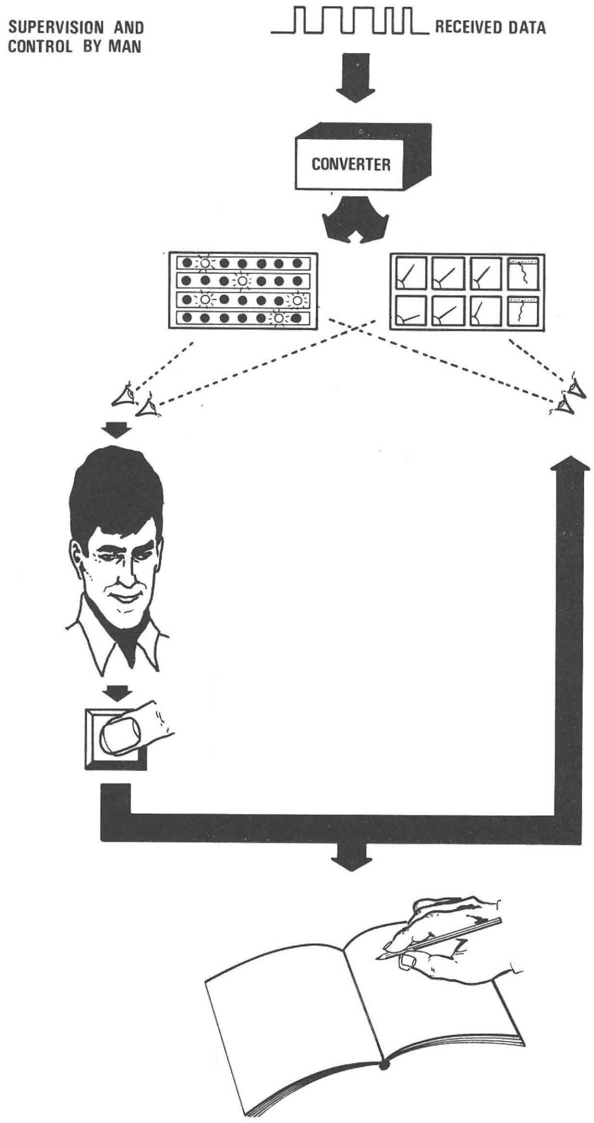


Fig.4 - Manually operated utility center

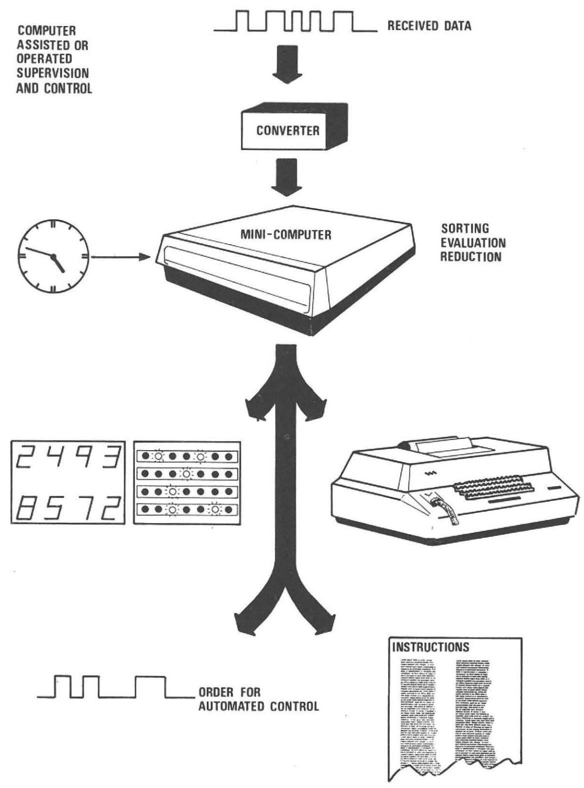


Fig.5 - Computerized utility center

INDUSTRIAL DATA LOGGING SYSTEM

J. Biri, J. Lukacs, L. Somlai, Gy. Vashegyi
Central Research Institute for Physics, Budapest, Hungary

ABSTRACT

The use of CAMAC systems in Hungary is related not only to nuclear applications. There are several systems working and being planned in industrial data logging, chemical, psychological and other laboratories and automatic measuring systems. One of the most interesting ones is a conventional power plant data logging and alarm system. The plant has 215 MW units, heated by oil or gas. Every unit has a separate data logging and alarm system based on CAMAC real time peripherals and a small computer.
The system has two purposes. It gathers data regularly in preset time intervals, calculates corrections, dimensions, linearizes data and prints different types of reports. It also calculates more complex quantities, e. g. efficiency and prepares data for later use. The operator has possibility to display interesting data continuously. The other purpose of the system is to detect alarm situations and give warnings. In this case it also "freezes" a post mortem report and changes the sequence of data gathering corresponding to the alarm situation.

ZUSAMMENFASSUNG

Wie die Anwendungen zeigen, ist CAMAC in Ungarn nicht mehr auf die nukleare Technik beschränkt. Mehrere schon arbeitende geplante Systeme werden als industrielle Datenerfassungssysteme, automatische Messysteme in chemischen, psychologischen oder anderen Laboratorien benützt.
Das Überwachungs- und Alarmsystem eines Kraftwerkes ist eine der interessantesten Anwendungen. Das Kraftwerk hat Blöcke mit 215 MW Nennleistung. Die Feuerungseinrichtung ist geeignet um Öl oder Erdgas, bzw. beides beliebigerweise gemischt, zu verwenden. Jeder Block hat ein eigenes Überwachungs- und Alarmsystem, welches auf CAMAC-Prozessperipherie und Kleinrechner basiert ist.
Das System dient zur Durchführung von zwei Aufgaben: eine von diesen ist: Datenerfassung mit bestimmter Zykluszeit, Korrektionsberechnung, Dimensionierung, Linearisierung und Protokollführung verschiedener Art. Ausserdem können noch kompliziertere technische Berechnungen, z. B. Wirkungsgradberechnungen durchgeführt und Daten für weitere Verarbeitung vorbereitet werden. Der Operator hat die Möglichkeit den Momentanwert von wichtigen Parametern kontinuierlich darzustellen. Eine andere Aufgabe ist die Erkennung der Alarmsituationen und deren Meldung. In diesem Fall wird auch ein "Post-Mortem"- Protokoll geführt, und entsprechend den Alarmsituationen werden die Parameter der Datenerfassung verändert.

RESUME

En Hongrie, l'utilisation des systèmes CAMAC n'est pas limitée aux applications nucléaires. Différents systèmes sont déjà en exploitation et d'autres se trouvent au stade du projet dans différents domaines, par ex. : enregistrement des données industrielles, systèmes de mesure automatiques dans les laboratoires de chimie et de psychologie et dans d'autres laboratoires.
Un des exemples les plus intéressants nous est fourni par l'enregistrement des données et par le système d'alarme d'une centrale électrique classique. La centrale est composée de différents blocs de 215 MW; elle fonctionne au mazout ou au gaz naturel. Chaque bloc possède son propre système d'enregistrement des données et son système d'alarme dont les installations de base sont des périphériques CAMAC en temps réel et un petit calculateur.
Ce système permet d'atteindre deux objectifs: il rassemble les données à des intervalles de temps réguliers, fixés d'avance, il calcule les corrections et les dimensions, il linéarise les données et imprime plusieurs types de rapports. En outre, il calcule des paramètres plus complexes, l'efficacité par ex. et il prépare les données qui seront traitées plus tard. L'opérateur a la possibilité de visualiser à tout moment les données intéressantes. Le deuxième objectif réalisé à l'aide de ce système est de détecter les situations d'alarme et de fournir des avertissements. Dans ce cas, le système "gèle" un rapport d'autopsie et change la séquence de la collecte des données correspondant à la situation d'alarme.

INTRODUCTION

The use of CAMAC systems in Hungary is related not only to nuclear applications. There are several systems working and planned in industrial data logging, chemical, psychological and other laboratory and automatic measuring systems.

One of the most interesting ones is a conventional power plant data logging and alarm system. The plant has 215 MW units, working with oil or gas heating. Every unit has a separate computer based data logging and alarm system.

1. CONNECTION TO THE POWER PLANT

The data logging system has 300 analogue inputs. These inputs get signals from different measuring devices, which measure temperature, pressure, volume, shaft vibration, carbon-dioxid contents, etc. The signals are thermocouple, thermoresistor signals or standard 20 mA transmitter signals. (Fig. 1.)

There are 480 digital on - off inputs, coming from valves, boundary switches, etc. These inputs are read in cyclically. Further 32 digital signals are connected to interrupt inputs of the system; these are alarm signals.

Several switches, push buttons and output units are placed in the control room of the power plant. Push buttons - causing interrupts - can alter the program. The operator has seven sets of decimal switches, so he can give in some parameters to the system. There are also numerical displays and analogue plotters in the control room, where the operator can get measured or calculated data. There are three output typewriters preparing different reports to the operator.

The computer and all the real time and conventional peripherals are in the computer room; so the plant operator has no possibility to stop the whole data logging system or start it from dead-start.

2. FEATURES OF THE REAL TIME PERIPHERALS

The real time peripherals are placed in 5 CAMAC crates. The analogue inputs are individually fitted with noise filters or matching impedances. The next stage is a reed relay multiplexer which switches 3 points for every input signal. A programmable amplifier and an integrating type analogue to digital converter follows the multiplexer.

The digital on - off inputs from the plant technology and the decimal switches of the control room are connected to digital input registers.

The digital inputs are dc decoupled from the data sources at the plant.

The alarm signals of the power generator and the push buttons of the operator in the control room are connected to change of state registers, causing interrupts.

The numerical display modules in the control room get binary coded decimal signals from digital output registers. There are three digital to analogue converters which control the plotters. The CAMAC crates are controlled by Al type crate controllers, which are connected to the computer via a Branch Driver.

3. COMPUTER AND CONVENTIONAL PERIPHERALS

The computer is a TPA/i type computer, produced in Hungary. The word length is 12 bits. The core memory has 16 K-words, cycle time 1.5 usec. The peripherals are the following:

- fast, fix head disc with 256 K-words capacity;
- paper tape peripherals;
- console typewriter.

The three output typewriters in the control room are connected to the standard I/O bus of the computer.

4. FUNCTIONS OF THE DATA LOGGING SYSTEM

There are three main functions of the system:

- analogue and digital signal handling;
- supplying special information to the plant operator;
- preparing reports.

The analogue inputs are measured cyclically at every 5, 20 or 120 seconds. The cycle times of the inputs are related to the different states of the plant, e.g. boiler heat up, turbine run up, etc. The computer checks validity of the measured values, checks different types of limits depending on the nature of the measurement, calculates linearity, corrects dimensions and in some cases corrects the measured value with some other measured values. The parameters of the different checks depend on the state of the power plant.

The digital inputs are read in with 1 sec cycle time. The program checks the inputs against either previous data or some prefixed parameters.

The plant operator can set the address of an analogue input or some calculated quantity on his decimal switches, and with his pushbuttons he can make changes in the measurings. With other pushbuttons he can instruct the system to display the measured and/or calculated result of the previously set address on the numerical display or in analogue form on the plotter. The operator has seven of these output devices with which he can trace some interesting trends in the change of plant parameters.

The data logging system produces several reports. If a measured and/or calculated value exceeds some preset limit, this is indicated in the limit report with some indication on the speed of the variation. The not expected digital input changes are reported in the change of state report.

The normal, routine report is produced every hour or quarter hour. This includes about 80 measured values, which are either momentary or averaged. There are about 20 calculated values, e.g. efficiency figures.

At the end of every shift (8 hours) there is a shift report which characterizes the plant in the previous shift. Every midnight a daily report is produced and the concentrated data of the previous day are punched out for further computing.

A post mortem report is written constantly and cyclically on the disc memory for some preselected data. If some malfunction occurs in the plant, the post mortem report "freezes" and can be used for further investigations. If everything is normal the post. mortem report overwrites itself about every 15 minutes.

5. SOFTWARE ORGANISATION

The operating system can handle interrupt, foreground and background tasks. The interrupt tasks are always in the core memory, while the foreground and background tasks are rolled in from the disc. Foreground tasks can be interrupted only by interrupt tasks, otherwise they continue their run, even if a higher priority foreground task wants to run. Tasks can be started either by time or by an other task.

SUMMARY AND CONCLUSION

The described data logging system has very simple functions. The main aim of this system is to get experience of the nature of the power plant and also of the logging system. The second aim is to help the operator specially in start up periods and to get continuous information on the trend of the plant variables. If the experiences are good we are planning to connect also some control devices to the system.

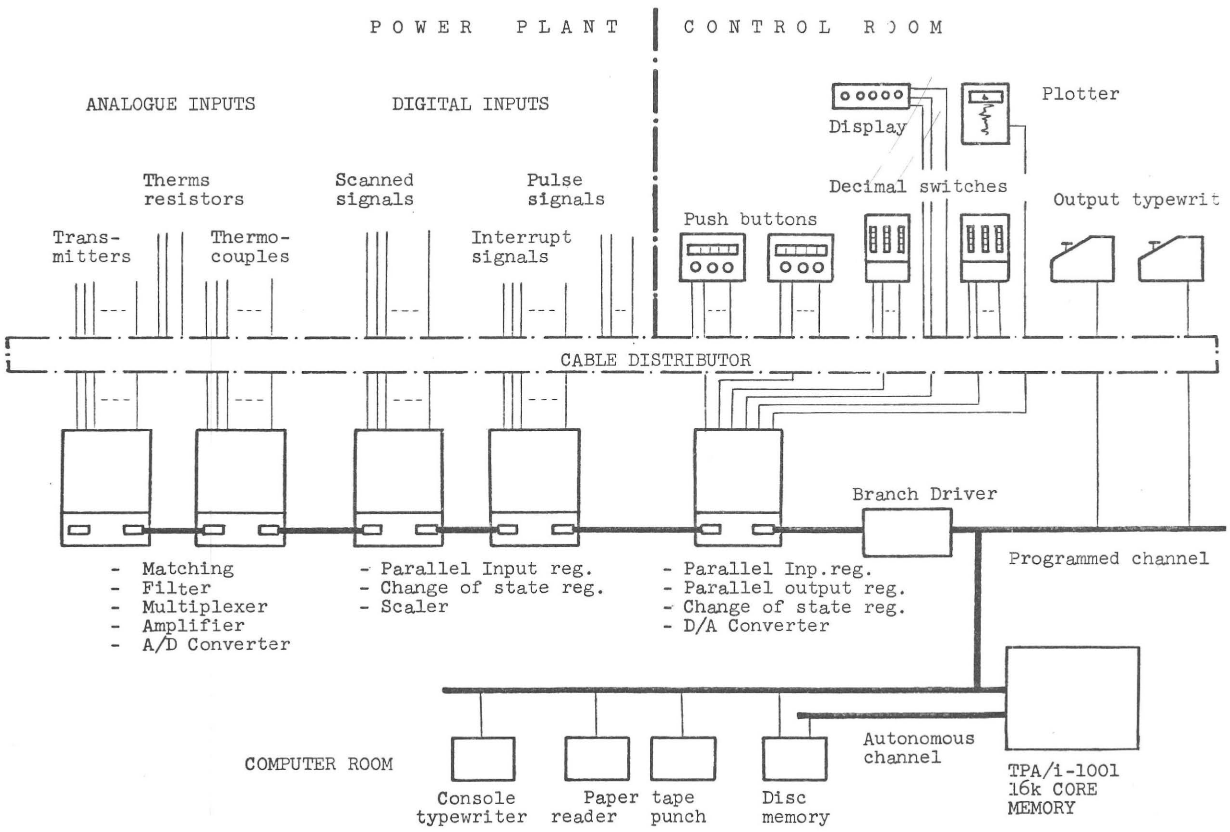


Fig. 1. Block scheme of the Data Logging System.

CONTROLE AUTOMATIQUE DES MODULES CAMAC REALISATION ET EXECUTION DE PROGRAMMES POUR LA VERIFICATION AUTOMATIQUE

J. -P. Guérin
Commissariat à l'Energie Atomique, Centre d'Etudes Nucléaires de Saclay,
Services d'Electronique de Saclay, Gif-sur-Yvette, France

RESUME

Afin de vérifier automatiquement le bon fonctionnement de modules CAMAC fabriqués industriellement, on a développé une méthodologie qui consiste principalement en l'écriture d'un "programme de test" spécifique de chaque module. Celui-ci permet le diagnostic et la localisation des pannes.

Un ensemble de sous-programmes a été mis au point afin de faciliter l'écriture des programmes eux-mêmes et de les rendre indépendants de la configuration du système CAMAC. On utilise aussi ces sous-programmes pour écrire rapidement un programme utilisant CAMAC.

ZUSAMMENFASSUNG

Um das Funktionieren von industriell hergestellten CAMAC Moduln zu überprüfen, hat man eine Methodik entwickelt,die das Erstellen eines spezifischen Testprogrammes für jeden Modul erlaubt. Dies ermöglicht die Diagnostik und Lokalisierung von Pannen.

Es wurde eine Gruppe von Nebenprogrammen aufgestellt, die das Erstellen des Hauptprogrammes erleichtern und es von der Konfiguration des CAMAC Systems unabhängig machen. Diese Nebenprogramme sind auch anwendbar für ein CAMAC Betriebsprogramm.

ABSTRACT

To test throughout industrially made CAMAC modules, we developped a method which consists mainly in writing "test programs" specific to each module. They give diagnostics of failures and their localisation.

Sub-programs give facilities in writing main program and make it independent of CAMAC configuration. These sub-programs can be also used in quickly writing programs using CAMAC.

1 - INTRODUCTION.-

Nous avons développé une méthode pour contrôler facilement les modules CAMAC fabriqués industriellement en série. Elle consiste en l'écriture d'un"programme de test" spécifique à chaque tiroir.

L'importance d'un tel outil apparaît:

- lors de l'*étude* du module : le programme de test appliqué à la maquette permet généralement d'améliorer la définition et la sécurité de fonctionnement du module,
- lors du *contrôle de fabrication* de série : il permet un gain de temps appréciable dans la recherche des défauts et la mise au point de l'appareil,
- lors du *contrôle de réception* par le client : il permet d'effectuer un contrôle rapide et complet de tous les modules d'une série,
- lors de *l'utilisation* du module : il permet à l'utilisateur de vérifier lui-même le bon fonctionnement de ces tiroirs dans son installation,
- lors de la *maintenance* : il facilite la recherche des défauts et diminue le taux d'indisponibilité.

Le système CAMAC procure une grande facilité pour vérifier un module : en envoyant chaque ordre et en vérifiant son exécution, on repère la partie ou le circuit défectueux ; on verra comment on utilise le programme partiellement conversationnel, comment on peut envoyer une suite d'ordres CAMAC(lectures, écritures ou contrôles) aboutissant à relever les défauts et comment on peut effectuer des boucles pour détecter les pannes en s'aidant au besoin d'un oscilloscope.

Ces programmes utilisent l'infrastructure standard, il ne nécessitent donc qu'une étude de software. Toutefois, pour certains tiroirs complexes, il est pratique d'utiliser des "modules répondeurs" adaptés au tiroir contrôlé.

2 - PROGRAMMES DE TEST.-

Les programmes de test sont composés de deux parties : les sous-programmes de service communs à tous les modules et le programme de test lui-même.

2.1 - Minimisation de la taille mémoire des programmes.

Les programmes de test comportent de très nombreux appels aux sous-programmes avec des configurations différentes à chaque fois. Ils prennent beaucoup de place mémoire. On a donc cherché à réduire la taille de l'appel (passage des arguments) et aussi la taille des sous-programmes. En conséquence, la durée d'exécution a été augmentée mais ceci n'a pas d'importance puisque c'est l'impression des messages qui prend le plus de temps dans l'exécution de tels programmes.

2.2 - Sous-programmes de service communs à tous les modules (1), (2).

Ils sont composés des sous-programmes d'édition de messages et d'entrée de données à l'aide de la *télétype*, d'un moniteur pour rendre le programme conversationnel et des sous-programmes nécessaires à l'exécution des commandes et ordres CAMAC.

2.2.1 Ordres CAMAC.

Les sous-programmes utilisés pour transmettre les ordres CAMAC tiennent compte de l'infrastructure CAMAC. Ils ont été écrits pour l'utilisation soit d'un châssis CAMAC avec son contrôleur relié directement au calculateur, soit de la branche CAMAC. Ils exécutent principalement les *commandes CAMAC* : écriture, lecture ou contrôle (ordre ou test). Ces commandes sont accompagnées d'arguments :

- l'adresse mémoire où se trouve l'adresse N du tiroir,
- la valeur de la sous-adresse A,
- la valeur de la fonction F,
- éventuellement, l'adresse mémoire où est rangée la donnée transmise,
- au retour, les réponses X et Q sont rangées dans deux registres de condition.

Ils exécutent aussi les *ordres* que tous les contrôleurs de châssis sont susceptibles de comprendre :

- émission de Z (ZPG)
- émission de C (CPG)
- autorisation des appels (Enable Demand=ED1)
- interdiction des appels (disable Demand=ED0)
- test de cette autorisation (TED)
- test d'une présence d'un appel (TD)
- mise à zéro de l'Inhibition (IN0)
- mise à un de l'Inhibition (IN1)
- test de cette Inhibition (TIN)

éventuellement :

- lecture de la configuration des appels (LECAPC)
- écriture dans le SNR (Registre de Multiadressage) (ECRSNR)

L'ordre multiadressé s'exécute en utilisant les commandes ci-dessus avec l'adresse N adéquate (N26 ou N28)

Après exécution de ces commandes CAMAC, le programmeur qui utilise ces sous-programmes dispose des réponses X et Q pour traitement éventuel.

pour faciliter l'écriture du programme principal, on a inclus des sous-programmes propres au coupleur : mise en entrées-sorties-simultanées (ESS) en mode arrêt ou scrutation d'adresse.

2.2.2 Edition de messages. Entrée des données à la télétype.

Ces sous-programmes sont classiques.

2.2.3 Comparaison de données de 24 bits CAMAC.

Il suffit d'une seule *instruction* pour vérifier l'identité entre la dernière donnée de 24 bits envoyée du CAMAC (ou celle qui est prévue) et la dernière donnée reçue. En cas de désaccord, un saut vers le sous-programme décrit ci-après est automatiquement réalisé.

2.2.4 Sous-programme d'édition des erreurs.

Il permet, d'après les arguments qui suivent, l'instruction d'appel, d'imprimer le numéro repère de l'erreur : ERREUR NUMERO 35, ainsi que le contenu de registres choisis de façon à renseigner le mieux possible l'opérateur qui teste le tiroir. Ainsi, on peut, ou non, faire imprimer :

- les valeurs de la dernière commande CAMAC transmises (C=...) N=... A=... F=...
- la donnée CAMAC transmise lors de la dernière écriture (qui peut aussi bien être celle qui est prévue) ENVOYEE (ou PREVUE)= =01010001 01010101 00011100
- la donnée CAMAC transmise lors de la dernière lecture R

  R lu=01010001 01010101 00011100
- le mot d'état du coupleur qui contient, entre autres, les dernières réponses Q et X.

  MOT D'ETAT=00110001

Lorsque le message d'erreur est imprimé, l'opérateur a trois possibilités :
- soit continuer le test en détail,
- soit sauter au groupe de tests suivant,
- soit utiliser le *moniteur* pour vérifier des contenus de registres à l'aide du listing, pour envoyer des commandes isolées, pour générer de nouvelles séquences ou pour faire boucler tout ou partie du programme jusqu'à ce que le défaut détecté soit localisé.

2.3 - Programme de test spécifique au tiroir à vérifier.

Ce programme déroule toutes les commandes CAMAC utiles pour le test du tiroir.

2.3.1 Ecriture du programme.

Ce programme est écrit en utilisant les les sous-programmes communs décrits sommairement ci-dessus.

Tout d'abord, on étudie les spécifications techniques et le schéma synoptique décrivant le tiroir. On dresse alors la liste de toutes les commandes et de toutes les fonctions exécutées.

Ensuite, on établit la suite la plus logique possible de ces commandes, en partant des plus simples, pour que toutes les fonctions soient exécutées par chacune de ces commandes. On vérifie ensuite directement ou indirectement que ces fonctions sont bien exécutées. On vérifie aussi que le tiroir ne réagit que s'il est adressé (N), dans une certaine mesure que les commandes qu'il reçoit n'exécutent pas autre chose que ce qui leur est imparti, et enfin, que le tiroir ne comprend pas d'autres commandes que celles qui sont prévues. Pour ce dernier cas, le test de la réponse X est jugé suffisant.

Il faut environ 1 à 2 semaines pour effectuer ce travail. Lorsque cette préparation est terminée, il suffit d'une semaine pour l'écrire dans le langage du calculateur utilisé et enfin, quelques jours pour effectuer la vérification de ce programme.

En définitive, on trouve une commande CAMAC ou une suite de commandes aboutissant à un test (positif ou négatif) ou à une comparaison de données. Dans le cas où le résultat ne concorde pas, le message d'erreur est édité.

3 - DEROULEMENT DU PROGRAMME DE TEST.

3.1 - Après avoir chargé et lancé son programme, l'opérateur indique où se trouvent les tiroirs utilisés, et quelles sont les options choisies. Parmi les options les plus courantes, on trouve l'option 'endurance' qui fera boucler le programme sur lui-même jusqu'à détection d'une éventuelle anomalie. Cette option élimine tous les tests où l'opérateur doit intervenir comme changer un inverseur de position, appuyer sur un bouton, etc.. On trouve également en option la suppression des tests longs et fastidieux qui ne sont fait que sur un échantillonnage des appareils du lot en recette.

Après avoir répondu aux questions de cette phase initiale, le test proprement dit *se déroule automatiquement*. Des messages indiquent quelle est la phase en cours. Ces messages sont supprimés en "endurance". Certains messages demandent à l'opérateur d'effectuer des actions : après leur exécution, celui-ci en prévient le programme à l'aide du clavier de la télétype et la suite du test continue de se dérouler.

A chaque anomalie détectée, un message est édité et, comme indiqué ci-dessus, l'opérateur peut :

- soit sauter à la fin de la partie du test en cours,
- soit continuer ce test,
- soit utiliser le moniteur.

Le premier cas est utilisé en contrôle de réception : telle partie de l'appareil est mauvaise.

Le deuxième cas sert à connaître tous les diagnostics d'une panne, et par recoupement, à sa localisation.

S'il n'y a pas assez d'éléments, le troisième cas sera utilisé.

3.2 - Durée d'exécution.

La durée d'exécution d'un tel programme peut varier entre quelques fractions de seconde et plusieurs dizaines de minutes ou plus : en effet, l'incrémentation d'une mémoire avec relecture et vérification ou des interventions d'opérateur fréquentes demande beaucoup de temps.

3.3 - Exemple : voir en appendice.

4 - CONCLUSION.-

L'intérêt d'une telle procédure est de ne rien laisser à l'initiative de l'opérateur pour s'assurer que toutes les fonctions prévues sont réellement vérifiées, la feuille imprimée par la télétype lors du test en faisant foi. Pour qu'un appareil soit considéré comme "bon", il ne faut aucun message indiquant une panne lorsque le test est exécuté avec les options correspondant au test complet.

De tels programmes de test, justifiés pour des modules fabriqués en série, permettent donc un gain de temps important dans les contrôles de fabrication, de réception, d'utilisation et de maintenance des modules et assurent une meilleure qualité du produit.

Le langage utilisé est aisément transposable sur différents calculateurs : nous avons utilisé :

- le MULTI 8 et le MULTI 20 (INTERTECHNIQUE),
- le T2000 de TELEMECANIQUE,
- le PDP 11 de Digital Equipment Corporation.

Ce langage permet le dialogue avec le CAMAC et par conséquent, il est utilisable lorsqu'on veut mettre au point une installation utilisant CAMAC.

Pour des modules réalisés à l'unité, nous utilisons un langage conversationnel permettant d'enchaîner plusieurs commandes CAMAC.

REFERENCES.-

(1) TESTCAM-JCM8 : Sous-programmes standards pour le test des modules CAMAC à l'aide d'un calculateur MULTI INTERTECHNIQUE.
J.P. GUERIN, R. MARIN
R-3.326, Juin 1973.

(2) TESTCAM-TIM8 - Sous-programmes standards pour le test des modules CAMAC à l'aide d'un calculateur MULTI 8 et d'une branche CAMAC.
J.P. GUERIN.

(3) Langage conversationnel CAMAC, Programme d'exécution d'une directive CAMAC pour calculateur TELEMECANIQUE T 2000.
J.P. GUERIN, J.L. BIECHE.
R-3.002, Juin 1971.

(4) CAMPACK - SAIP-SCHLUMBERGER. Octobre 1972.

EXEMPLE DE RESULTAT D'UN TEST AUTOMATIQUE.

↵ est le retour chariot
Les réponses de l'opérateur sont soulignées.

```
TEST DE LA BRANCHE CAMAC                                 )
M301?0↵                                                  )  Phase initiale :
C JCRC=4↵                                                )  entrée conversationnelle
Y A-T-IL UN REP20?0↵                                     )  de la configuration du
Y A-T-IL UN REP30?N↵                                     )  système et des options.
N DAI=7↵                                                 )
N JMT=17↵                                                )
N(GAUCHE) JREP40=11↵                                     )
CCOB EN LIGNE↵                                           )
METTRE JCRC EN LIGNE, C=4 ↵                              )
NB. PERIPH. BRANCHE(SAUF TTY,TIM8)=0↵                    )
ENDURANCE?0                                              )
TEST X?0↵                                                )
TEST ESS?N↵                                              )
CHANGER DANS JMT C9 PAR 68PF OTER R11 ET C5↵             )
TEST HOLD?N↵                                             )
TEST REPEAT?N↵                                           )
TEST COMPLET ADR. TIM8?N↵                                )
METTRE ADR. TIM8 =14↵                                    )
TEST EN COURS                                            )
OK MOT DE COMMANDE,MOTS D ETATS 1 ET 2                      Début du test.

VOYANT LTA ALLUME ? oui↵

7 INVERSEURS REP30 DECONNECTES ↵                            Nota : l'opérateur
METTRE JCRC EN LIGNE, C=4 ↵                                 exécute les ordres.
CLES DAI SUR 1 ↵
CLES DAI SUR 0 ↵
OK R/W
OK A,F
OK INVALIDATION
OK Q ET X CCOB
*ERREUR NUMERO  80                             Première détection de la panne.(X)
NAF ENVOYE          C= 4 N=27 A= 9 F=26        Indication complémentaire
?↵                                             Saut fin de ce test.

M X JCRC51
VOYANT    I    ETEINT ? oui↵

VOYANT    I    ALLUME ? oui↵
OK POUR BIST. I
OK Z,C,I,Q
OK 23 LIGNES N
OK MULTIADRESSAGE TOTAL ET PAR SNR
OK BIST. D(JCRC)
VOYANT    D    ETEINT ? oui↵
OK LIGNE D/BD
OK 23 LIGNES L
OK DEMANDE EXT.
OK INHIB DEMANDE
OK G ET L24
OK BIST. ESS,BBC
OK ESS 1 TIROIR
*ERREUR NUMERO 138                             Deuxième détection de panne (ESS)
?↵                                             suite du test.

ADRESSE= 17
*ERREUR NUMERO 141                             Troisième détection de panne (ESS)
PREVU=000000000011001000110011                 Indication complémentaire
R LU =000000000011010100110011                 Appel au moniteur M
?M

* 0C5D-
M38  1E-_ A7-_ 1E-_ A6-↵                       Vérifications complémentaires
G1B9A↵                                         Suite du test
OK RAZ SEQ. PAR F4 ET TIM8 SIMULTANES
OK LIGNE X DATAWAY
APPUYER APPEL DAI
APPUYER APPEL DAI ↵
OK INTERRUPTIONS                               Fin de la première partie du
METTRE TIM8 EN M352                            test.
```

DISTRIBUTED DATA ACQUISITION AND CONTROL

B. Chaudhri, H. Kunz
Borer Electronics AG, Solothurn, Switzerland

ABSTRACT

Extension of the CAMAC principle to simplify decentralization of data acquisition and control has led to the development of a complementary system. This new approach offers distributed installations unification possibilities. Expensive multi-core cable is eliminated by linking a practically unlimited number of modular on-site interface terminals with a simple twisted-pair cable. The outlines of this system are described with emphasis on the mesures taken to ensure an extremely low error rate.

ZUSAMMENFASSUNG

Ein zusätzlich zu den herkömmlichen CAMAC-Ausrüstungen entwickeltes System zur Datenerfassung und -ausgabe an dezentralisiert gelegenen Punkten ermöglicht deren zentrale Verarbeitung. Teure Mehrfachkabel zur Übertragung dieser Daten werden ersetzt durch ein einfaches, verdrilltes Adernpaar, welches eine nahezu unbegrenzte Anzahl von dezentralisierten, modularen Terminalen verbindet. Dieses Zusatzsystem zeichnet sich durch eine besonders tiefe Übertragungsfehlerrate aus.

RESUME

L'extension du principe CAMAC à l'acquisition et la commande décentralisées a conduit au développement d'un système complémentaire. Celui-ci représente une nouvelle solution des problèmes de raccordement d'installations éloignées. Les coûteux câbles multifilaires sont remplacés par un nombre pratiquement illimité d'interfaces modulaires locaux, reliés entre eux par un simple câble bifilaire. L'idée générale de ce système est décrite, avec une attention particulière pour les mesures assurant une excellente fiabilité de transmission.

1. INTRODUCTION

1.1 In the last few years, there has been a considerable upsurge in demand for distributed data acquisition and control using computers. CAMAC, as a modular interface, provides a highly flexible and inexpensive solution to data logging and similar types of supervisory problems over short distances. Automation applications within a limited radius can also be adequately served by CAMAC. However, data input/output points are almost invariably randomly spread and are usually distributed over large areas. Thus for the task of a widely-spread laboratory, process, supervision network or similar installation, a system is required that should be:

- inexpensive, yet able to operate over any distance and with any distribution of data input/output points by employing a very elastic transmission technique.
- an extension to the CAMAC principle, so that all major features of CAMAC as an interface are still available.

1.2 This paper describes a system based on CAMAC principles that has been evolved to meet these requirements.

2. SYSTEM REALIZATION

2.1 The basic CAMAC dataway offers system designers a rigid organisation on which to work, this is shown in Figure 1a. The dataway is usually divided into several parts as illustrated in Figure 1b., where each part of the dataway is contained in a separate crate and is effectively connected to the adjacent part by a link such as the branch highway. The dataway could, however, be further subdivided into numerous smaller segments as shown in Figure 1c. to achieve greater mobility. Taking this new concept a step further we see that the extreme ends of the dataway segments could be joined together to form a ring (Figure 1d).

2.2 In this case each segment of dataway represents a distributed remote data input/output centre hereafter called a "terminal". A single interface, to link the ring to CAMAC, and thus to the computer, could be used to perform the control function.

2.3 Data could still be transmitted, in parallel, for extremly high speed operation when the distance between the adjacent terminals is short. However, by serializing the data between each terminal, the interconnecting links can be of virtually any length since an inexpensive twisted pair cable may be used. Thus transmission over many kilo-metres would present no problem.

2.4 This concept forms the basis of Borer 3000, which is a unique and fully modular "building block" system. A typical Borer 3000 installation normally consists of a very simple ring cable interconnecting a number of remote terminals (Figure 2). These, through their modularity are readily equipped for their individual applications in the outside world.

3. TRANSMISSION TECHNIQUE

3.1 A very important feature of Borer 3000 is the method by which data is transferred. Data originating at the computer consisting of addresses, sub-addresses, information, commands, etc., are passed to the controller for transmission as a serial word around the ring cable through each terminal and back to the controller.

3.2 Each transmission is carefully monitored and compared in the controller to ensure that no fault has occured. Data originating at a terminal is transmitted twice for security. Since the routine work, such as supervising the transmission, comparisions, parity checks etc., is carried out by the controller no additional load is therefore placed on the computer. Should a terminal become defective, it switches itself out of the ring so that the rest of the system can continue to function normally.

4. BIT PATTERN

4.1 Data is transmitted in the form of an 11-bit word; serially for the more normal single twisted pair cable, although the possibility exists to transmit all 11 bits in parallel over multi-way cables for extreme speed. Of the 11 bits, eight are used for the carriage of information while the remaining three bits are used for control, interrupts, parity etc. (Figure 3).

5. TRANSMISSION SEQUENCE

5.1 If, the computer has something to say to a terminal, the appropriate terminal address is sent round the ring using eight bits (Figure 4a). The relative terminal responds by repeating its address, which can thus be compared in the controller with the information that was sent out. Any descrepancy caused by transmission error is overcome by sending the address again automatically. Once the terminal has been addressed it is ready to receive the sub-address, which defines the module in the terminal with which the computer wishes to make contact (Figure 4b). This is also repeated by the terminal and sent back to controller for comparison.

5.2 Bit 3 in the sub-address information word is used to indicate whether a Write or

Read command is intended. Following a Write instruction the terminal receives eight bits of information from the computer to memorize and repeats it back to controller for comparison (Figure 4c). If correct, an "execute" command is sent to the terminal to transfer the memorized information to the prior addressed module. Alternatively, with a "Read" command, the controller sends empty pulse trains twice which the terminal can modulate with the information that the computer requires. The controller checks the validity by comparing the two (Figure 4d).

5.3 Any terminal at any time has the chance of modulating bit 2 to indicate that it requires attention (Figure 4e). The controller will then complete the job in hand and send a pulse train that is empty all but for bit 1 (Figure 4f). The first terminal on the ring requesting an interrupt can therefore identify itself and the cycle (Figure 4a - 4d) is repeated with bit 2 set to "1". This prevents any terminal, further round the ring, requesting an interrupt until the previous interrupt has been handled. Immediately after completing the handling of one interrupt the controller sends another interrupt granted pulse train and the above mentioned sequence continues until all interrupts have been answered.

6. THE CONTROLLER

6.1 The controller acts in an organising manner as the interface between the computer and the ring cable. It is wholly responsible for the correct formation of the pulse trains on the cable, timing and error detection. Extensive flexibility in system planning and building is provided by the controller i.e. an installation can be conventionally composed of both local input/outputs interfaced directly through CAMAC as well as remote distributed input/output points via Borer 3000. Each controller can manage a ring cable having up to 256 terminals and practically any number of such controllers may be operated from a single computer, hence the potential capacity of the whole installation is virtually unlimited.

6.2 The block diagram of the controller is shown in Figure 5.

7. TERMINAL

7.1 The terminal has been designed as a very fundamental structure to which a large number of option combinations can be added. Internal interconnections are made almost exclusively through the dataway (Figure 6) which provides the rigidly defined organisation as in a CAMAC crate. Eight "Write" lines and eight "Read" lines are incorporated to accept information from, or give information to, the modules i.e. DAC/ADC, output/input gates, multiplexers, etc.

8. PERFORMANCE NOMOGRAM

8.1 The curves showing the transmission speed as a function of cable length between the terminals are shown in Figure 7. The cable length limits given represent half the actual cable length permissible between terminals without degrading the transmission rate so as to allow for the rare case where a terminal fails and thus automatically bridges over its connection into the ring. For operation in area "A" any type of twisted pair cable may be used. Operation in area "B" is also feasible if the twisted pair cable is replaced by HF cable.

9. SYSTEM CHARACTERISTICS

9.1 The outstanding features of the system can be summarized as follows:

- Flexibility: The terminals are constructed on fully modular lines so that each may be readily equipped for any application. According to application, an installation can have one or more ring cables each capable of linking more than 250 terminals.
- Economy: The basic modular units and their manner of being constructed into a system have been kept very simple so that the installed cost bears an almost linear relationship to the complexity of the application.
- Capacity: Because of its modularity, a Borer 3000 installation can be extended practically indefinitely; the only limiting factors are the performance and memory capacity of the computer.
- Bi-Directional: The system has the ability to give information and commands to the outside world (process, experiment, etc.) as well as to accept data. The system can therefore be used not only for data acquisition but also for control and regulation as well.
- Arbitrary Interrupt: A unique feature of Borer 3000 is the possibility for any terminal to request attention of the computer at any arbitrary time. The computer and transmission system are thus freed from routine enquiry cycles and can therefore be used much more efficiently.
- CAMAC compatible: The CAMAC compatibility enables Borer 3000 to be operated with virtually any computer. CCITT compatibility enables even greater flexibility.
- Mixed analogue and digital information: The system is able to handle arbitrary mixed analogue and digital information using ADC's, multiplexers, etc.
- Choice of transmission media: Borer 3000 has been conceived as a series of terminals

normally interlinked by a single twisted-pair cable. Ready adaption for partial or total use of other transmission media e.g. telephone lines, microwave links, etc. is however always possible, if necessary.

- Considerable distances and high speed: Although Borer 3000 is capable of transmission at speeds of up to 2 MBauds over short distances, the speed is usually set rather lower e.g. 150 kBauds, to obtain a good compromise between speed, distance and security.
- Multiple users: A single Borer 3000 system has the capability of being able to serve an arbitrary number of independent users.

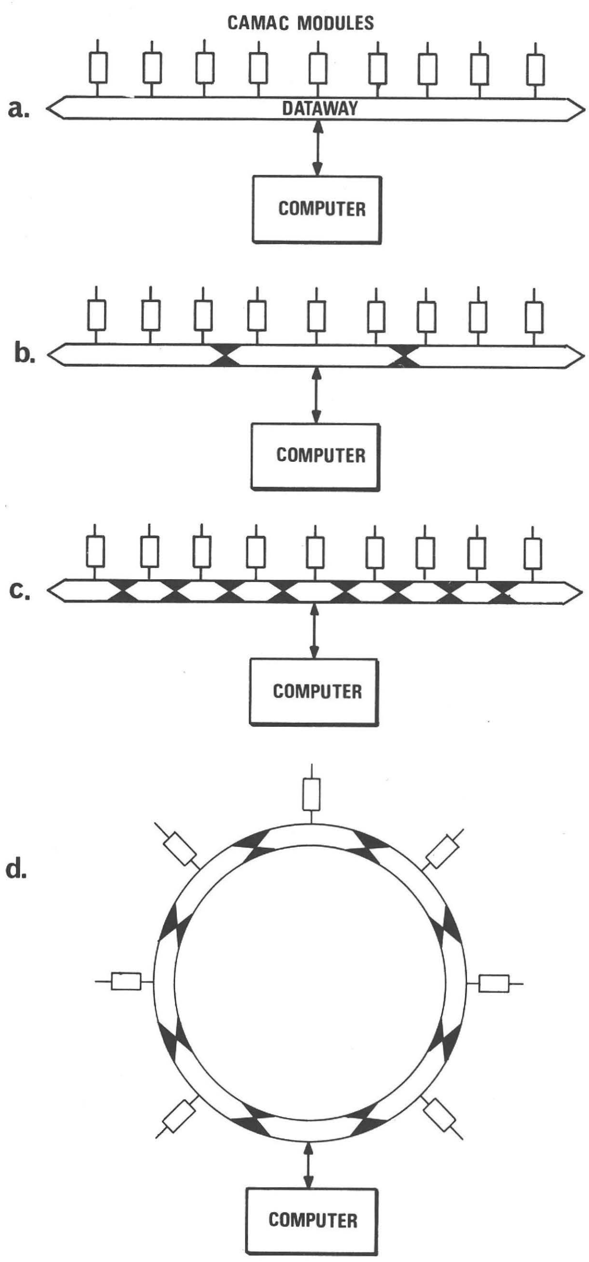


Fig.1 - Development of the distributed dataway theory

10. MAIN FEATURES

The main features of this system can be summarized as:

No of rings	unlimited
No of terminals/ring	256
No of sub-addresses/ Terminal	128
Word structure	8 bits Information 3 bits Function
Transmission speed	10 Baud....2 MBaud
Error probability	10^{-8}.....10^{-11}

The low error probability is achieved by:

1) double transmission.
2) good noise immunity due to high common mode rejection provided by differential input to the line receivers.
3) parity checks.

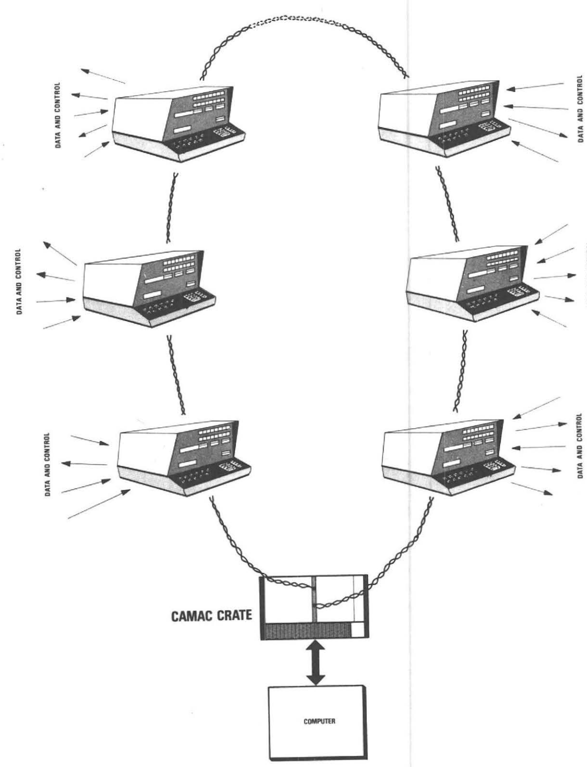


Fig.2 - Basic form of the Borer 3000 extension to CAMAC

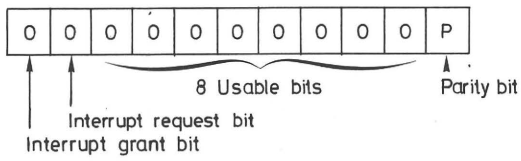


Fig.3 - Serial word structure

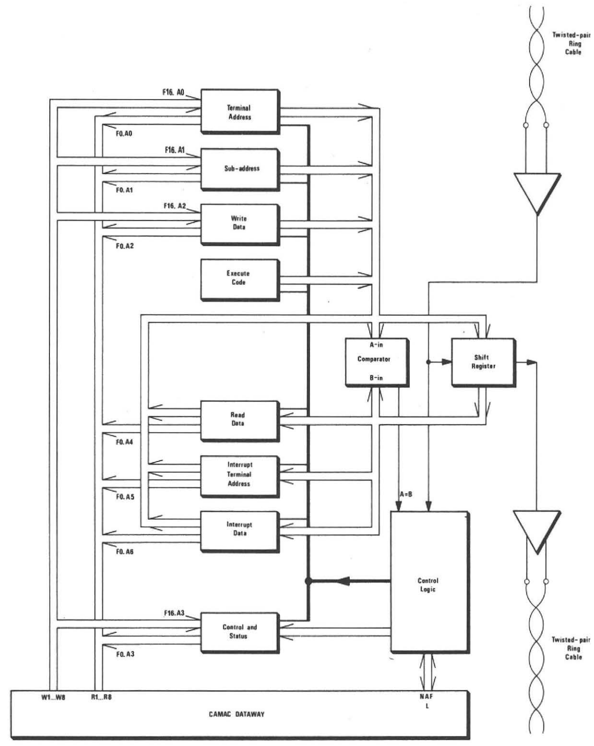


Fig.5 - The Controller, Connects the ring cable to CAMAC

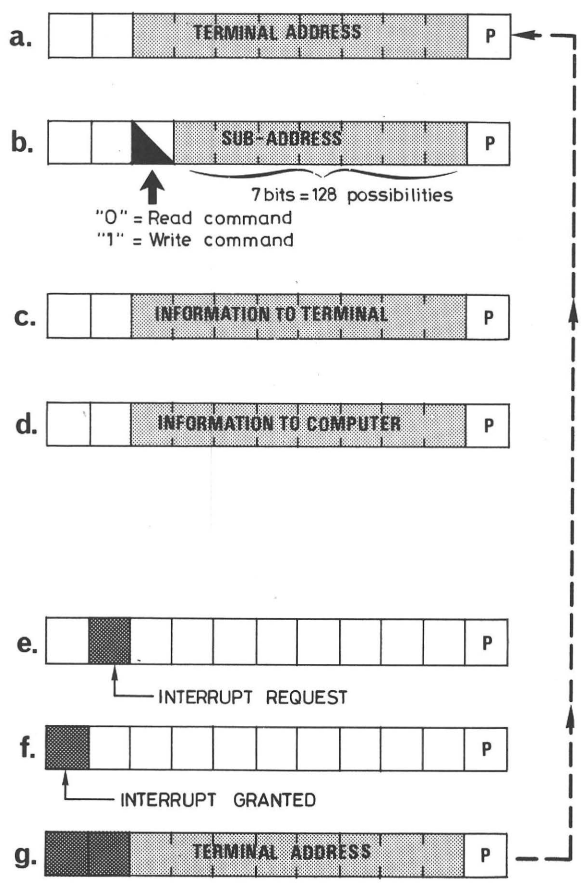


Fig.4 - Usage of the serial word

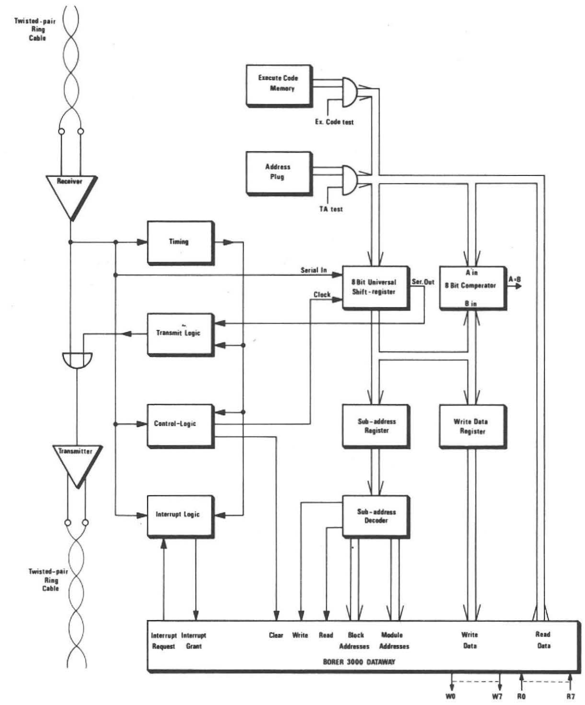


Fig.6 - The Terminal logic

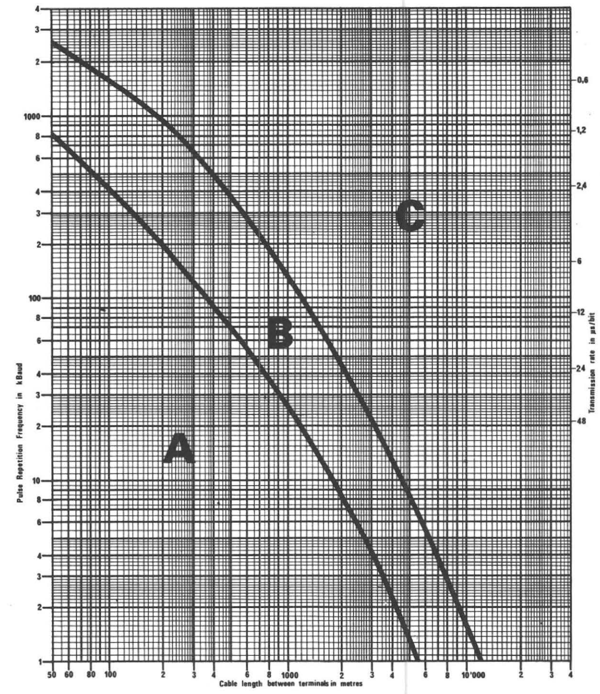


Fig.7 - Transmission rate as a function of distance between terminals

GENERAL DISCUSSION

Q - G. Huxtable

I would like to get down to a simple, basic hardware problem that I think CAMAC modules suffer from, and which might involve difficulties especially in continuous process applications. It is this, you can't safely pull a module out and replace it unless you switch off the power.

C - R. Patzelt

In the CAMAC specification (EUR 4600) it is assumed that the power is switched off before a module is taken out of the crate. Especially therefore the off-line and power-off states of the crate controller 'A' were specified with great care, so that you can switch off a crate and all the rest of the system remains uninfluenced and workable. I would like to know if the requirement that a unit must not be taken out of a running piece of equipment is only typical for CAMAC.

C - K. Hilton

For some systems I know it is possible to take out a module without problems. It may be a connector design question. Of course, it is easier to solve such a problem if you start with a clean piece of paper to design a system than if you are working to a standard.

V

AUTOMATION IN MEDIZIN UND GESUNDHEITSDIENST

AUTOMATION IN MEDICAL AND HEALTH SERVICE

AUTOMATISATION DES SERVICES MÉDICAUX

Präsident - Chairman - Président:

R.S. LARSEN, Stanford Linear Accelerator Center, Stanford, USA

PROZESSDATENVERARBEITUNG IN DER MEDIZIN

A. J. Porth
Arbeitsgruppe Labordatenverarbeitung der Medizinischen
Hochschule Hannover, Hannover, Germany

ZUSAMMENFASSUNG

Die computerunterstützten real-time-Datenverarbeitungsbereiche in der Medizin befassen sich mit der Erfassung und Weiterverarbeitung von Messdaten, die entweder direkt (Intensivpflege, Nuklearmedizin, EKG, EEG) oder indirekt (klinisches Laboratorium) vom Patienten gewonnen werden. Schnell verfügbare Daten helfen, zeitkritische Situationen zu meistern und tragen wesentlich zur erfolgreichen Diagnostik und Therapie bei.

ABSTRACT

The computer supported real-time data processing applications in medicine mainly apply to capture and use of measured data. These may be direct (intensive-care, nuclear medicine, ECG, EEG) or indirect (clinical laboratory) from patients themselves. Quickly available data help, in time-critical situations, to bring forth the essential informations and are necessary to good diagnosis and therapy.

RESUME

Les applications dans le domaine médical du traitement des données en temps réel par ordinateur concernent principalement la collecte et l'utilisation de données de mesures, fournies soit directement (soins intensifs, médecine nucléaire, ECG, EEG) soit indirectement (laboratoire clinique) par le malade. Dans les situations où le facteur temps est d'une extrême importance, la rapidité avec laquelle on peut obtenir des informations contribue à rassembler les données essentielles, ce qui est indispensable à la sûreté du diagnostic et à l'efficacité de la thérapeutique.

At this point in time medical information systems have gained a permanent and growing meaning in the physician/patient relationship. During the 60's we saw mostly conceptualisation and, in the main, there were good ideas and system designs. In the past few years technical possibilities have provided the background for some realization of these designs. Electronic data processing and allied techniques provide the necessary basis.
New branches of science have been developed. For example, computer science or, more significantly, medical information science, covering data aquisition, information management, information analysis and system monitoring in medicine (1).

A central medical information system is driven by the daily acquired data which, when possible, should be monitored for their correctness at the point of origin. This leads to the construction of largely self-supporting sub-systems that transmit properly qualified data to a central data-base. For data acquisition and communication within sub-systems and data exchange between these subsystems there are problems with regard to internal and external interfacing to be solved.

Further criteria determine the structure, field of emphasis, type and size of a subsystem, these being:
- a large amount of data may have to be acquired, processed or reduced,
- specific data structures may have to be handled,
- the desire for quick availability of results may demand a real-time mode,
- special administrative problems may need to be handled.

Within a medical information system typical examples of subsystems that can be better serviced through the rationalisation and automatisation of real-time methods are: intensive care, nuclear medicine, ECG, EEG, and the clinical laboratory.
The essential problems faced in these fields and the advantages of computerisation are explained in the following:

1. Intensive Care (2)

Intensive care deals with the monitoring of patients who are critically ill and whose condition may alter drastically over a short period of time. It is necessary to have control over measured values (such as blood pressure, heart rate, frequency of breathing, and temperature) over a large time scan. The data received from the in-vivo-measurements control the actual services performed such as instrument control, the comparison of measured values, the calculation of trends and the sending of alarm signals when necessary. As such this data need not be stored for a long period of time, all necessary information being conveyed to the doctors and nurses by means of visual display units or control panels. A considerable advantage of EDP usage is that critical situations and the violation of boundary values can be recognized and signalled much quicker, thus generating alarm signals before the actual critical situation arises.

2. Nuclear Medicine (3)

Nuclear medicine makes use of radioactive materials to assist in diagnosis and may be devided into two groups dependent on the medical examination to be carried out. The first is the study of a case over a long period of time where many values are considered, these being taken at intervals of hours (or even days). The second is the short-interval examinations in which a combination of measurements is used to determine the time-dependant development or the distribution of activity within specific organs. Since the emission of radio-active rays is a statistical phenomenon we observe scattered measurements that are subject to a statistical error. By use of EDP and implementation of mathematical methods it is possible to process large series of measured values in a short period of time. In order to improve accuracy of diagnosis it is necessary to provide good optical presentation of diagrams, allowing a choice of the area of intensity to be displayed, and determination of an interesting region for obtaining time-activity curves and intensity profiles.
The on-line data acquisition and instant processing there-of is especially advantageous for diagnostic use since it is often impossible for a quick repeat of the examination to be made because of the necessity of protection against exposure to radioactivity. The computer offers in addition great assistance in terms of documentation oft the use and stability of radio-active materials taking into account especially the natural fall-off of activity with time.

3. ECG (4,5)

Electrocardiography deals with the acquisition, by means of electrodes connected to the patient, and subsequent evaluation of graphs showing the electrical activity of the heart. Present publications show that the average time required by a doctor to evaluate a single electrocardiogram is 6 minutes. The automatic acquisition and evaluation provides a good service to the doctor. In the case of on-line data acquisition it is necessary to perform analog-to-digital conversion within the computer as the acquisition takes place, whereas off-line methods write the signals to magnetic tape in either analogue (giving a poor signal to noise ratio (S/N) of ca. 40 dB) or digital (better S/N ratio but more expensive) form.
The evaluation phase is primarily concerned with the recognition of the ECG rhythm and the elimination of

serious disturbances. This is followed by waverecognition, mostly in a 3-dimensional space, the definition of wave boundaries, and the determination of a representative wave complex. Thereupon the amplitude and time parameters are determined and the characteristic points within an ECG cycle are located. The diagnostic classification is performed on a deterministic or statistical basis. Measured values and classifications are finally documented.
The advantages of computer assistance in the automatic evaluation of the ECG are: consistent classification over repeated evaluation, considerable time-saving for the doctors and, optimal usage of the ECG apparatus.

4. EEG

Electroencephalography deals with the plotting and evaluation of the electrical signals from the brain. The methods in use today plot 8, 12 or 16 channels in which the frequencies occuring (0-30 Hz) may be divided into 4 interpretative classes:
Delta-waves (0-2.5 Hz), Theta-waves (2.5-7 Hz), Alpha-waves (7-12.5 Hz) and Beta-waves (12.5-30 Hz). The manual and visual evaluation makes use of a template and is very time-consuming. The main functions are those of: definition of the background activity and the wave-dispersion, pattern recognition, the determination of amplitude values and enumeration of zero points.

The problems of data acquisition, data evaluation and presentation of results with computer support are similar to those for electrocardiography.

5. Clinical Laboratory (6,7)

The work of the clinical laboratory is an analytical examination of corporeal liquids and waste products such as blood, liquor, plasma, urine or fecies. For the analysis there exist different types of automatic, semi-automatic or manual instruments which provide numerous measurements from a minimum sample volume. The problem of the provision of an internal interface is here especially acute due to the necessity of obtaining various combinations of measurements from a variety of instruments from numerous manufacturers and transmitting these to the computer. Each on-line measuring device has a pre-processing unit that takes care of the identification data along with the measurement data and then transmits standardised data blocks to the computer. Modularly designed data acquisition systems perform a large quantity of processing in the peripheral devices and, indeed, the laboratory computer itself uses up only a small amount of its time for real-time processing. On the other hand it is necessary to perform numerous administrative and commercial tasks such as scheduling of requests, provision of identification material and provision of supplementary data for sample analysis. Also necessary is the control of the results as regards their quality, correctness and plausibility, along with the provision of clearly arranged and easily interpreted output and also numerous statistics.

Computer assisted real-time medical data processing is concerned with the direct or indirect acquisition of patient data. The speedy evaluation of these data provides the clinician with considerable assistance regarding diagnosis and therapy. The technology required for this must primarily be reliable and safe. Measured values must not be tampered with and erroneous results must be recognised as such.
The ever growing technological advances allow the realisation of many things that were thought of as impossible only a few years ago. In order that we retain control over this technology it is necessary that we should work towards normalisation and standardisation as soon as possible.

Literatur:

(1) REICHERTZ, P.L.: Medizinische Informatik - Aufgaben, Wege und Bedeutung. IBM Nachrichten 215 (1973), S. 567-576.

(2) KUHRT, F.: Elektronik und Datenverarbeitung in der medizinischen Technik. Electromedical 1/73.

(3) FÖLBER, K.: Datenverarbeitung für die Nuklearmedizin. Aus IBM "Datenverarbeitung und Medizin" (März 1971).

(4) AMMENDE, H.P.: Datenverarbeitung in der Elektrokardiologie. Aus IBM "Datenverarbeitung und Medizin" (März 1971).

(5) ZYWIETZ, Ch.: Entwicklung von elektronischen Datenerfassungs- und diagnostischen Auswertungssystemen für die Elektrokardiographie. Medizinische Hochschule Hannover (August 1970).

(6) PORTH, A.J.: Erfordernisse und Möglichkeiten der elektronischen Datenverarbeitung im Dienste der klinischen Chemie. Z.Klin.Chem.Klin. Biochem. 10, Heft 10 (1972), S. 478-482.

(7) PORTH, A.J.: Klinische Chemie und elektronische Datenverarbeitung. Diagnostik 6/72, (1972), S.255-259.

DISCUSSION

Q - A. Fogels

Could you mention the CAMAC hardware that has been used in the work you have described? For example in EEG analysis.

A - H. Pangritz

A.J. Porth was reporting the different tasks, which might be solved using CAMAC in medicine. At the Medical Highschool in Hannover he is up to now not using CAMAC instrumentation, but a specially designed system.
But some applications with CAMAC are planned.
So, within the next two years there will be at least four different but coordinated projects for CAMAC applications in the medical field:
- Intensive care (papers of H. Schulz and E. Rehse)
- Scintigraphy for diagnosis in nuclear medicine
- In-vivo and in-vitro measurements for diagnosis in nuclear medicine (paper of H. Gahr and H. Pangritz)
- Laboratory automation

STANDARDIZING PROBLEMS IN ON-LINE AND REAL-TIME COMPUTERIZATION OF THE CLINICAL BIOCHEMISTRY LABORATORY

K. Killian, M. Knedel, H. Petsch
Hospital München Harlaching, Germany

ABSTRACT

In recent years, the large increase in medical care has produced an ever increasing demand on laboratory services. The most rational way to meet this demand for the most part is the introduction of automatic chemical instruments on-line connected to a computer for the tasks of real-time test data processing. Such a system has been developed and implemented in the laboratory of the Hospital München Harlaching with the following features which are described in detail:

- direct sample identification by precoded tubes
- automatic sample distribution
- on-line connexion of 28 laboratory instruments with the computer by so called pretreatment units containing standardized hard- and software connectors
- real-time drift-correction and indication of aberrant conditions such as abnormal drift, missing peaks and operation errors on displays directly exposed on the different working places
- multiprogramming software package with real-time variation of test data processing.

This presentation of an automatic laboratory system is intended to serve as a basis for the standardization of modular hardware and software units capable of linking laboratory instruments with computers.

ZUSAMMENFASSUNG

Mit dem Ausbau des Gesundheitswesens hat der Bedarf an Laborleistungen in den letzten Jahren ständig zugenommen. In den meisten Fällen kann den Anforderungen am rationellsten dadurch entsprochen werden, dass automatisch arbeitende chemische Instrumente on-line mit einem Rechner verbunden werden, der die Untersuchungsresultate im Echtzeitbetrieb verarbeitet. Ein solches System wurde in den Laboratorien des Krankenhauses München Harlaching entwickelt und eingesetzt. Nachfolgende Eigenschaften des Systems sind in dem Bericht ausführlich beschrieben:

- direkte Identifizierung von Proben durch Verwendung von vorcodierten Einzeluntersuchungsgefässen
- automatische Probenverteilung
- On-line-Verbindung von 28 Laborinstrumenten mit dem Rechner über sogennante Vorverarbeitungseinheiten mit standardisierten Hardware- und Software-Anschlüssen.
- Echtzeit-Driftkorrektur und Anzeige von abweichenden Zuständen wie anomaler Drift, fehlende Peaks und Betriebsfehler auf Sichtgeräten an den verschiedenen Arbeitsplätzen
- Software-Paket für Mehrprogrammbetrieb mit Veränderung der Testdatenverarbeitung unter Echtzeitbedingungen.

Dieses automatische Laborsystem soll als Basis für die Standardisierung modular aufgebauter Hardware- und Software-Systeme für den Anschluss von Laborinstrumenten an Rechner dienen.

RESUME

Au cours des dernières années, l'accroissement considérable des soins médicaux a suscité une demande sans cesse croissante en équipements de laboratoire. La procédure la plus rationnelle à utiliser pour satisfaire à la plus grande part de cette demande est d'adopter des instruments de laboratoire chimique automatiques, connectés en ligne à un ordinateur, pour le traitement des données expérimentales en temps réel. Un tel système a été mis au point et utilisé dans le laboratoire de l'hôpital de Munich Harlaching; les caractéristiques de ce système énumérées ci-dessous font l'objet d'une analyse détaillée:

- identification directe des échantillons à l'aide de tubes codés au préalable
- distribution automatique des échantillons
- connexion à l'ordinateur en ligne de 28 instruments de laboratoire, à l'aide d'unités de prétraitement contenant un système cablé et un système programmé
- correction de la dérive en temps réel et indication des conditions anormales, par ex. dérive anormale, pics manquants et erreurs de fonctionnement par affichage direct, aux divers points de travail
- multiprogrammation (software package) avec variation en temps réel du traitement des données testées.

Cette présentation d'un système automatique destiné aux laboratoires est destinée à servir de base à la normalisation d'unités modulaires de hardware et de software capables de relier les instruments de laboratoire aux ordinateurs.

1 The Data Acquisition System

The demand for laboratory work especially in the sectors of preventive medicine and intensive care on the one hand and the advances made in the field of quantitative examination methods on the other hand have increased the daily examination requirement in bio-chemical laboratories. In the last five years the number of tests in our laboratory has an exponential ascent. Table I

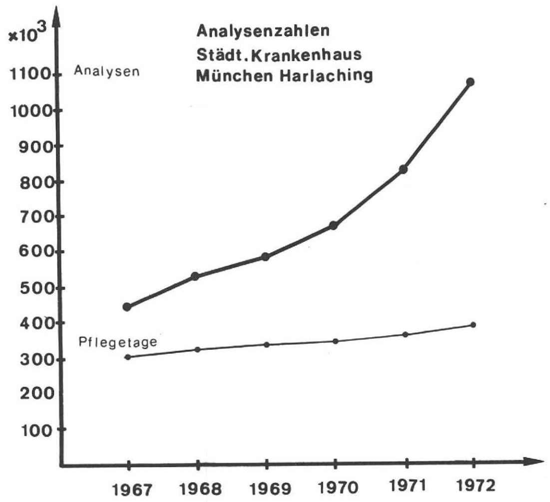


Table 1 : Number of Laboratory Tests

This trend can not mastered by the given structure and organizational form of existing laboratories. These tasks can only be carried out economically and efficiently by the use of an electronic data processing system. The fundamential problem in the realization of such systems is the on-line connexion of a great number of different automatic, semi-automatic or manual analysing equipments to the computer. In addition, the various commercially available analytical instruments have to some extend no on-line connectors or provide measured figures and curves on different analog or digital outputs. Even the output voltage the timing requirements for the signals and the connectors are of different kind.

To solve those problems the first automatic laboratory systems in Germany in the University of Tübingen and Erlangen used computers with digital and analog in-output controlers (IBM 1800, SIEMENS 305). In consequence of this solution all command, strobe and information signals had to be iniciated by extensive soft-ware modules, instead of using the available capacity of the core memory for the calculation and presentation of results.

The new systems in the University of Hannover and in our Hospital in München Harlaching use pre-processing units between the computer and the analytical instruments.

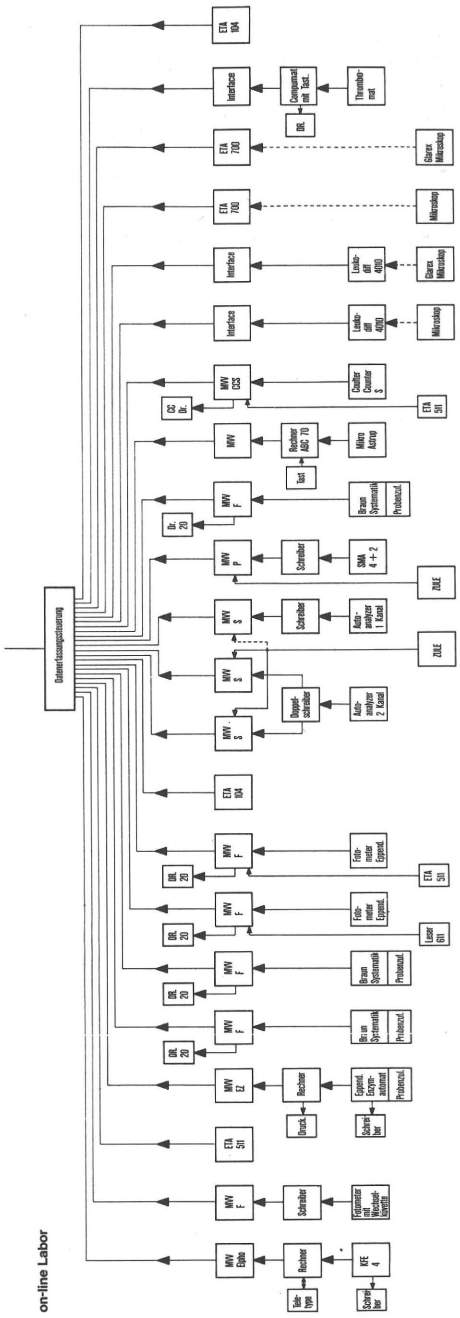


Fig. 1 : On-line Data Acquisition System

These units carry out various tasks of which only the most important are listet here:

- Take up of the electrically presented measured values and transformation the same in TTL signals
- Processing of the measured values, for example peak recognition, logarithmation or multiplication by a factor direct on the working place. Fig. 2 shows such a preprocessing unit for photometers (SILAB-System SIEMENS). The assistant can select the different measurement forms by pressing the corresponding key.

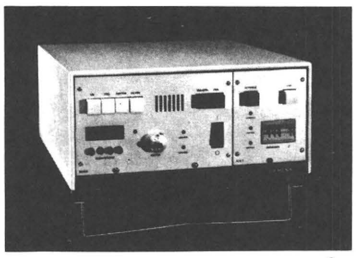


Fig. 2 : Pre-processing Unit

- Limit value and error checks with appropriate display on the operational panel without using the computer for such simple works.
- Output of the so pre-processed measured data on a numerical printer so that the measurement equipment can work without computer aid.
- Acquisition of patient identification and examination numbers by reading the coded examination vessels.
- Presentation of measurement errors detected by the real-time programme system on displays direct on the different working places.

This idea to convert the measured data, which are available in varying forms, into a form suitable for further processing in the computer by modular units performing independent functions is equivalent the root of the CAMAC-System. Unfortunately the commercially available systems have however no standardised connectors to ensure compatibility between units from different sources of design and production. The single solution to combine such existing systems is to-day the development of a CAMAC plug-in unit for each multiplexer on which the so called pre-processing units of the one respectively the other system are connected. With that however an "interface chain" is performed.
The possibility to use conventional CAMAC-connectors directly on the different laboratory instruments had till now to be rejected because of the following arguments:

- The CAMAC-System is too expensive for the automation of clinical-chemistry laboratories since most of the analysing equipments used in the daily routine work have not such an expended structure as nuclear instruments. The costs for the interface may not account the costs for the instrument itself with regard to the economical carrying out of the most frequent routine examinations.
- The data communication between clinical chemistry instruments and the computer is normally very simple. In many cases a BCD-output is complete sufficient.
- Nearly in all laboratories the analysing equipments are established on several places, in special circumstances even in different buildings, so that on one hand a too great many of CAMAC-creates with create controlers must be used or on the other hand too long cables between the analytical instrument and the plug in unit requires special key-boards on the working places for the handling of the data acquisition system.
- The fundamental principle to supply the working place in real time with all informations, necessary to handle the system in the right way demands an appropriate display directly on the operating panel of the instrument or the controler next to it.
- A direct identification system has to guarantee the reliable assignment of the patient number to the examination material, so that each result is presented to the computer with all necessary information for the treatment by the programme system.

1, 2	Messplatznummer
3, 4, 5	Verfahrensnummer
6, 7, 8, 9, 10, 11	Patientennummer
12	Prüfziffer
13	Messart / Messbereich
14	Verdünnungsstufe
15, 16	Zustandsanzeigen
17, 18, 19, 20, 21	Meßwert incl. Punkt
22	Max 4x5 Byte für Messwerte
37	Paryty des Satzes
38	Endezeichen

Table 2 : Laboratory Result Set

The problem in this case is to compose the identification number sensed by mecanical or optical readers on the beginning of examination-cycle and the result including the incubation time. A solution using two separate input channels is in this case not satisfactory. A interruption on one channel producing a shifting between the identification number and the result cannot or can only in some special cases detected by the programme system.

The standardization of the efficient communication of clinical chemistry instruments with a computer shown in fig 3 cannot begin on the photometer or the automatic analysers itself.

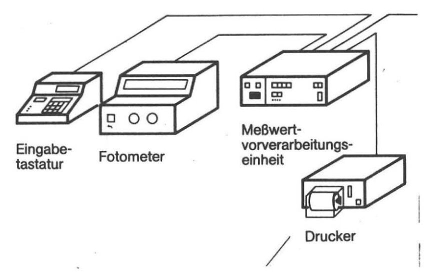


Fig. 3 : Pre-processing Unit and Key-board and Printer for Photometers

Those instruments have to contain the identification reader, the display controler and the pre-processing unit. Modules which can developed by the manufacturer of the chemical instrument or by other electronic manufacturers.

In this case to minimize the cost we propose a reduced CAMAC-version with a connector to the computer of a 8 bit (one byte) read-write bus-line for data transmission over more than 5 km and free potential as it is requested for instruments used in hospitals.

2 The Identification System

While the data acquisition system guarantees the conversion and control of the measured data the identification system is indispensable for the reliable assignment of the patient number to the examination material right from the ward to the measurement itself.

The flow of information in this system begins in the patient-reception of the hospital fig. 4

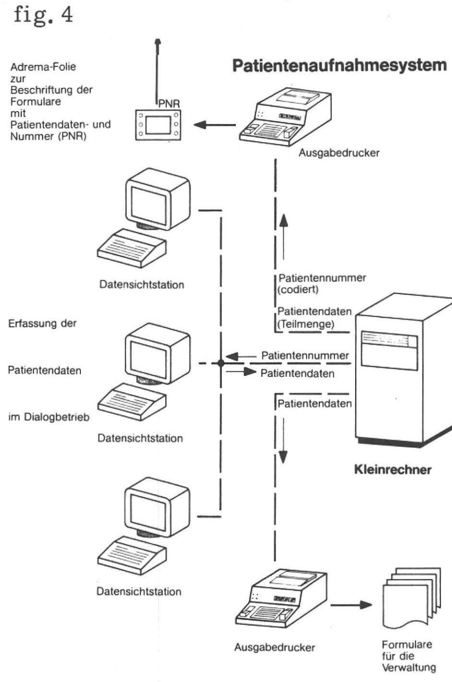


Fig. 4 : Patient Reception System

with the requisition of the patient data with aid of displays connected to a satellite-computer. This system represents an autonomous module connectable also to the CAMAC-input of the hospital information computer system. In this first step a visuell and computer legible identification adressette is printed, to accompany the patient. With aid of this foil, the requisition form and the specimen (in the full automatic case) is identified on the wards.

In the laboratory fig. 5 the fact that the coded identification of the material under examination can be read by machine, makes it possible to automate the frequently-occurring routine work. This begins by the use of manual, semi automatic or automatic sample distributors which fill with a selectable amount of serum the individual examination vessels fig. 6 sorts them and passes them out.

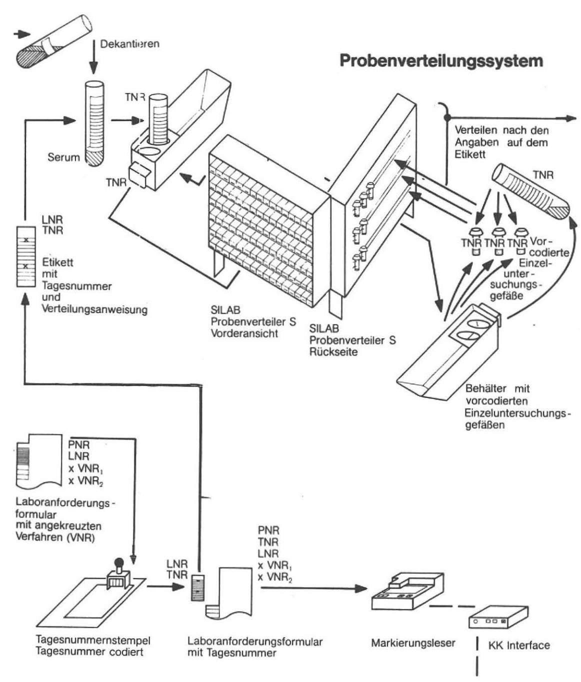


Fig. 5 : Distribution System

In the manual version fig. 5 the requisition form is read by an optical reader and the specimen-provided with a daily number - used to distribute the serum with help of a distribution-board. On the front of this board exist 400 drawers filled with precoded SILAB-vessels. The assistant takes out that drawer with the number marked on the label of the primary specimen and distributes the material in the vessels precoded with the same number. The so filled vessels are then distributed in SILAB-chains destined for the different laboratory instruments fixed on the rear side of the distribution board.

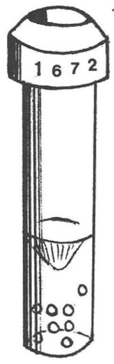


Fig. 6 : Pre-coded Vessel (SIEMENS)

In the semi automatic distribution systems the required vessels are provided automatically with the patient number in accordance with the identification on the requisition form or the label on the primary specimen.

In the full automatic distribution units a small computer controls the different automatic steps coding, dosing, distribution and filling of the examination vessels. This autonomous working module should also be supplied with a CAMAC-standardized connector so that in the data acquisition system of the laboratory even the requisition and distribution data have the same structure as the measurement results.
At the measuring equipments, mecanical or optical readers fig. 7 sense the patient identification number of the vessels so that together with the measurement data the data acquisition system obtains a completely identified result for further processing.

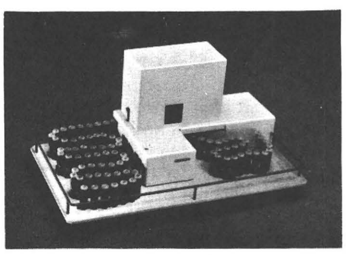


Fig. 7 : Patient Identification Reader

3 The Programme System

As far as the further processing of the laboratory data in the computer is concerned, we are, with the present state of biochemical and medical requirement at the beginning of the possibilities which data processing open up to us in this field.

Some of the initial problems in this aspect have however been resolved in our system in München Harlaching, working since half a year in practice in the clinical-chemistry, haematological, enzymatic laboratories:

- on-line data acquisition with real-time error checks and display
- determination of calibration curves
- precision checking
- plausibility control
- print out of result forms with trend control
- pattern recognition of result combinations

The modular structure of the programme system with several programmes simultaneous in overlay execution can be divided into 8 fundamental parts:

- the data acquisition programme
- the measurement processing drivers for each laboratory instrument
- the programme for the transmission of results in the patient file
- the error output routines
- the result presentation programmes
- the pattern and trend control programme
- the control programme for the real-time variation of the information management in the whole system.

A series of further checking possibilities also with reference to the function of the equipments and of the chemical processes as well as further correlation conditions are our future problems. Prior condition for the solution of such problems is however the standardization of the discussed laboratory in-output systems and equipments

DISCUSSION

Q - B. Evans

You mentioned a proposed reduced CAMAC version with an 8 bit read-write bus line. I believe a proposal for a similar bus has been made in IEC/TC66/WG3. Can you tell me if this is the same interface as you propose.
Can anyone else comment on the TC66 proposal?

A - K. Killian

The connecting specification presented in the paper is quite different from the bus line which has been proposed to the IEC Committee.
The SL-system has no bus line. Each instrument is connected directly to a multiplexer.
In the German association of medical documentation and statistics, section III, we have discussed the IEC-Specification and we have found some critical points:
- the maximum distance between the instrument and the computer is too small. A distance of 1000 m is necessary.
- the number of instruments to be connected with the system is limited to 16.
In our laboratories we have many more measurement places.
- in the IEC-system not more than the half of the input units can be in off-power state.
In the existing laboratories these restrictions are not acceptable.

A - H. Bisby

The reference by previous speakers to the 8-bit bus system as an IEC standard is incorrect. The status is that it is a Hewlett-Packard system that has been proposed for consideration by a working group of the IEC Technical Committee 66, and it is still in the proposal stage.

Q - P. Quick

How will the CAMAC-system work together with the proposed IEC byte-parallel, word-serial bus system.

A - H. Trebst

1) There will be no difficulties designing a CAMAC module driving the HP/IEC-bus.
2) Information on the CAMAC Serial Highway was presented to the German group of IEC/TC66/WG3 some weeks ago and was favorably received, but no decisions have been taken on this subject up to now.

DATA ACQUISITION WITH A CAMAC-BASED SUBSYSTEM IN CLINICAL APPLICATIONS

P. M. Czaikowski[x], D. Reimer[xx], H. J. Schulz[x]
[x]AEG-Telefunken, Konstanz, Germany
[xx]Dornier GmbH, Friedrichshafen, Germany

ABSTRACT

This report introduces a CAMAC system, which was designed as a subsystem of a centralizing computer for use in clinical data processing. For this application, autonomy of the system and radial wiring of the crates (hierarchical structure) were added to the usual specifications of a CAMAC system. In our opinion, this does not represent a violation of standardization but a useful extension.

ZUSAMMENFASSUNG

Dieser Beitrag beschreibt ein CAMAC-System, welches als Untersystem für einen zentralen Rechner zur Verarbeitung klinischer Daten konzipiert wurde. Für diesen Anwendungszweck wurden die üblichen Spezifikationen eines CAMAC-Systems in der Weise ergänzt, dass eine Systemautonomie möglich ist und dass die CAMAC-Überrahmen sternförmig an die Systemsteuerung angeschlossen werden. Es besteht die Ansicht, dass dabei nicht von den CAMAC-Regeln abgewichen wird, sondern dass diese nur sinnvoll erweitert werden.

RESUME

Nous avons présenté, dans un exposé précédent, un système CAMAC conçu pour être utilisé comme sous-système d'un calculateur pour le contrôle de processus centralisé qui servira au traitement des données médicales. Pour cette utilisation, l'autonomie du système et le câblage en étoile des châssis (structure hiérarchique) sont mis en parallèle avec les spécifications courantes d'un système CAMAC, ce qui à notre avis, ne s'oppose pas à la normalisation mais permet au contraire un élargissement utile.

1. Requirements for a Medical Processing Periphery System

Clinical data processing systems are, to an increasing degree, realized with computer hierarchies permitting a distribution of tasks among largely independent units and computers (fig. 1):

1st stage: medical measuring instruments

2nd stage: measuring computers or subsystems for limited tasks of data acquisition and data preprocessing

3rd stage: centralizing processing computer for more sophisticated data processing, for data integration and storage, for communication with the medical personnel

4th stage: administration computer

The early development state of the clinical instrumentation technique and the subsequent data processing leads to insecurity and difficulties in connecting the units (stage 1 - 3). Based on the few existing experiences it is expected that conception and realization of the data acquisition will undergo considerable changes.

It is therefore recommended to select, between the first and the third stage of the described hierarchy, a flexible solution of the instrumental connection that permits future changes without expensive development costs.

2. Advantages of CAMAC

The mentioned requirements can be fullfilled with CAMAC. The advantages of CAMAC mainly consist of two items:

- the modular system structure permitting a step-by-step extension of the system (1st stage = plug-in modules in the crate, 2nd stage = interconnected crates);
- the transformation characteristics of the CAMAC system regarding the interfaces.

The CAMAC system allows to transmit flexibly structured input data to the standardized CAMAC dataway. The degree of flexibility is given by the large number of different plug-in modules offered by the market.

Due to standardization (CAMAC dataway or branch highway) the transferability of the computer interface is guaranteed.

3. Disadvantages of Present CAMAC Systems

The advantages are opposed by decisive disadvantages with respect to the application in clinical data processing:

- The CAMAC system with a normal system controller does not possess enough intelligence and facilities for intermediate information storage. Thus the controlling computer is charged by too many individual instructions and data. The transmission of information between subsystem and central computer is furthermore scattered with respect to time, i.e. uneconomically.
- For medical data processing the missing autonomy of the peripheral system is, besides the high computer load, disadvantageous in itself. An emergency operation of the devices is hardly possible without this autonomy in case of a computer failure.
- The present bus conception of the branch highway encounters wiring difficulties due to the room conditions of a hospital. For the installation of the devices to be connected, e.g. in laboratories in different rooms, the radial wiring of individual crates appears more suitable. With such a structure failures on the transfer line would not paralyse the entire system.

4. CAMAC-DVM System

It is the objective of the developments carried out by Dornier and AEG-Telefunken to eliminate these disadvantages and thus to make the system better suited for medical data processing.

4.1. Hardware

The modified system structure plans a data processor in a master crate for central control. Slave crates are radially connected to the master crate via a serial transfer line. The central computer is coupled to the dataway of the master crate via an interface card like any other unit (fig. 2).

The master crate with the parts:

crate,

interface unit computer/dataway,

data processor with PROM and RAM storage,

interface unit dataway/RAM,

teletype interface and teletype,

time generator, chronometer

is defined as CAMAC body covering all essential functions:

coupling of the computer,

autonomous operation of a CAMAC crate with up to 17 plug-in modules.

A teletype has to be connected to the master crate in order to control the data processor in case of a computer failure. The interface of the teletype connection is the CAMAC dataway. Time generator and chronometer serve for time interrogation in real-time programming.

The CAMAC body can be extended by connection of slave crates. For this purpose interface units are provided for a serial transfer line in the master crate and in the slave crates.

The interface unit in the master crate acts like a normal CAMAC plug-in module such that, with a view to the data processor, the CAMAC system seems to consist of one crate. The alarm pattern of the slave crates is permanently available in the appropriate adaptor module of the master crate. Thus, a lot of time can be saved in real-time operation as for scanning the alarm pattern searching the module with the highest priority only transfers on the dataway of the master crate are necessary. The modules in the slave crates can be attained via the interface unit with two addressing commands. The interface unit in the slave crates cooperates with a normal crate controller CCA.

The transfer speed on the serial transfer line amounts to approx. 50 μ sec/word at a distance of about 200 m.

4.2. Software (fig. 3)

The programs for the operation of the CAMAC-DVM system are on PROM and RAM of the data processor:

- fixed basic programs that are required for every system on PROM,
- variable information (lists, data) and the special programs for the respective system on RAM.

PROM

The basic programs comprise above all the operating system (alarm handling and distributor): Alarms get through to the processor by interrupt or setting of a flag. The reason for these alarms is the request for correspondence of the modules connected to the system. This includes all modules linked via the dataway, i.e. the real CAMAC plug-in modules as well as the connected computer. The alarms are signalized to the distributor (Regieverteiler) which, according to the given priorities, activates the corresponding software module. Priorities are fixed for the complete crate as well as for the individual modules in the crate. For instance: the highest priority level of crate 3 will be served by the distributor earlier than the second level of crate 2. In case of particularly long programs a return to the distributor can be provided in the program, and the program can be interrupted, if necessary.

Load Program for RAM

The RAM can be loaded by the connected computer and by the teletypewriter (especially in case of computer failure). The interface for any transfer is the CAMAC dataway.

The basic programs comprise furthermore drivers for modules of the CAMAC body.

RAM

Data, dynamic lists, and user programs on the RAM are application-specific.

All information can be overwritten at any time. This, of course, applies to the data storage and the dynamic lists during program execution. It is also possible to load the program memory anew during operation such that other user program systems are available. This dynamic reloading is necessary for extensive tasks, in particular, and reduces the limitation by the data processor capacity.

5. Example of Application

It is planned to employ the described system for data processing tasks in intensive care, the clinical chemical laboratory, and the nuclear medicine. The system configuration for the intensive care station is shown in fig. 4. In this configuration, the test of the CAMAC-DVM system will be started this year.

It is the objective of this plan to feed the central computer TR 86 with data of clinical measuring instruments for display, storage, and subsequent processing under real-time conditions. For this purpose the devices are connected at the patients' beds by groups to the slave crates , and after simple preprocessing (e.g. analogue-digital conversion) in the slave crates and intermediate storage in the master crate the data are transferred to the computer.

Vice versa, data processed by the computer may be signalized at the beds or centrally by the system.

The advantage of CAMAC inherent in this application consists above all in the flexibility of the device connection . Specifications of the output data vary to a high extent in the field of medical devices such that a connection of all devices to an unflexible interface would require high adaption costs.

As a data processing system for the intensive care has to fulfil a great number of communication tasks (information, data storage) besides data acquisition it is mandatory to discharge the central computer of individual data acquisition tasks as performed by the described subsystem.

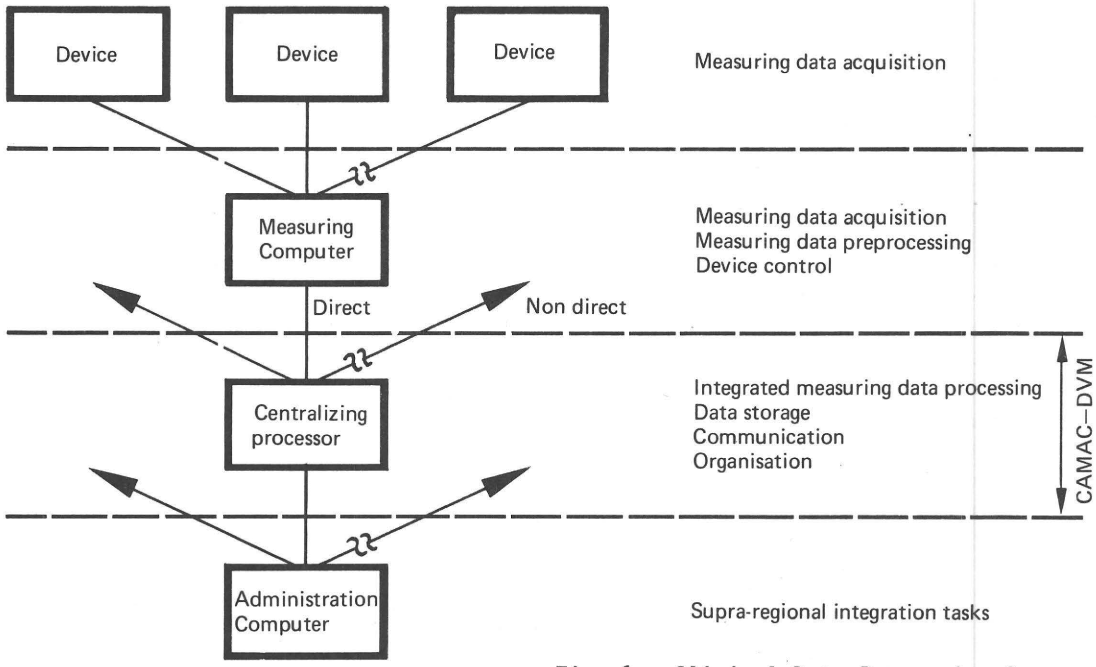


Fig. 1: Clinical Data Processing System

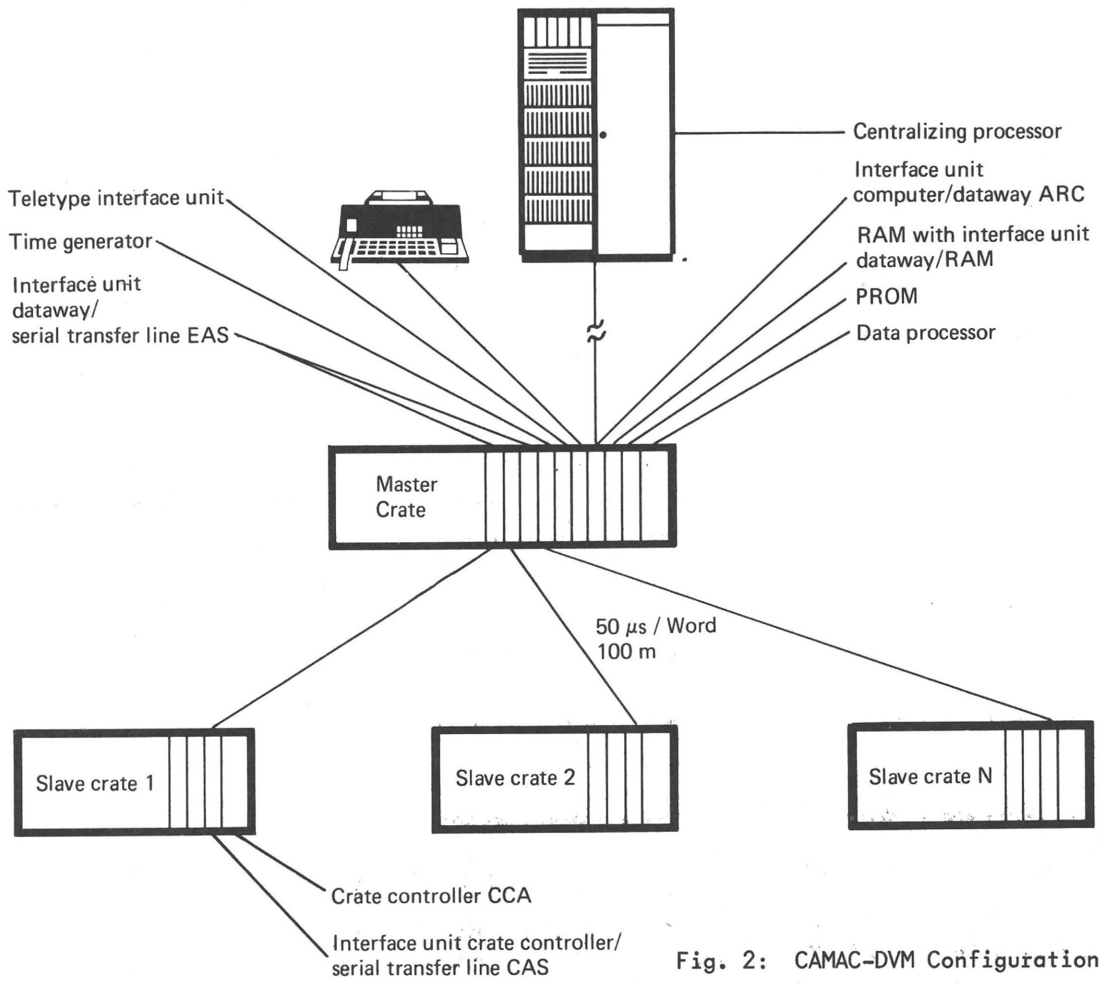


Fig. 2: CAMAC-DVM Configuration

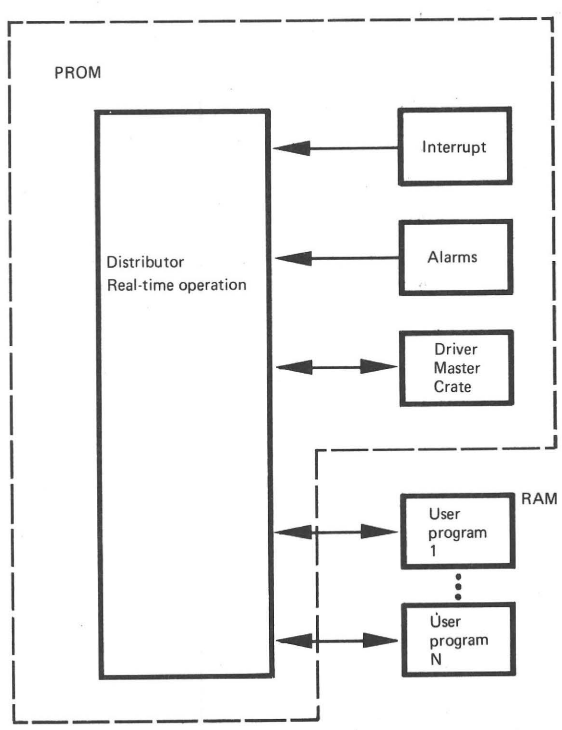


Fig. 3: CAMAC-DVM Software

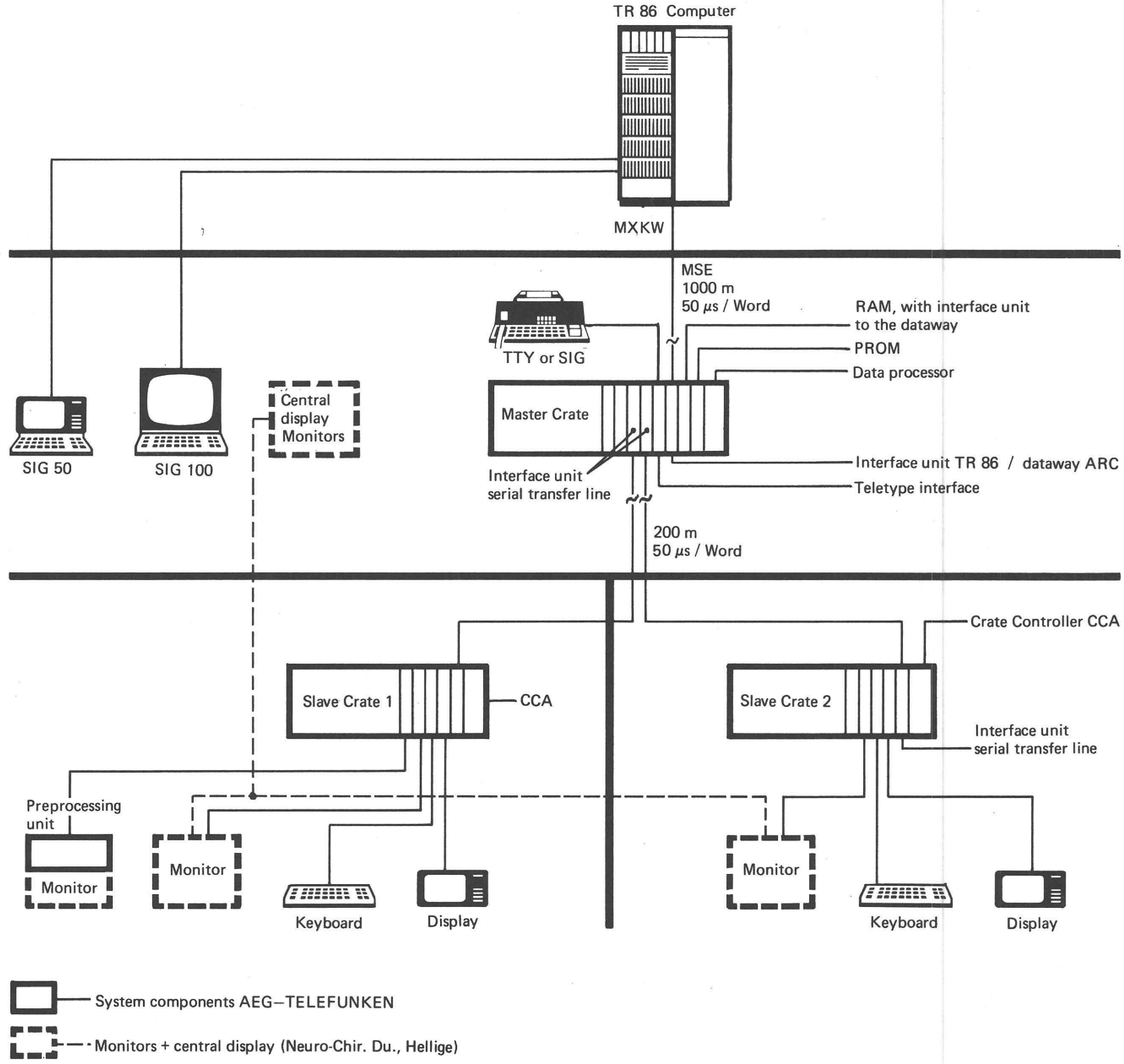


Fig. 4: System Configuration Intensive Care

CAMAC IN DER INTENSIVÜBERWACHUNG

E. Rehse, K. H. Müller
Neurochirurgische Klinik der Universität Düsseldorf
Düsseldorf, Germany

ZUSAMMENFASSUNG

Ausgehend von der heutigen Situation auf dem Gebiet der Instrumentierung für die rechnergestützte Intensivüberwachung werden die Möglichkeiten, die das CAMAC-System zur Verbesserung bietet, aufgezeigt. Es wird besonders auf die Möglichkeiten und Vorteile einer Auslegung der patientennahen Monitorelektronik als echter rechnergesteuerter Prozessperipherie hingewiesen. Eine erste CAMAC-Versuchskonfiguration wird kurz beschrieben.

ABSTRACT

Based on the present day situation in the field of instrumentation for computer aided intensive care the potential of the CAMAC system for improvement is outlined. In particular the possibilities and advantages of computer controlled peripheral bedside equipment are discussed. The initial stage of a CAMAC-monitoring system is briefly described.

RESUME

En se basant sur la situation actuelle dans le domaine de l'instrumentation pour les soins intensifs assistés par calculateur, on esquisse les possibilités d'amélioration offertes par le système CAMAC. En particulier, on expose les possibilités et les avantages des équipements périphériques situés à proximité du lit des malades et commandés par calculateur. On décrit brièvement la première version d'un système de monitoring CAMAC.

1. Computer finden in zunehmendem Maße Anwendung in der Medizin. Die Rechner werden bis jetzt hauptsächlich in der Verwaltung und Krankenhausorganisation eingesetzt. Im unmittelbar ärztlichen Bereich bei Diagnose, Therapie und Intensivpflege findet eine Rechneranwendung noch nicht auf breiterer Basis statt.
Die EDV in der Intensivpflege hat im wesentlichen vier Arbeitsbereiche:

a) Verwaltungsaufgaben, Haltung von Patientendateien, Medikationslisten, Weckerdienste für diagnostische oder therapeutische Maßnahmen etc.

b) Erfassung physiologischer Daten, z.B. EKG, EEG, Blutdrücke etc.

c) Aufbereitung und Präsentation von Daten.

d) Steuerung diagnostischer und therapeutischer Maßnahmen.

Auf dem Arbeitsgebiet der Rechneranwendung in der Intensivüberwachung Schwerkranker arbeiten seit einigen Jahren eine Reihe von Gruppen in USA und Europa. Hier werden die Rechner bis jetzt nahezu ausschließlich in den Bereichen a)-c), also Auskunft, Datenerfassung und Analyse, Datenrepresentation, benutzt. Während für den Bereich a), Verwaltungsaufgaben-Auskunftssysteme, ohne größere Schwierigkeiten Lösungen aus dem kaufmännischen und verwaltungstechnischen Bereich übernommen werden können, sind für die übrigen Bereiche die Lösungen erst zu schaffen. Die große Mehrheit der Probleme bei Signalerfassung, Aufbereitung, Präsentation (Display) und z.T. auch bei der Steuerung liegt im technischen Bereich. Einigen Arbeitsgruppen ist es bisher gelungen, Teillösungen oder Lösungsvorschläge für einige Probleme zu erarbeiten. Diese Lösungen sind jedoch weitgehend auf die jeweilige Situation, Rechnerkonfiguration und Kliniksorganisation abgestimmt. Darüber hinaus sind die Systeme stark abhängig von Güte und Umfang der verfügbaren Hardware- und Softwaremannschaften.
Der Hauptnachteil der vorgestellten Lösungen ist, daß sie alle nicht ohne weiteres übertragbar und unter anderen klinischen Bedingungen oft nicht funktionsfähig sind.
Diese Situation führt auf dem Gebiet der rechnerunterstützten Intensivüberwachung Schwerkranker dazu, daß es für verschiedene Teams nahezu unmöglich ist wirklich eng und fruchtbar an der Entwicklung von Hard- und Software zusammenzuarbeiten; es sei denn, verschiedene Arbeitsgruppen haben sich vor Beginn der Arbeiten auf identische Systeme festgelegt (MIRU).

2. Es ist nun zu untersuchen, inwieweit sich die technischen und auch organisatorischen Probleme durch die Verwendung des CAMAC-Systems lösen bzw. vereinfachen lassen.

2.1 Erfassung physiologischer Daten.
Bevor physiologische Größen einem datenverarbeitenden System zugeführt werden können, müssen sie erst in elektrische Signale gewandelt, verstärkt, gefiltert und digitalisiert werden. Wenn man die auf dem Markt befindlichen Monitore analysiert, kommt man leider zu der Einsicht, daß kaum ein Gerät diese Aufgaben erfüllt und praktisch kein Gerät systemgeeignet ist. Die heutigen Monitore sind sehr bedienungsintensiv, die elektrischen Eigenschaften lassen hinsichtlich der Langzeitstabilität zu wünschen übrig, die Ausgänge bzw. Anzeigen sind fast ausschließlich analog, die elektrischen und mechanischen "Formate" schwanken zwischen den einzelnen Herstellern erheblich.
Sollen solche Monitore an ein datenverarbeitendes System angeschlossen werden, so wird diese Aufgabe durch Einsatz des CAMAC-Systems ganz wesentlich erleichtert, einfach weil bereits eine breite Palette von CAMAC-Modulen kommerziell erhältlich ist, die hier eingesetzt werden können: Analoge Multiplexer, AD-Converter, Timer, evtl. Pufferspeicher etc.
Auch ohne Rechnereinsatz könnte CAMAC als Organisationsrahmen für zentralisierte Meßwert- und Alarmanzeigen dienen und damit den Nutzwert vorhandener Anlagen erhöhen.
Will man jedoch auf dem Gebiet der Intensivpflege die Möglichkeiten, die ein Rechner bietet, voll ausnutzen, muß man zwangsläufig neue Monitore fordern; Monitore, die nicht nur (digitale) Daten liefern, sondern auch Daten vom Rechner entgegennehmen können, z.B. für Funktionstests, Eichung, Verstärkungseinstellung usw. CAMAC bietet dazu eine standardisierte, herstellerunabhängige Schnittstelle und eine mechanische Norm.
Während sich die Integration in das CAMAC-System bei einigen Monitortypen - z.B. solchen, die Raten als Ausgangsgrößen liefern - anbietet, sollen die Schwierigkeiten bei anderen Typen nicht verkannt werden: Hybride Technik ist verhältnismäßig voluminös, während der Platz im CAMAC-Crate (als Organisationseinheit pro Patient) doch beschränkt ist. AD-Converter sind noch ein echter Kostenfaktor. Hier bietet sich ein analoger Bus mit zentralem ADC über die freie Verdrahtungsebene im CAMAC-Crate an.
Das für den industriellen Hersteller von Medizinelektronik wichtigste Argument ist natürlich die fehlende Stückzahl; denn in absehbarer Zeit wird der Computer nicht zur Standardausrüstung einer Klinik gehören. So bietet sich ein Kompromiss an, der auch längere Zeit bei elektronischen Meßgeräten praktiziert wurde: Weitergehende Digitalisierung und Normierung auch der analogen Ausgänge, Ergänzung oder Ersetzung von Handschaltern durch elektrische Schalter (Relais, Halbleiterschalter), die auch von An-

schlußsteckern her erreicht werden können, und Ergänzung solcher im wesentlichen noch konventioneller Monitore durch Lieferung von Interfacekarten zu CAMAC.

2.2 Representation von Daten bzw. Ergebnissen.

Ein Zweck des EDV-Einsatzes in der Intensivpflege ist das Aufbereiten der gesammelten Daten zu "lesbarer Information", ferner Verknüpfung von Parametern und Erstellen von Entscheidungshilfen für die Therapie. Die wohl anschaulichste Darstellung von solchen Rechnerergebnissen sind Graphiken, das bedeutet: das Ausgabemedium muß graphikfähig sein. Denkt man in computerbezogenen Zeitmaßstäben, so sind solche Rechnerausgaben jedoch "seltene Ereignisse". Außerdem wird das Ausgabegerät immer in einiger räumlichen Entfernung vom Rechner stehen, also patientennahe. Deshalb ist für ein geeignetes Ausgabemedium unbedingt ein eigener Anzeigespeicher zu fordern. Das Verhältnis von Bildschirmgröße zur darstellbaren Information (Bildspeicherkapazität) sollte gut sein (hoher Informationsgehalt einer Graphik), und schließlich muß das Gerät erschwinglich sein. Nimmt man alle Forderungen zusammen, so scheiden die meisten der üblichen Computerterminals aus.
Für unsere Zwecke hat sich als gute Lösung die Verwendung von Speicheroszillographen mit ca. 15 x 20 cm Schirmfläche herauskristallisiert. Neben der Speicherfähigkeit und der hohen Informationskapazität bietet der Betrieb als normales Oszilloskop zur Darstellung analoger Signale einen entscheidenden Vorteil gegenüber Sichtgeräten, die im TV-Modus arbeiten.
Da die Ausgabe von Graphiken ein relativ seltenes Ereignis ist und keine hohe Priorität hat, kann trotz der großen zu übertragenden Informationsmenge der normale CAMAC-Weg benutzt werden, ohne das Gesamtsystem übermäßig zu belasten. Im einfachsten Fall können kommerzielle CAMAC-DA-Converter zur Ansteuerung dieser Sichtgeräte benutzt werden. Ergänzt man ein derartiges Sichtgerät durch eine kleine Tastatur, deren Implementierung wegen ihres digitalen Charakters im CAMAC keine Probleme aufwirft, so ist der Abruf von Programmen, die Eingabe von Zahlen und Primitivdialog mit dem Rechner möglich.

2.3 Therapiesteuerung.

Auch auf diesem, noch sehr in den Anfängen stehenden Gebiet ermöglicht der Einsatz von CAMAC praktisch ohne neu zu erfindende Hardware vom Rechner aus das Setzen von Sollwerten, von Schaltern, die Ausgabe von Führungsgrößen durch Verwendung von kommerziellen Output-Registern und DA-Convertern.

3. Im vorausgegangenen wurde das technische Problem des Anschlusses von peripherer Elektronik an einen Rechner dargestellt. Es gibt jedoch im klinischen Bereich auch ein räumliches Problem:
Die Patienten in einer Intensivstation liegen in abgeteilten Boxen oder sogar in verschiedenen Räumen. Geht man von der naheliegenden Konzeption aus, einem Patienten ein Crate zuzuordnen, so ergeben sich zwischen den einzelnen Crates nicht zu vernachlässigende Entfernungen. Um die Wartung der Elektronik zu erleichtern, um die Pflege des Patienten für das Personal zu erleichtern und um damit die Technik dem Wachstationspersonal akzeptabler zu machen, sollte man zwar versuchen die Elektronik aus dem direkten "Pflegebereich" zu entfernen, doch ist das praktisch bei der Raumnot vor allem in älteren Kliniken nicht zu verwirklichen. Bei solcher "abgesetzter Elektronik" wäre eine gewisse Zentralisierung und damit der Einsatz des bisherigen parallelen CAMAC-branch möglich. In allen übrigen Fällen ist der (geplante) serielle branch eine wahrscheinlich annehmbare Lösung dieses Problems (Spezifikationen stehen noch aus), zumal die Datenraten in diesem Aufgabengebiet nicht übermäßig hoch sind (Maximum: ca. $5 \cdot 10^3$ Werte/sec und Bett bei 3-kanaliger EKG-Digitalisierung). Voraussichtlich nur bei extremen Anwendungsfällen muß man auf eine non-standard-Sternkonfiguration zurückgreifen, wie sie von einigen Firmen angeboten wird, wenn man es nicht vorzieht mehrere branch-highways zu verwenden.

4. Schließlich soll noch kurz geschildert werden, wie wir unser erstes Versuchssystem mit CAMAC in der Intensivpflege organisieren wollen. Es ist zunächst nur für einen Patienten gedacht, umfaßt also nur ein Crate als Interface. Als Monitore sind sowohl konventionelle Typen als auch bei uns entwickelte Geräte mit digitalen Ein/Ausgängen vorgesehen. So ergeben sich drei Typen von Datenquellen:

a) digitale Minutenmittelwerte
b) stoch. digitale Werte (Trigger)
c) Analogsignale.

a) Die digitalen Minutenmittelwerte stehen quasistatisch mit 12 bis 16 bit Wortlänge an. Es handelt sich z.B. um mittlere Herz- und Atemrate, Temperatur u.ä. In festen Zeitintervallen werden die einzelnen Monitore auf Rechnerinitiative über einen digitalen Multiplexer (Eigenbau) und ein CAMAC-Input-Register eingelesen.
b) Auch diese Werte liegen quasistatisch digital mit 12 bit an den Monitorausgängen. Sie müssen jedoch auf Triggersignale hin abgeholt werden, (z.B. Herz-, Atem-Intervalldauer etc.). Um hier den Datentransfer auf dem CAMAC-Weg effektiv zu gestalten, werden die Intervalle zusammen mit einer 4-bit-Kennung zunächst in einem 256-Wort CAMAC-Puffer-Modul abgelegt, der, wenn gefüllt, in einem autonomen Blocktransfer seinen Inhalt an den

Rechner übergibt.

c) Zunächst ist nur vorgesehen, langsame Analogsignale wie mittlerer Blutdruck etc. zu digitalisieren, was unter Rechnerinitiative von einem CAMAC-ADC-Modul mit Multiplexer geschehen kann. In einer späteren Ausbaustufe können bei zusätzlicher Verwendung eines Timers und zweier Puffer-Module auch schnelle Analogsignale wie das EKG mehrkanalig konvertiert werden.

Als Datensenken sind einmal ein Infusionsgerät vorgesehen, das über ein Output-Register seinen Sollwert erhält; ferner in der Verstärkung per Rechner setzbare EEG/EKG-Verstärker (in CAMAC integriert, Eigenbau); und schließlich ein Terminal, das aus einem Tektronix 611 Speicherschirm, einer Tastatur, DA-Wandlern und Steuerelektronik besteht (Eigenbau) und eine 8-bit-Schnittstelle hat, die über eine Interfacekarte an den CAMAC-Datenweg angeschlossen und im Blocktransfer bedient werden kann.
Das Crate mit Cratecontroller A 1 wird an einen vom ILL-Grenoble entwickelten Systemcontroller angeschlossen, der über einen Kanal mit DMA-Eigenschaften mit dem Rechner verkehrt.

COMPUTERS ON LINE IN NEUROPSYCHOLOGICAL RESEARCH

E. Andreewsky
Groupe de Recherche de Neuropsychologie, Hôpital de la Salpêtrière, Paris, France

ABSTRACT

Psychological experimentation, with a teletype on line, was implemented, in the general framework of Neuropsychological research. It illustrated how to use a computer in some other way than usual statistics.
It seems now impossible to deal easily with the different types of experimental data (namely physiological signals) without the help of a special purpose on-line computer, and a suitable interface.

ZUSAMMENFASSUNG

Im Gesamtrahmen der neuropsychologischen Forschung sind psychologische Experimente unter Verwendung eines Fernschreibers im On-line-Betrieb durchgeführt worden. Damit wurde der Nutzen des Rechnereinsatzes für andere Zwecke als die üblichen statistischen Analysen veranschaulicht.
Ohne einen besonderen Rechner im On-line-Betrieb und einem geeigneten Interface dürfte es heute nicht mehr möglich sein, die verschiedenen Arten von experimentellen Daten (namentlich die physiologischen Signale) einfach zu verarbeiten.

RESUME

Un télétype en ligne a été utilisé pour réaliser des expériences en psychologie, dans le cadre d'un Laboratoire de Recherches en Neuropsychologie. On a ainsi montré que l'intérêt de l'ordinateur ne se limitait pas aux analyses statistiques habituelles.
Il semble maintenant impossible de traiter correctement les diverses données expérimentales (et notamment les signaux physiologiques) sans l'aide d'un calculateur spécialisé, pourvu d'un interface adéquat.

The aim of Neuropsychological Research is the study of psychological mechanisms underlying human cognitive function. The study of the pathology of these functions may indicate some elements of human normal brain processes.

Computer science can and must be implicated in these researchs on different levels (1) ; for the experimental ground the computer is a tool for the implementation of experiments and for the analysis of results - it is not only a passive tool and may be helpful for the design of new types of experimental research.

I - Computers and psychological processes

Computers are used in many experimental processes at least for data analysis. A computer on line is able to be more closely part of these processes, with the ability of direct access. Moreover, it is possible, with a suitable real time system, to implement experimentation more or less completely automatically controlled by means of an appropriate feed-back. Usually, it is impossible to introduce directly the experimental data : namely, all the physiological signals to be processed by a computer, must be collected through an interface (such as CAMAC). Only some very specific processes may be handled without such a device - using only a standard input output (teletype for instance) of a Computer. Some psychological processes, in keeping with the rythm of a general purpose time sharing system - may be handled with a usual teletype. Up to the present date, a teletype (remote console of a PDP 10 time-sharing computer) being all the hardware of the laboratory, we have dealt only with this terminal for psychological experimentations using computers. It is now necessary to implement new experimentations, using a suitable computer, and an appropriate interface, in order to :

- improve the efficiency of all the experimentations,namely by the suppression of manual handling of data,
- provide the possibility of really new types of experimental process.

2 - Experimentation with a teletype on-line

The applications range from the acquisition of data, to man-machine interaction through the use of computer as a measurement tool.

2.1. Acquisition of data on-line

The production of a sequence of strikes made on the keys of the keyboard teletype by the subject, with a given instruction, are the data. The input is performed on-line. The records are constituted for each subject, by :

- the sequence of the strikes (a,m,p,r, etc...)
- the time intervals between every strike (they are approximatively several tenths of a second).

These data allow the study of the strategy of both normal and brain damaged subjects,to obey the given instruction. We find, for instance, that if "to strike at random on four keys" is the given instruction, the strategy of subjects are well correlated with their cultural level (for both normal and brain damaged subjects); we can say also that specific strategy are used by some pathological subjects. A theoretical random process, computer simulated, enables to make comparisons with the experimental sequences (2).

2.2. The computer terminal as a measurement tool.

The acquisition of the strikes on-line had been used in order to estimate neuropsychological parameters For instance, to deal with visual parameters : a rather spectacular phenomenon is shown by some patients with brain lesions : they ignore the half of the space corresponding to their visual deficiency. This is called "neglect phenomenon". Two sets of striking were performed by subjects showing negligence troubles, with, in both case, the instruction to strike at random on all the keys of the keyboard. They were blindfolded for one of the sets; they strike normally (that is to say without neglecting one half of the keyboard) only in this case ; these data give some modality of the desease (4). The computer program supplies, in real time, numbers which are the measure of the negligence rate.

2.3. Man-machine conversation

Experiments with man-machine conversation give the calculator the care of directing progressively the behaviors of subjects. It offers very favourable conditions for the recording of the experimental parameters : stimuli, response, reaction time,etc, which are directly stored into the computer. The psychological data corresponding to the contact of the brain damaged subjects with the machine are positive : no patient showed surprise in front of this typewriter which "writes by itself" such as the output terminal appears to be : this very important point allows to undertake researches on what could be called "computer assisted rehabilitation" of aphasic patients. We have been able to note, with some aphasic patients, by the experimentation below, a true learning phenomenon : we have given a rather simple linguistic game, by means of the computer terminal, to the subjects. The subject had to guess a french word of five letters. He put forward a sery of words (all french, with five letters : this was checked up by the computer) and the computer gave for each word the number of identical letters which are in the same position as in the word to be found. For instance, if the word to be found was :

ECOLE, and the subject proposed
MOULE
computer answer was "2"
for CHIEN
computer answer was "0".

This game presents a priori difficulties for all aphasic subjects. The algorithm which must be established, in order to find words presenting severe morphological characteristics, is a new linguistic algorithm for the patient. This kind of game may be successful to help aphasic to deal again correctly with their language ; on the other hand, it allows the test of pedagogical strategies, with the recording of all data.

3 - Experimentation to be carried out.

We will try to point out some different research, where the data (essentially physiological signals) must be handled by a special purpose computer, with an appropriate interface.

3.1. Speech pathology

Computers assisted rehabilitation for aphasic patients must deal with both spoken and written words. To handle spoken words, it is possible to use the algorithms developped within the researchs in automatic speech recognition. To analyse the production of pathological phonems, uttered by aphasic patients, it is necessary to implement new algorithms. In both case, the on-line acquisition, with some preprocessing, of all the speech data, is the first step to allow further researches. Speech parameters are more or less ill defined, and experimental researchs on this topic are necessary to understand some pathological phenomenon, in the framework of aphasia.

3.2. Electro-oculography

It is impossible, without a direct input in a computer, to implement a set of systematic tests, using electro-oculography techniques.

Electro-oculography give the measure of electrical data (from a negative electrical charge of about 1 mv, close to the macula) provided by the eye movements.

These data must be processed to characterize these movements : one must know the speed, the latency of the movement for both right and left side, in order to determine pathological parameters. More elaborated characterization is that of the used strategies to recognize visual patterns, or to follow a moving spot.

It is possible, then, to study the defective planning of the searching process ; for instance, a test aiming at recognizing a chimera, showed for "negligent" subject the grasp of the right or left part of the chimera, with a wrong brain completion of the other half. All these measures are implemented for the determination of pathological parameters such as space preference for subjects with hemianopsia (loss of half of the visual field), or wrong scanning strategy in the visual exploration of an image or in the process of reading (5).

With a computer on line, all these measures could be performed on numerous subjects, and statistical analysis could be easily carried out. On the other hand, more sophisticated tests could also be easily implemented.

3.3. E.E.G

The evoked potentials, measured with the E.E.G method, allow a comparison of the performance of a subject at two different times, or, at the same time, the comparison of the activity of the right and left visual fields.

It is very necessary to process these data into statistics, in order to understand the disturbances in cerebral pathology.

The same necessity is to be found with the data of CNV (contingent negative variation) or these of pathological sleep encephalograms.

All these data must be preprocessed before recording in order to carry out only discriminating features. If not, it becomes non realistic to process these too numerous data. Here also, the need of online acquisition and preprocessing is clear.

4 - Conclusion

Up to now, the use of computers in the implementation of psychological experimentation was limited by the hardware disponibility.

Experimentation with a teletype on line was carried out, in the general framework of Neuropsychological Research as an illustration of how to use a computer in experimental research in some other way than usual statistic.

Now, neuropsychologists wish a special purpose calculator one line, with a suitable interface, to deal easily with the different types of experimental data, and to implement more sophisticated experimentation.

BIBLIOGRAPHY

(1) ANDREEWSKY E. Informatique et Recherche en Neuropsychologie L'Encéphale N°1, 1972, pp 1- 20.

(2) ANDREEWSKY E. et NICOLAS P. Analyse d'un comportement pseudo-aléatoire. L'Encéphale N°2, 1973, pp.105-125.

(3) ANDREEWSKY E. Temps réel et mode conversationnel en Recherche expérimentale Neuropsychologique. Rev. Informatique Médicale, 4 (2) pp. 83-88, 1973.

(4)) CHEDRU F., LEBLANC M. A visual searching test. Its application to the study of unilataral inattention. Int. J. of Mental Health pp. 55-64, 1972

(5) CHEDRU F., LEBLANC M., LHERMITTE F. Visual searching in normal and brain damaged subjects. CORTEX, Vol IX, pp. 94-111, 1973.

(6) LHERMITTE F., CHAIN F. Données préliminaires sur les mouvements du regard en pathologie cérébrale in: La Fonction du Regard, ed. by A. DUBOIS-POULSEN, G.C. LAIRY, A. REMOND. Colloques de l'INSERM, Paris, 1969.

AUTOMATIC ANALYSIS OF SLEEP

F. von Niederhäusern
SEN Electronique, Geneva, Switzerland

ABSTRACT

Polygraphic recordings of human sleep are automatically analysed by a system which consists of analog and digital electronic instruments and a small computer. The system gives a minute by minute diagnosis of sleep stages, along with comments (e. g. artefacts) and numerical results for rapid and slow eye movements, muscle tone and heart respiratory rates.

ZUSAMMENFASSUNG

Polygraphische Aufzeichnungen des menschlichen Schlafs werden automatisch von einem System analysiert, das aus elektronischen Analog- und Digital-Instrumenten und einem Kleinrechner besteht. Das System liefert Minute für Minute eine Diagnose der Schlafstufen mit Erläuterungen (z. B. Artefakte) sowie numerische Ergebnisse betreffend schnelle und langsame Augenbewegungen, Muskeltonus sowie Herz- und Atmungsfrequenz.

RESUME

Les enregistrements polygraphiques du sommeil humain sont analysés automatiquement par un système composé d'instruments électroniques analogiques et digitaux associés à un petit ordinateur. Le système fournit minute par minute un diagnostic des différentes phases du sommeil, accompagné de commentaires ("artefacts" - phénomènes d'origine artificielle - par ex.) ainsi que des résultats numériques concernant les mouvements de l'oeil rapides et lents, le tonus musculaire, les rythmes cardiaque et respiratoire.

1. INTRODUCTION

In 1955 Aserinsky and Kleitman detected paradoxical sleep, i.e. a sleep stage with dream periods. Since then many studies of the physiology of sleep have been made on humans and animals in both normal and pathological conditions. Another line of research tries to classify the disturbances of human sleep in order to improve the specificity of well-oriented therapies. This article contributes to that field of research.

The common classification of sleep comprises 3 sleep stages: awake; "slow" sleep (with a slow Electroencephalogram (EEG) activity), "fast" or paradoxical sleep (with a fast EEG activity, sleep with dreams and rapid eye movements). Slow sleep itself can again be divided into four stages numbered from "1" to "4", where "1" corresponds to lightest and "4" to deepest sleep. In normal sleep the duration of each stage and its sequence throughout the night is well known: the change of the stages occurring periodically according to certain rules. In pathological sleep, bad organisation of these sleep stages is one of the most frequent disturbances: the sequence of the sleep stages may be abnormal, some stages being deficient or completely absent. In some cases the sleep stage itself shows abnormal characteristics.

The recording of the EEG alone is not sufficient to characterize the sleep stages. There are other physiological data which have characteristic patterns and are therefore important to know. The eye movements (EOG) and muscle activity (EMG) are quite different in the awake state, "slow" and "fast" sleep. Heart (ECG) rate and respiratory (ESG) rate give additional information, but the knowledge of these two parameters alone would generally not allow reliable recognition of the sleep stages. The study of the stages for sleep diagnosis is therefore based on the EEG, the EOG and the EMG graphs. Moreover, disturbances occurring during the recording like e.g. body movements or humming have carefully to be taken into account.

The graphs recorded during a whole night have a paper length of 200 m to 300 m. The visual inspection is therefore time-consuming, the diagnosis also varying slightly from one examiner to another. By automatic analysis quick and reproducible results can be obtained permitting comparisons and further calculations, especially by statistical methods.

Several methods of automatic analysis, either digital or analog, have so far been proposed. However, they do not always allow continuous analysis of the EEG over a long period and seldom treat all the variables usually recorded on paper. Here a hybrid solution with both analog and digital techniques has been preferred. This procedure corresponds partly to the analysis made by a human examiner, i.e. pattern recognition followed by logical conclusions.

2. BRIEF SYSTEM DESCRIPTION

The system presented here (Fig. 1) has been designed in accordance with hybrid analog/digital technique. It is relatively small equipment allowing automatic and standardized analysis of sleep EEG for a whole night, to be performed in the laboratory. The satisfactory results obtained are due to close collaboration of a medical doctor and an electronic engineer (psychiatric clinic Bel-Air, Geneva and SEN Electronique). During the night a 7-track analog tape recorder (Philips Analog 7) is branched to the output of a standard EEG equipment. In addition to the EEG, eye movements (EOG), muscle activity (EMG), respiration (ESG) and the electrocardiogram (ECG) are recorded (Fig. 2). Afterwards this tape is analysed by a system consisting of analog and digital measuring electronics (realized in standard CAMAC instrumentation) combined with a small computer (Nova). The cheapest output device is a typewriter, whereas peripherals like plotters or visual display are optional. The computer calculates the diagnosis. Output of the results may occur in several ways. The most precise results are obtained by printing minute by minute the diagnosis, the number of rapid and slow eye movements, muscle activity and heart and respiratory rates (Fig. 3).

By plotting curves instead of printing numbers the results are a little less precise but more expressive.
For the analysis the tape runs 16 times faster than for the recording, i.e. a whole night is analysed in about half an hour (the printout of the results may take slightly longer). Therefore one analysing equipment can analyse the tape of several patients within one day. For medical staff the use of the magnetic tape technique offers another advantage: during the night the only recording element is the tape recorder. Paper is not required and therefore no technician must supervise the paper feed and pens all through the night. If a copy of the graphs is still wanted on paper, this copy can be made afterwards from the magnetic tape.

The first results obtained with this equipment are very encouraging: differences between visual classification and automatic diagnosis are small (10-15%). Moreover the quantitative information on EOG, EMG, ECG and ESG which is printed out could not be available by visual methods unless a technician would work for days and weeks counting signal peaks and measuring slopes or amplitudes. Research laboratories need flexible kinds of equipment which can in the future be adapted to new techniques. The equipment described in this leaflet fulfills this condition excellently: the modularity of the hardware allowing the exchange of channels or addition of new modules and it is easy to modify the software which controls the whole system. Moreover, experiments on animals can also be analysed since the measuring ranges have been made wide enough to include these signals.

3. ELECTRONICS

3.1 Recording during the night.

The recording of the seven channels on tape during the night can be done by different kinds of commercially available EEG equipment which can be linked to the Philips Analog 7 tape recorder. Other tape recorders may involve a modification in the analysing system since the filters which reject spikes due to the FM of the recorder are normally matched with the Philips Analog 7.

Recording speed:	15/16 inches/sec.
Full range:	± 1 V
EEG amplification:	8'000 (50 uV on the electrodes corresponds to 400 mV on the tape).

As mentioned before the tape runs 16 times faster for the analysis than during the night. Therefore there are two time-scales which have to be specified: real time (t_r) refers to the recording, machine time (t_m) refers to the timing in the analysing system and is 16 times faster. A minute (t_r) of the night e.g. lasts 1/16 minute = 3,75 sec (t_m) for the analysis. Another example of these different times will be given in the following option:

Automatic start of the analysis (optional):
At the beginning of the night a start mark can be recorded onto track No. 1 of the tape at the point where the subsequent analysis should start. The start mark will automatically be detected by the analysing system. This start technique allows a precise and reproducible definition of the start of the analysis. The specifications of this start mark are:

Amplitude:	+ 800mV
Duration:	5 sec (t_r) or 0.312 sec (t_m)

Contact SEN for the installation of this optional start marker.

3.2 Analysis

The block diagram in fig. 4 shows 3 essential blocks: the tape recorder, the analog and digital electronic instruments (=CAMAC modules) and the Nova computer with teletype.

3.2.1 The recorder:

The reproducing speed is 15 inches/sec. For further information on the recorder, see chapter D 1.

3.2.2 Electronic instruments:

These are all CAMAC instruments, and the modularity of CAMAC allows the exchange of channels or later addition of other modules like e.g. visual display drivers or new measuring channels. It can be seen in Fig. 5 that the left half of the crate is still empty.

The analysing system contains analog and digital electronics as previously mentioned. The analog modules (type numbers 120-87 to 120-90) extract characteristic patterns out of the recordings or digitizing analog values. This brings about a vast reduction of data. The digital modules (4S2004, 2OR2008, 2IR2010) link the analog modules to the CAMAC dataway and the crate controller CC2023 interfaces the CAMAC dataway to the computer. The analog modules have been designed especially for this EEG analysing system whereas the digital modules are standard SEN products widely used in all kinds of applications.
The analysis of the different graphs is done as follows:

EEG:

There are twelve band-pass filters (6 per module). The frequencies are given in table 1. The EEG tracks of the recorder are connected to several filters according to medical need, but each filter can of course accept only one track. The output signal of the band-pass filter (fig.6) is rectified, slightly integrated and sensed by an amplitude discriminator. The level of the discriminator is computer-controlled through the output register 2OR 2008 and the state of the discriminator is sensed by the input register 2IR2010. By varying the discriminator level the analysing system is more or less sensitive to that frequency.

The EEG graphs are also analysed by the discriminator module 120-90. The discriminator D01 detects fast slopes and the discriminator D02 detects signals which exceed an amplitude level for a certain time. The sensivities are selectable by switches on the front

panel.
If the automatic start detector in module 120-88 finds a start mark on track No. 1 (see chapter D1) it sends a signal to the input register 2IR2010. This signal can be used to start the analysing program in the computer. If there is no start mark recorded on the tape the analysis can be started manually by means of the start push-button on module 120-88.

EOG:

The module 120-88 contains two discriminators. The first analyses slopes and the amplitude of these slopes to extract the rapid eye movements (REM) out of the EOG graphs. The second discriminator measures amplitudes to detect the slow eye movements (SEM). The states of the discriminators are sensed by the input register 2IR2010. The levels of the discriminators are switch selectable.

EMG. ESG. ECG (module 120-89):

EMG: the muscle signal is rectified and integrated. The resulting voltage is converted into a frequency that is analysed by one of the scalers in the 4S2004 four-fold scaler module.

ESG: the respiration signals are reshaped by a discriminator before they are counted by one of the scalers in the 4S2004.

ECG: in order to extract the heart-rate a discriminator detects the QRS peaks. The output pulses of the discriminator are counted by a scaler in the 4S2004. The level of the discriminator is automatically optimized.

The ranges and optimum levels of all the discriminators as well as the specifications of the filters have been defined in a careful study and analysis of several recordings. The detailed specifications are given in the instruction manuals.

3.2.3 Nova computer with teletype:

The Nova or the Nova 1200 computer can be used for this application. Other Nova types are faster and are also adequate but their speed is not needed. (Nova computers are 16 bit machines with 4 accumulators and are used here with a 4 K memory).
The cheapest output device is a teletype. Besides that all small computer peripherals like a fast reader, a puncher or a plotter are optional.

4. SOFTWARE

There are two kinds of software: the standard software of the Nova computer and the special software written for the EEG analysis. The standard software is a typical small computer software library. This library is especially useful for generating new programs and for computer applications not in connection with the analysing system.

One excellent example is the program "Basic" which transforms the Nova computer into a powerful desk calculator.

The special software concerning the EEG analysis is constantly growing. Beyond the diagnostic program there exists a set of special programs which e.g. print out the number of responses of the different discriminators or allow the printing of hypnogram of the night on teletype.

The diagnostic program consists of three main parts: the data acquisition, the calculation of the diagnosis and the print-out of the results. The diagnosis of minute n is calculated while the data of minute n+1 are collected. The print-out starts as soon as the diagnosis of the first minute of the night is calculated and follows during the subsequent minutes as fast as possible, i.e. the speed is normally limited by the output device. An eight-hours tape is analysed in 8/16 hours = 30 minutes (t_m) whereas the print-out on teletype will last about 45 minutes.

The data acquisition is based on the following principle: the states of the different discriminator outputs are transferred to the computer memory every 160 ms (t_r), i.e. 375 times per minute (t_r). The number of "yes" responses of a minute (t_r) are summed up for every discriminator. This means that e.g. a band-pass filter (fig.6) gives a minimum of 0 and a maximum of 375 responses per minute. The scalers of the ECG and ESG are read every minute (t_r); the scaler of the EMG is read 15 times per minute but only two of these values are kept in the memory: the maximum and the minimum. The timing of the data acquisition is controlled by the clock of the module 120-88 which request a program interrupt every 160 ms (t_r) = 10 ms (t_m) and the interrupt subroutine transfers the information into the memory as mentioned above.

The calculation of the diagnosis starts with a control of the collected data as there may be too many artefacts which can falsify the results. Artefacts are recognized by their frequencies or slopes. The program therefore compares results of the different analysers which may detect these artefacts and calculates corrected data if necessary. The diagnosis is calculated by equations, the variables of which are: responses of the different filters, K potentials (essentially discriminator D02), rapid eye movements, minimum muscle activity and different kinds of artefacts. The flow chart of the dia-

gnostic program as well as the equations have been determined by a detailed analysis of several recordings. Normally the awake state and stage 3 plus 4 are the easiest to detect. The recognition of stages 1, 2 and paradoxical sleep is more delicate, especially because of the variations between different patients. However there are some criteria which seem reliable in spite of these differences. The program recognizes and prints out comments on special events like excessive humming (50 Hz), artefacts, body movements or electrode signals which are too small or flat.
The output of the results may occur in several ways. The most precise results are obtained by printing minute by minute: the diagnosis, the number of rapid and slow eye movements, muscle activity and heart and respiratory rates. By plotting curves instead of printing numbers the results are a little less precise but more expressive.

5. CONCLUSION

This equipment has been in semi-continuous operation for a little over two years, and has given complete satisfaction to the users. The soft-ware is now at the fourth revision stage, having been updated by the operators in step with developments made by data general, and expanded to cover their own medical requirements which have become more extensive as the full potential of the system has been exploited.

A second system has been ordered by the same clients, and it is a measure of the success of the initial design that the new system will be identical apart from the incorporation of CAMAC modules conforming to the latest EUR specifications.

References:

J.M. Gaillard. A.E. Simmen. R. Tissot. G. Bastard: Analyse automatique des enregistrements polygraphiques du sommeil. Published in the Electroenceph. Clin. Neurophysiol. (1971).

J.M. Gaillard. A.E. Simmen. R.Tissot. G. Bastard: Dépouillement automatique des enregistrements polygraphiques du sommeil. Published in Experiencia (1971).

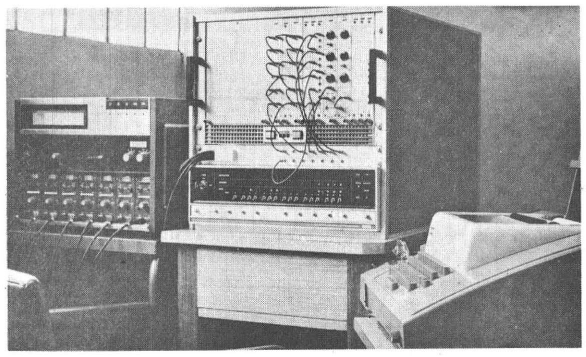


Fig 1 Analysing system with Philips recorder and teletype.

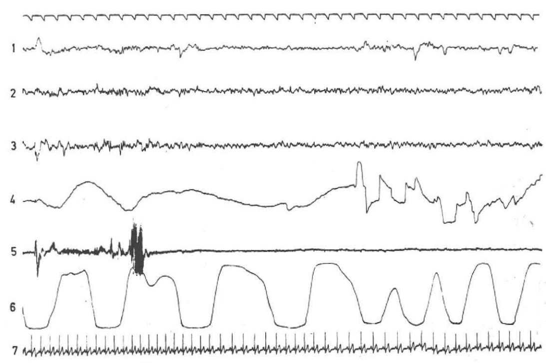


Fig 2 Recorded graphs
1,2,3 = EEG; 4 = EOG; 5 = EMG; 6 = ESG; 7 = ECG.

178	2	3	0	85	96	73	13
179	P	5	40	86	94	72	12
180	2	3	41	84	106	70	13
181	M 2	11	25	90	734	72	14
182	P	24	95	89	102	71	16
183	P	2	18	87	101	71	11
184	M P	14	40	89	441	73	11
185	P	25	97	86	111	71	12
186	P	4	18	85	169	70	12
187	P	5	35	88	100	66	13
188	P	17	78	90	100	73	15
189	P	11	71	86	112	74	12
190	A M P	3	31	85	393	72	12
191	P	0	0	86	100	70	11
192	P	0	0	84	114	74	10
193	P	4	68	84	107	63	15
194	A M P	9	23	84	536	73	7
195	P	0	0	82	93	64	13
196	P	5	22	81	93	64	12
197	P	0	13	85	94	68	13
198	P	0	0	84	165	68	11
199	P	0	0	84	94	64	11
200	P	0	6	83	102	64	11
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**

1 = minute of the night
2 = diagnosis
3 = rapid eye movements
4 = slow eye movements
5 = muscle tonus (min)
6 = muscle tonus (max)
7 = heart rate
8 = respiration rate

Fig 3 Output.

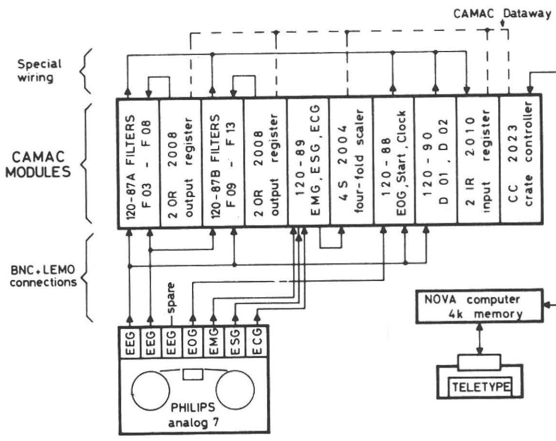


Fig 4
Block diagram of the analysing system.

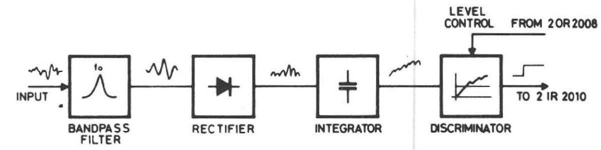


Fig 6
Block diagram of a filter channel
(six of each are in one filter module).

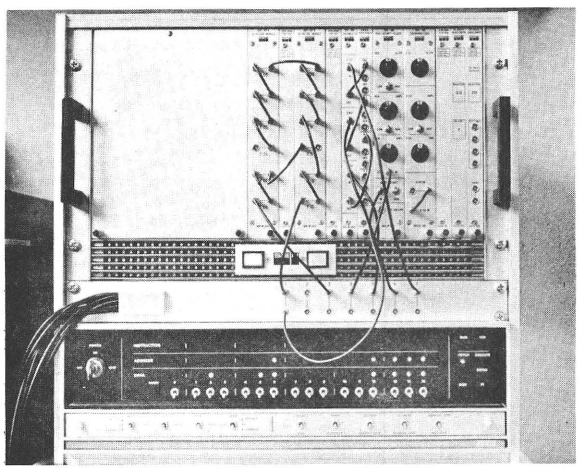


Fig 5
(from the top) CAMAC Crate with CAMAC
Modules; patch board for the 7 cables
coming from the recorder; Nova computer.

detected activities	filter	f_{res} (tr) c/s
slow artefacts	F03	0.3
delta (slow)	F04	1.3
delta (fast)	F05	2.3
delta-theta	F06	3.9
theta	F07	5.75
alpha	F08	9.5
beta (spindles)	F09	14.3
beta	F10	20
beta (fast)	F13	32
line frequency	F11 (twice)	50
artefacts (muscle)	F12	80

Table 1 Filter frequencies.

ON-LINE COMPUTING SYSTEM FOR DIAGNOSIS IN THE NUCLEAR MEDICINE

H. Gahr, Rudolf Virchow Hospital, Berlin, Germany
H. Pangritz, Hahn-Meitner-Institut für Kernforschung GmbH, Berlin, Germany

ABSTRACT

This report concerns an on-line computing system for routine measurements in the nuclear medicine, like test of the thyroid gland, nephrography and blood volume evaluation. In order to develop a system which can be adopted by other users with different computers, the CAMAC Branch Highway was chosen to be the only linkage between the computer and the peripherals necessary for the measurements, for the computer aided diagnostics and for the output of medical reports.

ZUSAMMENFASSUNG

Gegenstand des Berichts ist ein On-line-Datenverarbeitungssystem für Routinemessungen in der Nuklearmedizin, z. B. Schilddrüsentest, Nephrographie und Messungen des Blutvolumens. Damit das System von anderen Benutzern verwendet werden kann, welche verschiedene Rechner einsetzen, ist der CAMAC-Branch Highway als einziges Bindeglied zwischen dem Rechner und den Peripheriegeräten gewählt worden, welche für die Messungen, die Rechnergestützte Diagnostik und für die Ausgabe medizinischer Berichte erforderlich sind.

RESUME

Le présent rapport concerne un système d'ordinateurs en ligne destiné aux mesures de routine effectuées en médecine nucléaire, par ex. : test de la glande thyroïde, néphrographie et évaluation du volume sanguin. En vue de la mise au point d'un système susceptible d'être adopté par des utilisateurs employant des ordinateurs différents, l'Interconnexion de branche CAMAC a été adoptée : elle constituera le seul maillon entre l'ordinateur et les périphériques nécessaires aux mesures, aux diagnostics assistés par ordinateur et à la sortie des rapports médicaux.

ON-LINE COMPUTING SYSTEM FOR THE DIAGNOSTICS IN NUCLEAR MEDICINE

This report concerns a project for which work is going on for about half a year at the Rudolf-Virchow-Hospital in cooperation with the Hahn-Meitner-Institute in Berlin, but the work is not yet finished.

The on-line computing system is built-up for routine measurements in the nuclear medicine, for which about 3000 patients per year are sent to the hospital by their doctors. The examinations involve the radio jodine test of the thyroid gland, the nephrography with isotopes, the Schilling test and the measurement of blood volume and the residue of urine in the bladder. These are all measurements, which belong to the routine work of a hospital and which do not require new and complicated methods of measurement. Because of that this report will not give any detailed descriptions of the measurements. Instead there will be an explanation of concept as a whole referring to a highly simplified block diagram. Especially some of the problems will be mentioned, which we have already encountered during the starting phase of the project.

Figure

The upper part of the block diagram of Fig. 1 shows a computer together with a console teletype, paper tape input and output means and a disc. One of several aims of the project is to develop a system, which as a whole can be adopted by other users without major difficulties. It cannot be expected that other users of the system will all use the same type of computer. For that reason the CAMAC Branch Highway was chosen to be the only linkage between the computer with its peripherals and the other equipment necessary for the diagnostics in nuclear medicine. So several devices, which are usually belonging to the standard peripherals of an on-line computer, are connected to interface modules in the CAMAC-System.

It is known, that a change of computers requires an enormous software work due to the lack of an effective programming language for process control in conjunction with the CAMAC system. Nevertheless this form of hardware configuration seems to us to be the first step towards transferability of complete systems.

Measurements

Some of the measurements concern the evaluation of the up-take rate, storage time and elimination rate of a radioisotope within a specific organ of the human body, i.e. the measuring of a pulse rate as a function of time. A patient gets the radioactive substances either as a drink or by injection . The following measurements with a detector, which is located as near as possible to the respective part of the body, involve either many measuring points within a few minutes, as for example with the function test of the kidneys, or only a few values taken within hours or days. This applies to the function test of the thyroid gland. The detector includes a NaJ scintillation crystal, a multiplier and a preamplifier. The measuring equipment further includes a high voltage power supply for the detector and a main amplifier for the output pulses. These output pulses are applied to a differential discriminator for excluding all events, which do not fall into the energy range of the given isotope. The output pulses of the discriminator are counted during predetermined time periods.

The implementation of such a measuring equipment by means of the CAMAC system today still raises some difficulties, because there is no fairly broad spectrum of analogue devices in the CAMAC system, which can be adjusted and monitored via the Dataway. One has to choose other computer controllable devices in combination with CAMAC input and output registers, if one can find suitable devices for that purpose.

Computer controllable analogue devices are a supposition for a safe routine operation in a hospital. The adjustment by hand of an amplification factor or a threshold of the discriminator is always a potential source of faults. If these adjustments are done by the computer program, the preparation time of a measurement is shorter, measurement results are to a great extent reproducible and it is easy to increase the security of measurements by means of periodic calibration measurements.

Sample changer

A sample changer for the in vitro measurements, e.g. for the evaluation of the blood volume, is connected to the CAMAC instrumentation. In the present case special difficulties resulted from the fact that a device had to be used, which was not constructed to be operated in connection with a computer. It needs not to be emphasized that the problems of automatic identification and correlation of samples cannot be solved satisfactory with such a sample changer. In general the connection of medical devices to the CAMAC system involves the construction of special interfaces. That applies to a sample changer as well as to an automatic enzyme analyzer, the respiration rate monitor and the analytical balance. It is impossible to find anything like compatible ports.

In Germany there is an attempt to do the first step in this direction. A group of users of electronic devices in medicine is recommending a bit-serial, a byte-serial and a word-serial port hoping that the manufacturers will adopt those recommendations. If this effort is successful one day it will be possible to use commercially available CAMAC modules as interfaces for devices of that type.

In our opinion it was a significant omission not to recommend a narrow spectrum of external ports for CAMAC modules. The elaboration of such a recommendation would have been time consuming, and we had no time, and would have needed the confidence that the manufacturers of computer peripherals of any kind observe the CAMAC recommendations. Nevertheless such a recommendation would have saved work and time and money at various places.

When selecting devices which have to be integrated into a computerized system, it is advisable to choose those which do not only exchange data but allow the control or at least the monitoring of the main functions by the computer. This must be regarded as an important factor for increasing the security of a measuring system.

Dialogue
Every operating place is equipped with a device for alphanumeric input and output procedures to prepare, start and close the measurements in a dialogue with the computer. If a protocol is not necessary, it may be done with a display terminal, otherwise it has to be a teletype. The drawback of a teletype is the slow speed and the inevitable noise, which can be lowered by switching the motor on and off, if the pauses between the operations are long enough. The dialogue between the operating person and the computer is desirable and necessary for the correct performance of long series of measurements and should be very detailed to overcome all initial difficulties of the untrained staff. On the other hand the dialogue must be programmed in such a form, that a radical shortening by command is possible, because after some time a detailed dialogue is nothing more than a boring load for the operator.

There are some difficulties in programming when connecting devices to the CAMAC instrumentation, which normally are regarded as standard peripherals of a computer like teletypes, printers, displays and so on. A programmer of the user program must be able to use devices like that with the same comfort as the standard peripherals for which the operating systems of the computers include the respective routines like queue organisation, reservation, input and output macros. The integration of CAMAC peripherals therefore in most cases requires changes or additions to the operating system of the computer. It would be of great help if the computer manufacturers could in this respect offer some more support.

Diagnosis
Following to the measurements the doctors have to elaborate the diagnosis for a patient, and this diagnosis will be reported to the doctor, who has sent the patient to the hospital. This finding of the diagnosis and the writing of the report is supported by the computer. The measurement results are displayed on the screen of a display terminal as numbers or in the form of a diagram. The background memory of the computer contains an extensive library of textes for the medical reports. The doctor may call paragraphs of the text with the help of a keyboard to collect the whole text of a report on the screen of the display. The addition of new and special segments to the stored parts is possible. It is intended to support the diagnosis with respect to the measurement results by the computer program itself in displaying proposals for the text of the report. We are absolutely aware of the fact, that this "computer diagnosis" is very difficult because there are as an average only about 20 % classical symptoms for each case, i.e. the atypical symptoms prevail. The symptoms are more dependant on position, structure and function of the respective organ than on the cause of the illness. The mutal influence of several illnesses with the consequence of an immense number of combinations and differences of symptoms leads to a situation, where computer decisions according to logical connections can bring not only difficulties but pure nonsens. Moreover most of the biomathematical models do not take into account the dynamic character of the course of an illness.

A conventional storage tube display is used for the display terminals. Perhaps next year new plasma displays will replace the storage tube displays. The main features of these plasma displays are a greater brilliance, digital selection of the points to be displayed and partial erasement of the displayed information. Since the screen of a plasma display is a sandwich configuration of transparent glass plates with free access from front and back, constant information can be projected on the screen from behind and to overlaid by information from the computer. This way either the load of the computer may be reduced or information, which is not stored in the computer, can easily be compared with new information from the computer, especially when displaying curves. In the near future there will be a hardcopy device available for the plasma display, which so we hope will bring a noticeable simplification to our work.

The medical reports, which are edited by a doctor with the help of the computer and the display terminal and which contain the measurement results and the diagnosis, are printed by an electrostatic printer and plotter, which is connected to a versatile CAMAC interface module. The advantage of a print/plot device is the capability of simultaneous output of alphanumeric and graphic information. This way curves and text can be arranged side by side on the same sheet of paper, avoiding the chance of wrong combinations of text and diagram, which are always possible with separate output devices. The text should be printed with capital and small letters as it is with any typewriter. The changeover from operation by hand to a computerized system for psychological reasons should not bring results, which are in any respect worse than before. And a common computer print cannot be read as easily as a text written with a typewriter. These are those small things, which influence the decision whether a whole system will be adopted by the people concerned.

Operation
During the coming year the individual measuring setups will go into routine operation in the hospital. This demands a very thorough test phase, because a breakdown of the system leads to immense difficulties, since the patients have to be examined in any case. For an extended time period the present conventional measuring equipments will be running in parallel to the computer controlled CAMAC system. It is not before the end of the clinical routine test of hardware and software and after a profound training of the clinical staff, that all tasks will be taken over by the CAMAC system alone.

The modularity of the CAMAC system allows to repair all breakdowns within the instrumentation very fast and with a relatively small number of spare modules. On the other hand a breakdown of the computer affords an effective and fast maintenance crew. Although the measurement equipments can be operated with manual controllers the results lateron have to be fed into the computer for updating the data sets of the patients and for bringing the final report into the archives.

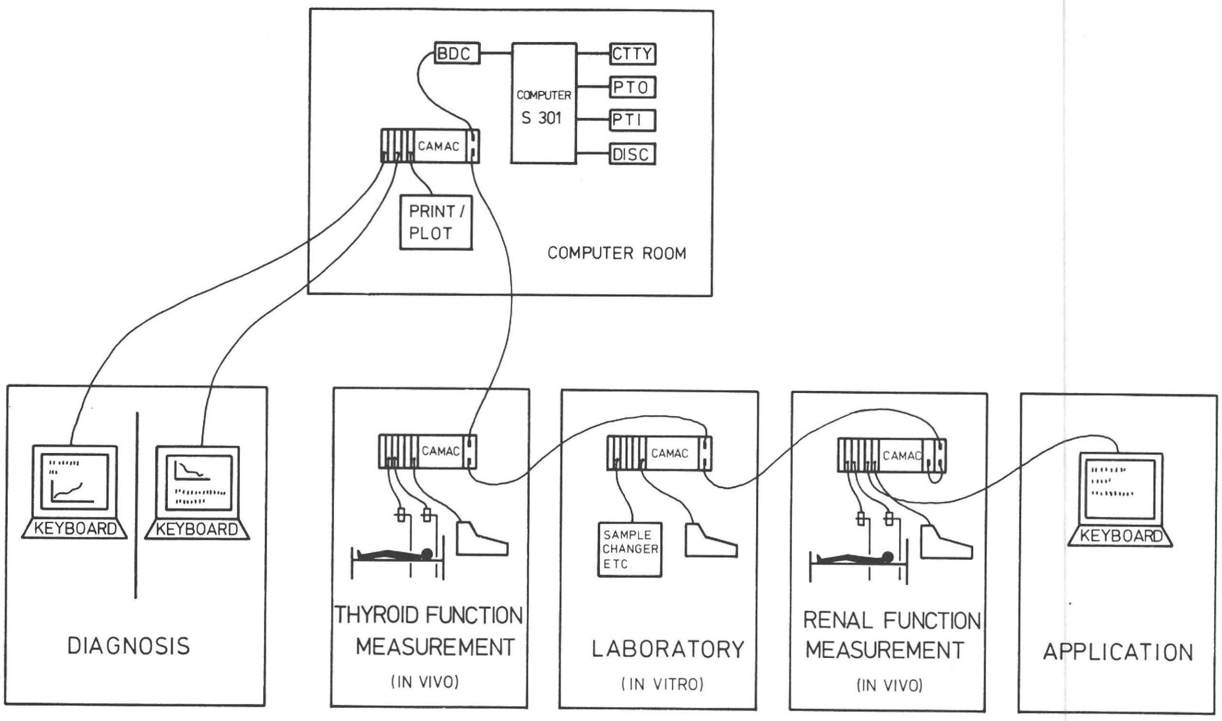


FIG.1: ON-LINE COMPUTING SYSTEM FOR THE DIAGNOSTICS IN NUCLEAR MEDICINE

A COMPUTER-CAMAC CONTROLLED SCANNING WHOLE-BODY COUNTER IN CLINICAL DIAGNOSIS

H. Heer, H. Halling
Kernforschungsanlage Jülich GmbH, Zentrallabor für Elektronik,
Jülich, Germany

ABSTRACT

A PDP-11/20 DOS-System (1,2 Megaword disc RKØ5, Dual Dectape) using CAMAC peripherals - with stepper motor controllers, interface modules for ADC's interrupt registers and scalers - is controlling a whole body counter. The total measurement procedure consisting of automatic adjustment using an anatomical model, set up of prologue data holding information about specific measurements and patient parameters, scanning and measurement program, on-line data acquisition and storage on disc as well as final control-procedures is done by the system, using FORTRAN and MACROS. An important feature of the system is the simplicity of handling the system by untrained personnel.

ZUSAMMENFASSUNG

Ein PDP-11/20-DOS-System (1,2 Megawort-Platte RKØ5, Dual-DEC-Magnetband) mit CAMAC-Peripheriegeräten - Steuereinheiten für Schrittschaltmotoren, Interfaceeinheiten für Anforderungsregister von Analog-Digital Wandlern und für Zählgeräte - steuert einen Ganzkörperzähler. Das System führt unter Verwendung von FORTRAN und MACROS das gesamte Messverfahren durch. Dieses umfasst die automatische Anpassung unter Verwendung eines anatomischen Modells, das Schreiben von Prolog-Daten mit Angaben über spezifische Messungen und über Parameter des Patienten, das Abtast- und Messprogramm, die Datenerfassung im On-line Betrieb, die Speicherung der Daten auf Platten sowie die abschliessenden Überwachungsprozeduren. Das System zeichnet sich dadurch aus, dass seine Handhabung auch für ungeschultes Personal einfach ist.

RESUME

Un système DOS PDP-11/20 (disque RKØ5 de 1,2 méga-mot, bande magnétique DEC Dual) avec périphériques CAMAC - avec dispositifs de contrôle du moteur pas à pas, modules d'interface pour registres et compteurs d'interruption d'ordinateurs analogiques et digitaux - contrôle un anthroporadiamètre. A l'aide de FORTRAN et de MACROS, le système effectue l'ensemble des mesures, à savoir l'adaptation automatique d'un modèle anatomique, la mise en place de données préliminaires comprenant des informations sur les mesures spécifiques au malade et sur ses paramètres, le programme d'exploration et de mesures, la collecte des données on-line, la mémorisation sur disque ainsi que les procédures de contrôle. Ce système est caractérisé par la simplicité de son fonctionnement, ce qui permet au personnel non qualifié de l'utiliser.

Assuming that not all of you are familar with medical research I will try to give a short description of the measurement-procedures.

During the measurements the patient, after being injected with isotopes, becomes a radioactive source. The activities are measured as a function of the position in the body. The decay times vary down to a magnitude of 1 minute. In addition to whole body measurements the system acts as a low level whole body scanner by detecting several isotopes simultaneously.

When measuring the two-dimensional activity distribution of the whole body the results say something about iron and magnesium resorption as well as concentration of vitamin B12 and potash.

On the other hand the activity as a function of time is measured at the organs to study the disintegration of protein and insulin.

The third group of measurements deals with the activity as a three dimensional function in the organism to search for tumours.

The measurement equipment is installed in a small steel room. The patient is situated on a scanning bed. Underneath and above four detectors are fixed. The bed can be moved in transversal direction, the detectors in longitudinal direction. The detectors deliver data simultaneously. The detected spectra usually have a size of 512 channels with measuring times from o.5 to 9o minutes. The data rates go up to 15ooo c/s/detector and up to 4ooo spectra are acquired per patient. The accuracy of the scanning runs is about 1 cm in each direction.

The computer should finally acquire the data, convert them, control the movement of the detectors and the scanning bed and organize the colour-display of the fitted data distributions for the doctor. As fas as the organisation of the measurements is concerned the different measurement modes are only distinguished by the positioning programs of the detectors and the scanning bed.

At the present time the data are transferred to the IBM/37o-165 by disc and after running the fitting procedures they are plotted.

Hardware description:

The computer system consists of a PDP-11/2o with 16 K memory, two RK05 discs (1,2 million words each) a Dual DEC-Tape and high speed papertape reader/puncher.

The great variety of equipment to be connected to the computer-system demanded a flexible modular interface. Therefore the CAMAC standard was chosen. Using a control-unit that is directly connected to the Unibus the different modules in a crate - the stepping motor controllers, ADC-interfaces, interrupt modules, counters, timers and multichannel analyser-adapter - are controlled.

The pulse height of the detectors is calibrated by the value of high voltage during an adjustment procedure. The detector pulses of the four detectors are fed to nuclear ADC's (13 bit Wilkinson). The digital outputs, that means the address, is fed to the input of a CAMAC ADC interface. A buffer-window for the incoming ADC data is set by software and adds the ADC address to a base and checks for overflow against a limit. When an overflow occurs the LAM is inhibited.

When the measuring time interval is completed the buffer contents are transferred to the disc.

Independent of the activity of the whole body counter, a blood activity measurement is performed using a 4oo channel multichannel analyser. After the data acquisition phase the 4oo channels are transferred to the computer via an adapter CAMAC module. The handshake controlled transfer is interrupt-driven. The data are converted and transferred to the disc and finally to the DEC-tape.

The stepping motors to move the detectors and the scanning bed are controlled by CAMAC stepper motor controllers which allow the adjustment of speed, direction and step count.

The Softwaresystem:

The Software consists of routines working under DOS. An assembler-program supports the adjustment of the detector-high-voltage at the beginning. The programming system for the whole body counter is also written in assembly lan-

guage. It controls the acquisition of the ADC data, the acquisition of the multichannel analyser data, the dead-time-measurement for the ADC's and the positioning of the detectors. Also the length of the measurement interval is changed in an interactive way using an interrupt routine.
The adjusting program as well as the measuring and controlling program use DOS-System I/O for data transfer. The second program also manages the organisation on the disc. The data are stored in real-format to be compatible to FORTRAN. The final data processing on the large machine (IBM/37o - 165) is done by FORTRAN and PL/1 routines and sometimes runs for some hours.
The main program, written in assembly language, calls FORTRAN routines for calculations as well as MACROs for controlling the hardware. The MACRO's are very similar to the single instruction-MACRO's of IML and offer high execution speed and easy programming.

The measurement procedure:

The measurement starts with the detector adjusting program which runs semiautomatically in the way that the program tells the operator how to adjust the high voltage manually. After doing this the optimisation procedure is repeated until the optimal adjustment is found.
Then the parameters for the patient are written with the aid of a prologue-dialogue program. Data concerning the detectors, the isotopes, the name of the patient, the date and the measurement-interval time are typed in and stored on disc. After that the background measurement using a dummy takes place. Then the patient is prepared with isotopes placed on the scanning bed and the main procedure starts. After each time interval up to four spectra are transferred to the disc together with deadtime of the ADC's, number and length of the interval. Via teletype the intervals can be changed during an interval gap. Data from the blood activity measurement are also transferred to the disc during an interval gap. After transfer of the data of the last interval, reference spectra are transferred for later comparison. This procedure is controlled by the operator supported by an interactive epilogue.
For the operator, the use of the system is simple in order to make the measurement procedure easier in spite of the use of a computer.
The former system used paper tape output to store the data. This caused a dead-time of up to 6oo % of the measuring time. It is very important for the comfort of the patient to get rid of such dead times which are now negligible.
The higher data rate drastically improved the time and scanning resolution.
In the near future it is planned to connect a coloured display via CAMAC to give the doctor immediate response.
To achieve this, immediate feedback from the IBM will be needed. Therefore the PDP-11 will be coupled to the IBM using a bit-seriel high speed link.

(1) ENGELS, H., SCHNIPPERING, H.G. Bisher unveröffentlichte Ergebnisse

(2) FEINENDEGEN, L.E., RITZL,F. Insulin-Umsatzmessungen in vivo mit 125-Jod und 51-Chrom, 7. Jahrestagung der Gesellschaft für Nuklearmedizin, Zürich, September 1969

(3) RITZL, F., FEINENDEGEN, L.E. Untersuchungen zum Insulinabbau im Menschen mit Hilfe der Doppelisotopentechnik, 8. Jahrestagung der Gesellschaft für Nuklearmedizin, Hannover, September 197o

GENERAL DISCUSSION

(concerning no particular session)

Q - R. Hankins

I would like to ask a very general question with regard to M.S.I. and L.S.I. (medium and large scale integration) development in association with CAMAC. It seems to me that their use in special purpose CAMAC module designs could solve the type of problem raised at the end of session IV. In addition the CAMAC/module interface (which is repeated in every module produced) is a particular case where such an I.C. could be readily produced by some enterprising manufacturer, and I would like to know if any work has been commissioned in this respect.

A - H. Klessmann

I think the ESONE Dataway Working Group (EDWG) is very much aware of the situation that L.S.I. could indeed help in simplifying CAMAC modules, especially in those cases where the same basic functions are required in all CAMAC modules.
However up until now the EDWG did not have any time available to do detailed work on this. Investigations have been made however in several laboratories.
We have made no real approach to manufacturers concerning this point because we believed there would be too small a market as long as CAMAC was used only for nuclear applications.
However, as this conference has shown CAMAC is really spreading into other application areas and it seems to me therefore that it is time to ask manufacturers to consider special MSI elements in order to improve the situation.

Q - R. Hankins

Has the ESONE Committee considered setting up a program Library Service, and if not, does one consider that such a service would be advantageous if provided.

A - P. Christensen

The Information Working Group of ESONE is at the moment considering a registration principle for manufacturers' software, as a basis for publishing a guide for software products in the CAMAC Bulletin. The principle may also be applicable to users' software. Tentative discussions on the topic of a users' society especially for software, have also taken place.

Q - H. Walze

I have a more general question, which refers not only to medical applications of CAMAC:
Because of the wide range of industrial and medical processes in which a serial branch highway system could be successfully applied, I had expected more interest in it. Two days ago Mr.Klessmann described the system briefly and announced that the specification will be available soon. Why, do you think, there have been no questions concerning the serial branch, what is the reason?

A - H. Klessmann

I should say that I have covered this matter only shortly during the introductory session and therefore not many questions could have been expected from the auditorium. We know however from discussions with potential users in different fields such as medicine, industrial process control, nuclear applications and astronomy that there are very many applications of the serial highway where the CAMAC parallel branch highway is not applicable. We know also that several manufacturers are interested.
At the ESONE General Assembly, this afternoon, I will give a status report of the Dataway Working Group where the serial highway will be considered much more in detail and I hope the available Description will be adopted by the ESONE Committee as a basis for a formal specification. Then this Description will be available to all interested persons, possibly next week, and the questions expected here will then come to us.

Q - B. Fefilov

The future of CAMAC was discussed in the first session. But a very interesting question has remained unanswered.
It concerns the possibility of substituting a miniprocessor for a crate controller. Contemporary technology allows the manufacture of miniprocessors of appropriate size to plug into a crate.
The price of controllers and miniprocessors is now comparable and the use of miniprocessors would solve many problems.

A - H. Bisby

There are many examples already given in CAMAC Bulletin of a miniprocessor type device in self contained autonomous systems within a single CAMAC crate (see for instance, CAMAC Bulletin no. 8, p. 15). I also know of a case where a pocket calculator has been incorporated in a double width CAMAC module as a local processor.
But already micro-mini computers are becoming available and will surely be used in CAMAC modular systems as you suggest. I shall be referring to the future implications of doing this in the concluding session.

SCHLUSS-SITZUNG

CLOSING SESSION

SESSION DE CLÔTURE

THE FUTURE OF CAMAC

H. Bisby

AERE, Harwell, England

1. Authoritative predictions from sources of far greater eminence than myself have often gone astray because, in the ultimate, men are motivated by considerations that are philosophical or financial, political or personal. Predictions of the future in the area of data processing, and therefore CAMAC, are made even more uncertain by the dramatic and rampaging advances in electronic technology. I shall therefore only indicate what might be possible and not be in the least surprised if no part is ever realised in practice.

2. This Symposium has demonstrated that CAMAC is a living standard and, without risking too much of my reputation, I am going to suggest that CAMAC will have a part to play in the unfolding interaction between the technological revolution and the advances in information processing. Some form of rationalisation or standardisation must be inevitable if the enormous potential of digital processing in real-time systems is not to be impeded by increasing effort and cost considerations and I believe this opinion is now gaining wide acceptance. It would be rewarding to think that this is the reason why so many have been attracted to this Symposium to see, and hear all about, CAMAC.

3. The foundations for CAMAC equipment and systems have been well and truly laid by the ESONE and NIM Committees in the preparation and specification of the Dataway, the Branch and Serial Highways and the Intermediate Language (IML). Since the first part of the CAMAC Specification appeared in 1969, nuclear laboratories and the instrument industry have built upon these foundations by implementing equipment designs that have enabled an escalation in systems applications. This is illustrated, (Fig. 1) by the rapid increase in the number of different types of equipment that are now commercially available. Some of this increase is due to supply companies providing equipment units that are either very similar or duplicates that satisfy broadcast specifications from the larger laboratories. Nervertheless, the increase is also due to new varieties of equipment being designed to satisfy new applications of CAMAC. During 1973, several examples have occurred

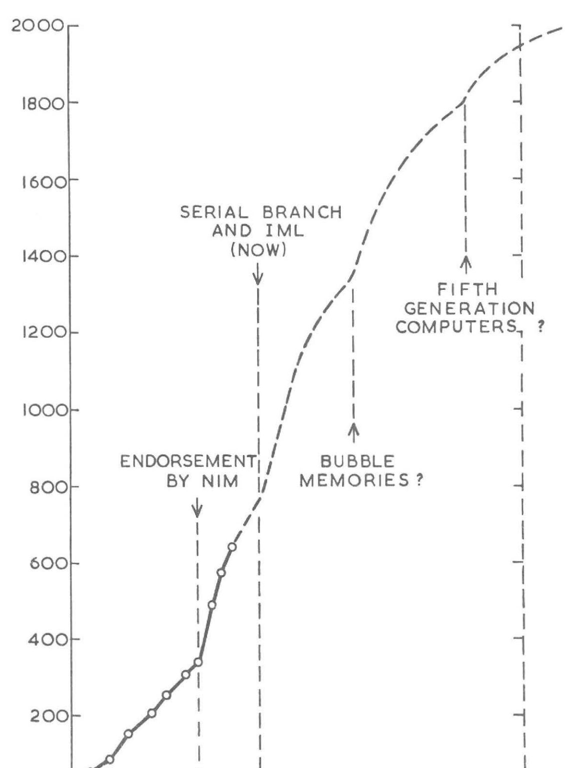


Fig. 1 Numbers of types of CAMAC Modules + Controllers. (Commercially available)

which indicate that CAMAC is being tested for its suitability and usefulness on the fringes of many non-nuclear applications. Without any reservations, I am convinced that CAMAC will find widespread use in these other areas during the next two years. The impetus provided by the imminent appearance of the Serial Branch and IML for a rapid expansion of new CAMAC applications will cause the generation of new varieties of equipment and thereby increase the number of equipment designs available commercially. Although, as time goes by, some of the earlier units will become commercially non-viable, this reduction will be more than off-set by the increased variety caused by wider applications and new technologies, as indicated in Figure 1.

By prediction of ~ 2000 varieties should not dismay the supply companies, because the increased varieties will cover world markets that are transparent for product supply, nor dismay the potential user of CAMAC, because the variety will enable a closer match between the system requirements and equipment available, or give the user a wider selection for given tasks. How this graph will relate to annual values of equipment and systems commissioned is more difficult to forecast, but I cannot imagine the increase in turnover being less than figures of 20% – 40% per annum that are accepted for the type of small computer that is used in on-line measurement and control systems.

4. Much of what I have said so far relies to some extent on the ability of CAMAC to satisfy the interfacing problems in non-nuclear areas. These requirements are broadly similar to those in the nuclear world, so that they should not need any changes in the compatibility rules of CAMAC so far specified. However, adaptation to those cases where the operational environment is different, could well lead to further developments and extensions of the CAMAC Specifications. In addition, and although the greatest impact of large-scale integration and the availability of very low-cost memories will be on the implementation of equipment designs and an increase in the facilities thereby available, modifications to CAMAC will be essential to keep it updated and acceptable. These standardisation activities will be initiated from the needs of CAMAC users, as they have been in the past and, clearly, there will have to be an organisation to deal with this work. Many possibilities spring to mind, such as a World Association of CAMAC Users and Suppliers. These questions will have to be resolved very shortly but, whatever the outcome, collaboration with the recognised International Standards Organisations will be essential. I would like to emphasise that CAMAC has no precedent in the international technical field, either by virtue of its origins, the breadth of its technical features or its total independence from invested interests in the commercial world. Therefore I conclude that the continued success of CAMAC and its very existence will be assured only if support for its maintenance is provided and, I like to think, shared between Suppliers and Users. An executive body of this nature, capable of dealing with the maintenance and development of CAMAC is not readily available within the frameworks of the recognised international standards organisations. And clearly, the problem is far greater than can be handled by the combined efforts of the ESONE and NIM Committees, as presently constituted.

Whether the Commission of the European Communities is prepared to play the part of an innovator in this context, by providing the catalyst that can bring about a workable interaction between Users and Suppliers, is a possibility that should be urgently explored, perhaps in conjunction with increased support within the various national zones by Government supported User/Supplier associations.

5. Again, the future of CAMAC must be intimately bound-up with the technological developments mentioned earlier that will provide inexpensive, large random-access memories, cheaper and more powerful data processing. These developments will give rise to a clearer distinction between the functions of data storage and computing. Already, for example, there is a trend towards setting up systems with identifiable separate locations that contain programmable local storage. This trend will accelerate because the distribution of computer-like capability has an enormous attraction for the local user, whether he is a doctor at a patient's bed-side, or a line-manager in a production plant or a scientist in a multi-experiment. These local stations will have to be connected to the transducers, the actuators and other peripherals such as displays, switch registers etc., and there is little indication that these devices will change radically within the near future. It is therefore reasonable to assume that CAMAC has an enormous potential for providing the multiplexing interface at these local sites. The local stations will also require access to more powerful and specialised data-processing facilities elsewhere, and again the fact that CAMAC can supply a standard port for bit, byte or word transmission will be a major consideration in its favour.

6. Decreasing comparative prices for equipment and increasing costs of manpower for program development will both emphasise the need to program these local facilities in a high-level language, a practice that has only recently become recognised as technically feasible. Within this context and because the flexibility of CAMAC enables the hardware to be very adaptable to the users' problems, it is vital that the program statements for CAMAC should be application isolated and computer independent. The IML language goes some way towards this and the fact that CAMAC statements can be embedded in existing high-level languages will make it possible for these local processing stations to be incorporated into the program instructions of the overall system both efficiently and with low-cost programming effort.

7. Within the foreseeable future, CAMAC has a bright prospect because it has so effectively anticipated large-scale integration and the massive storage that will form the basis of the evolution in data processing during the remaining years of this decade. By 1980, the users of automated data collection and reduction systems will have come to expect a degree of system adaptability and reliability that can only be obtained through distributed storage, processing and computing power. CAMAC is already proving its strength in such system concepts. Its future rests now on the degree of support available for its maintenance, adaptation and utilisation. Each of these three aspects are important in their own right, however, standardisation for the sake of standardisation is fruitless. The real value of any standard comes from its application and therefore its adaptation of the contemporary yet transitory state of electronic technology.

POTENTIAL METHODS OF FUTURE PROMOTION OF CAMAC APPLICATIONS

C. Layton

Director, General Directorate for Industry and Technology, CEC

Ladies and Gentlemen,

It is even more a pleasure for me to wind up this congress than it was to participate in opening it, because I have been so deeply impressed, not only by the size of the gathering but by the practicality of your work these last two days and by the feeling that CAMAC is on the point of making a break-through.

I think the congress has been extremely useful as a way of learning from each other, as a way of communicating between users and manufacturers and my impression has been that it has played a fairly critical role in generating the essential confidence that is needed to spread the use of a standard, such as CAMAC.

I think the number of those who finally registered was bigger than 500 participants and that in itself was an indication of the importance and the value of the congress.

I suspect that you have learned, most certainly I have, from learning from those who could teach by example, from the pioneers in new applications who are showing that CAMAC has a great many new kinds of uses, which are now being explored practically and bringing savings, calculable savings to users. In the field of applications to me, who as one who has not followed closely the development of CAMAC in the past, the break-through into the medical field was obviously extremely impressive. But the many new industrial applications also point the way to important future developments. Of course, the test will be, how many of you go away and in the next two years, learning from these examples, apply CAMAC systems to your own uses. But I am encouraged and I hope that some of you are too, and have learned things which will encourage you to at least examine the possibilities of using the standard for your own needs. All that I have learned confirmed the feeling I expressed when this congress opened, that the Commission must seek to support the further development of CAMAC, a living standard which obviously works and is being used more widely, in order that users can achieve the full benefits of interchangeability, in order that manufacturers can develop economies of scale in their production of modules and that all the wide benefits to both of a common standard are realized. We learned, I think, that there are also certain limitations to the use of CAMAC. It is not appropriate necessarily to the smallest uses, it is more appropriate perhaps in more complex applications, but I feel confident that the growing industrial use of CAMAC may well bring simplifications and improvements and cheapening and perhaps more rugged modules, always in the framework of the standard. It seems to be breaking throughout of the laboratory world into the industrial world and I am sure this will give it a new lease of life. Certainly the whole decentralisation of computing power, which will be a great feature of the next decade, provides and opens up an obvious need for a standard interface such as CAMAC.

Now, how to follow it up? H. Bisby mentioned already certain practical possibilities. I am sure the idea of a users and manufacturers association is a good one. It is right and excellent that CAMAC has originated amongst users and their voice must remain essential in the healthy development of the standard, but the time has come, I think, to bring manufacturers in too, so that both can benefit from the exchange and communication of ideas about the development within the framework of the standard. It obviously seems right to extend the group, really the ESONE Committee, which exists to-day for pioneering the system, into something wider. So I would underline the request that you fill in thoughtfully the question in the questionnaire that was distributed: would you like to participate in a users and manufacturers association? It is a very practical and important question, I think a question you should not answer academically, would it be a pretty thought to have such a thing but do you wish to join it. It was asked just now, would such an association be supported by the Community. Well, I will not reject the question, but the first thing the Community wants to know before being asked to support something is whether people want to create it.

It is the users and the manufacturers who need to communicate with each other, who have to decide whether such an association is valid and valuable as a permanent arrangement as compared with an occasion like a congress, which comes and goes. Do you need a permanent framework for association? If you do, then we can consider it and I hope and believe the Commission will look very favourably on the possibility of finding some means of supporting such an association. But, you would you please reflect carefully on this question, as to whether this might be a first important lasting step to bring you together and encourage the further development of CAMAC.

A second possibility, which I will simply mention, and I think it must depend on the fruits and results of this gathering, is whether a further congress of a similar nature might be valuable in say, two years time; obviously no one should make commitments about such an arrangement at the present time, but it is a possibility which should be considered.

A third and I think critical task for us in the Commission and if you like in the political organs of the Community is to consider whether and how procurement power of governments and public bodies may be used more effectively to encourage the use of the CAMAC standard. One of the objectives of our policy for data processing is to try to establish valid and useful European standards and gradually to make use of procurement power and the guidance and leadership of public authorities to introduce standards as the National Bureau of Standards has for instance in the United States. I think what we learned about the medical applications which have been introduced in Germany shows rather strikingly how a government or public authority can have a policy on such a matter and have a major influence on its use. I think we must look for instance in the Community at the possibility of whether the CAMAC standard may usefully be introduced through policy in other countries of the Community, for instance in the medical field, but not only in the medical field. This, I think, is perhaps the most specific and important conclusion which I have drawn as to our tasks and our possible fields of activity.

If we succeed in the Community in mounting over the next few years as we hope, a number of significant European applications development programmes designed to foster the development of applications of international character or design to save resources when applications are needed by several governments or authorities in different countries and if you want to save resources by pooling the development effort or pooling procurement, I am sure that in this work we must reflect carefully and see whether again we can make it policy when appropriate to encourage and promote the use of the CAMAC standard. The medical field is an obvious field where European collaboration is going to develop. I think it must be one of our priorities, not only the field of intensive care, but social security in administrative systems. Another field already mentioned and opening up for the use of CAMAC is environmental monitoring, I only mentioned it because I heard about it and that is certainly a field, which will come to have a European character and where therefore a European standard will be relevant. We also at least have in mind, as a potential field of policy, the sponsoring of certain applications developments in industry. What I have heard about the possible applications of CAMAC in the field of numerically controlled machine tool systems, is relevant and interesting. Now, therefore we should look very carefully at where we can use the real political power of the Community and the developing data processing policy we just began as a means of fostering and helping the spread of this valuable standard.

It is clear that there are certain other tasks related to the development of CAMAC which you and perhaps we may be interested in encouraging. One is the development of CAMAC related software, software that is interchangeable and can easily be used and transferred to different CAMAC systems. I would add that we also have a further interest at Community level in another related field of standards and that is the rather simpler standard interface between peripherals and computers on a one-to-one basis and that it seems to me, such a standard at a European level ought to be CAMAC compatible and related to it.

I believe in the Community we are entering a new phase where we can hope to try to make use of the best work that has been done so far in the standards field and bring some measure of political power and leadership to bear. I use the word leadership because in such work of course it is essential not to penalize the innovator, not to obstruct possible alternatives and variety, but leadership is necessary, as we know, to bring about the wide spread of a standard which can be useful to all. I was also interested and impressed by what was said about the future development of CAMAC. If we make a major effort to spread the use of CAMAC as an International and European standard it will certainly be essential that we are vigilant in ensuring that it is not a common obstacle to progress and that we are ready to transform it to take account of the radical technological changes which will happen in the next decade. I am sure that there will be some years now where the essential is to spread the existing CAMAC standard to new applications. But this readyness, this recognition of the need to change and evolve at a later point in time will be essential.

AUTOREN INDEX
AUTHOR INDEX
INDEX DES AUTEURS

A

ABEND P. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

ALATABER J. 121
CERN
CH-1211 Genève 23
Switzerland

x ALBROW M. G. 195
CERN
CH-1211 Genève 23
Switzerland

ANDREEWSKI E. 311
Hôpital de la Salpétrière
91, Bd. de l'Hôpital
F-75013 Paris
France

B

x BARBER D. P. 195
CERN
CH-1211 Genève 23
Switzerland

BECKER H. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

BIRI J. 269
Central Research Institute for Physics
P. O. B. 49
Budapest 114
Hungary

BISBY H. 3, 333
UKAEA, AERE
Dept. E & AP
Building 347. 2
Harwell, Didcot, Berks. OX11ORA
England

x BOGAERTS A. 195
CERN
CH-1211 Genève 23
Switzerland

x BOSNJAKOVIC B. 195
CERN
CH-1211 Genève 23
Switzerland

BRÄUNIG D. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

van BREDA I. G. 157
University Observatory
St Andrews
England

BRODDA B. G. 137
KFA Jülich, ZEL
D-517 Jülich
Postfach 365
Germany

x BROOKS J. R. 195
CERN
CH-1211 Genève 23
Switzerland

BUBLITZ R. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

C

CAMPBELL R. J. 157
University Observatory
St Andrews
England

CARR D. M. 157
University Observatory
St Andrews
England

CHAUDHRI B. 279
Borer Electronics & Co. AG
CH-4500 Solothurn 2
Solothurnstrasse, Biberist
Switzerland

x CLEGG A. B. 195
CERN
CH-1211 Genève 23
Switzerland

CLOUT P. N. 185
Daresbury Nuclear Physics
Laboratories
Daresbury, Warrington
England

COLLINS M. 121
CERN
CH-1211 Genève 23
Switzerland

CROWLEY-MILLING M. C. 121
CERN
Lab. II SPS, Control Group
CH-1211 Genève 23
Switzerland

CZAIKOWSKI P. M. 301
AEG-Telefunken
D-775 Konstanz
Germany

D

DEGENHARDT K. H. 101
HMI G. m. b. H.
D-1 Berlin 39
Glienickerstr. 100
Germany

DRURY D. 25
GEC-Elliott Process Automation
New Parks, Leicester LE3 1UF
England

E

ERNE F. C. 195
CERN
CH-1211 Genève 23
Switzerland

F

FARREN J. 109
UKAEA, AERE
Building 148
Harwell, Didcot, Berks. OX11ORA
England

G

GAHR H. 321
Städt. Rudolf Virchow Krankenhaus
D-1 Berlin 65
Augustenburger Platz 1
Germany

x GEE C. N. P. 195
CERN
CH-1211 Genève 23
Switzerland

GERTENBACH W. P. 141
Atomic Energy Board
Pelindaba
Private Bag 256
Pretoria
South Africa

GUERIN J. P. 273
CEA-CEN Saclay
SES/SERA
B. P. No. 2
F-91 Gif-sur-Yvette
France

H

HAASE V. 91
UNICOMP
D-7501 Blankenloch/Karlsruhe
Singerstr. 5
Germany

HALLING H. 63, 137, 325
KFA Jülich, ZEL
D-517 Jülich
Postfach 365
Germany

HAMMER D. 115
Technische Hochschule Wien
A-1040 Wien
Gusshausstr. 25
Austria

HEER H. 325
KFA Jülich, ZEL
D-517 Jülich
Postfach 365
Germany

HEINRICH G. 261
Borer Electronics & Co. AG
CH-4500 Solothurn 2
Solothurnstrasse, Biberist
Switzerland

HERDAM G. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

HILL P. W. 157
University Observatory
St Andrews
England

HILTON K. G. 207
GEC-Elliott Process Automation
New Parks, Leicester LE3 1UF
England

HOOTON I. 23, 57
UKAEA, AERE
Dept. E & AP
Building 347
Harwell, Didcot, Berks. OX11ORA
England

HUGHES G. 185
Daresbury Nuclear Physics
Laboratories
Daresbury, Warrington
England

HYMANN T. 121
CERN
CH-1211 Genève 23
Switzerland

I

ISELIN F. 9
CERN
Division NP
CH-1211 Genève 23
Switzerland

K

KAISER J. 173
CEA-CEN Saclay
SES/SERA
B. P. No. 2
F-91 Gif-sur-Yvette
France

KILLIAN K. P. 293
Städtisches Krankenhaus
Herlaching
Institut f. Klinische Chemie
D-8 München 90
Sanatoriumplatz 2
Germany

KINGHAM E. G. 129
CERL
Central Electricity Generating Board
Kelvin Avenue
Leatherhead, Surrey
England

MUSSTOPF G. 83
SCS G. m. b. H.
D-2 Hamburg 39
Überseering 8
Germany

N

von NIEDERHÄUSERN F. 315
SEN Electronique
31, av. E. Pictet
CH-1211 Genève 23
Switzerland

O

OTTES J. G. 251
GfK
Abt. LEM
D-75 Karlsruhe
Weberstrasse 5
Germany

P

PANGRITZ H. 321
HMI G. m. b. H.
D-1 Berlin 39
Glienickerstr. 100
Germany

PEATFIELD A. C. 169
Daresbury Nuclear Physics
Laboratories
Daresbury, Warrington
England

PETSCH H. 293
Städtisches Krankenhaus
Herlaching
Institut f. Klinische Chemie
D-8 München 90
Sanatoriumplatz 2
Germany

PFEIFER T. 243
Laboratorium für Werkzeugmaschinen
u. Betriebslehre der T. H.
D-5100 Aachen
Wüllnerstrasse 5
Germany

PORTH A. J. 289
Medizinische Hochschule
Institut f. Klinische Chemie
D-3 Hannover-Kleefeld
Postfach 180
Germany

R

RAUSCH R. 121
CERN
CH-1211 Genève 23
Switzerland

REHSE E. 307
Neurochirurgische Univ. Klinik
Abt. Rechenzentrum
D-4 Düsseldorf
Moorenstrasse 5
Germany

REIMER D. 221, 301
Dornier G. m. b. H.
D-799 Friedrichshafen
Postfach 317
Germany

RICHARD J. P. 153
CEA-CEN Saclay
B. P. No. 2
F-91 Gif-sur-Yvette
France

RIMMER E. M. 69
CERN
CH-1211 Genève 23
Switzerland

RISPOLI B. XVII
CNEN
Laboratori Nazionali
125 Vle Regina Margherita
I-00198 Roma
Italy

x RUDGE A. 195
CERN
CH-1211 Genève 23
Switzerland

RUSSEL R. D. 79
CERN
Data Handling Division
CH-1211 Genève 23
Switzerland

RYS A. 51
CEA
Bureau Technique
29, rue de la Fédération
F-75015 Paris
France

S

SCHERING G. 121
CERN
CH-1211 Genève 23
Switzerland

SCHINDLER A. 227
SEN Electronique
31, av. E. Pictet
CH-1211 Genève 23
Switzerland

SCHULZ H. J. 301
AEG-Telefunken
D-775 Konstanz
Germany

x SENS J. C. 195
CERN
CH-1211 Genève 23
Switzerland

SERVENT J. 97, 233
SAIP-Schlumberger
B. P. No. 47
F-92222 Bagneux
France

SOMLAI L. 269
Central Research Institute for Physics
P. O. Box 49
Budapest 114
Hungary

SPENCKER A. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

SPURLING K. 169
Daresbury Nuclear Physics
Laboratories
Daresbury, Warrington
England

T

THIEL R. 261
Borer Electronics & Co. AG
CH-4500 Solothurn 2
Solothurnstrasse, Biberist
Switzerland

x TIMMER J. 195
CERN
CH-1211 Genève 23
Switzerland

TRECHCIŃSKI R. 39
Institute of Nuclear Research
Swierk
Poland

V

VASHEGYI Gy. 269
Central Research Institute for Physics
P. O. B. 49
Budapest 114
Hungary

W

WAGEMANN H. G. 163
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

WAGNER H. 147
Siemens AG
Abt. E. MTEI
D-75 Karlsruhe
Rheinbrückenstrasse 50
Germany

WAWER W.
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

WEBSTER R. K. 109
UKAEA, AERE
Building 148
Harwell, Didcot, Berks. OX11ORA
England

WILHELM R. 121
CERN
CH-1211 Genève 23
Switzerland

WOLETZ W. 101
HMI G. m. b. H.
D-1 Berlin 39
Glienickerstr. 100
Germany

Z

ZACHAROV B. 169, 179
Daresbury Nuclear Physics
Laboratories
Daresbury, Warrington
England

ZANDER K. 31
HMI G. m. b. H.
Abt. Datenverarbeitung und Elektronik
D-1 Berlin 39
Glienickerstr. 100
Germany

ZENNER K. 243
Laboratorium für Werkzeugmaschinen
u. Betriebslehre der T. H.
D-5100 Aachen
Wüllnerstr. 5
Germany

(x) CHLM Collaboration