It may be seen from Table 1 that the program written in PL-11 may be directly transferred to MICE, where it runs as fast as highly optimized FORTRAN code runs on the CERN IBM 370/168. About 40% of the time is consumed by CAMAC instructions; this can be reduced to 25% with a special microcoded CAMAC instruction, or eliminated altogether by adding a FIFO buffer between the CPU bus of MICE and the histogramming memories. Finally, the event rate is doubled by microcoding the histogramming loop, which comprises only seven PDP-11 instructions, into eleven microcycles. This work required only one hour of a specialist's time.

We have found MICE to be totally reliable and easy to use. With the minimum of assistance from a specialist, we have implemented back-projection of our data, on-line, that runs three times faster than is possible off-line on the CERN central computer.

* * *

REFERENCES

1) J. Anthonioz-Blanc, C. Halatsis, J. Joosten, M.F. Letheren, A. van Dam, A. van Praag and C. Verkerk, MICE, A fast user-microprogrammable emulator of the PDP-11, Internal Report CERN-DD/80-14, to be published in Proc. Int. Conf. on Experimentation at LEP, Uppsala, 1980.

2) J. Anthonioz-Blanc, J. Joosten, M. Letheren and A. van Praag, User aspects of MICE, these proceedings.

3) A. Jeavons, K. Kull, B Lindberg, G. Lee, D. Townsend, P. Frey, A. Donath, A proportional chamber positron camera for medical imaging, Nucl. Instrum. methods 176 89 (1980).

DEALING WITH DISTRIBUTED INTELLIGENCE IN MONITORING AND CONTROL SYSTEMS

R.A. McLaren, CERN, Geneva, Switzerland

ABSTRACT

The European Hybrid Spectrometer is built up of many individual detectors, each having widely varying monitoring and control requirements.

With the advent of cheap microprocessor systems a shift from the concept of a single monitoring and control computer to that of distributed intelligent controllers has been economically feasible.

A detector designer can now thoroughly test and debug a complete monitoring and control system on a local, dedicated micro-computer, while during operation, the central computer can be relieved of many simple repetitive tasks.

Rapidly, however, it has become obvious that the designers of these systems have to take into account the final operational environment and build into both the hardware and software, features allowing easy integration into a central monitoring and control chain. In addition, the problems of maintenance and eventual modification have to be taken into consideration early in the development.

Examples of currently operational systems will be briefly described to demonstrate how a set of basic guidelines plus standardisation of hardware/software can minimise the problems of integration and maintenance.

Based on practical experience gained in the European Hybrid Spectrometer, investigations are proceeding on various possible alternatives for future micro-computer based monitoring and control systems.

1. Introduction to the European Hybrid Spectrometer

The European Hybrid Spectrometer (EHS) is composed of several detectors including the Rapid Cycling Bubble Chamber (RCBC) constructed by the Rutherford Laboratory, with its associated magnet M1, ISIS a large volume Multiwire Proportional Drift Chamber constructed by Oxford University, the Intermediate Gamma Detector, the Forward Gamma Detector and many other devices[1].

The tasks of data acquisition and monitoring are handled by two NORSK DATA mini-computers, the data acquisition by a NORD 100 and the monitoring by a NORD 10. It was decided to use the SPS multiplex system as an efficient means of handling the monitoring and control of these detectors[2]. The SPS multiplex system[3] is composed of one or more local crates (stations) which contain inexpensive analog and digital input/output modules, interfaced via a controller to the CAMAC highway.

2. Guidelines adopted for Monitoring and Control Microprocessors in EHS

2.1 Standardisation of hardware

When the basic monitoring and control concepts of EHS were initially considered it was foreseen that many of the groups participating in the construction would provide not only the detectors but also the associated monitoring and control systems.

It was decided to standardise as much as possible within EHS. A processor family (INTEL) and a general purpose system (SBC 80/20 based on the Multibus) were chosen and equipment designers outside CERN were asked to comply, if possible, with these standards. Since at this time, the potential designers of the microprocessor systems had little investment either in terms of experience or in development tools, the standards were readily accepted.

Several of the system designers chose to program in a high level language and purchased development systems to allow compilation and emulation. To provide support for on-site testing and to ensure future maintenance, the same development system was purchased for EHS. This has since proved an invaluable tool for the in-house development of microprocessor controlled quipment.

The various microprocessor systems in EHS were roughly categorized into three classes:

i) Random logic replacement - defined as any small system with a fixed mode of operation. The unit obeys simple external commands.
ii) Monitoring and control systems - defined as a system whose program is fixed but whose operation can be modified by commands from a central computer.
iii) Stand-alone microcomputer system - defined as a system whose program can be written on the system, normally supporting an interpretive language plus a simple operating system.

2.2 Communication protocols

An important requirement of these microprocessor systems was the need to communicate with the central monitoring and control computer and the data acquisition computer.

It was considered vital that, even if the type of processor or the system chosen did not comply with the standard, at least all processor systems should have a common interface to the central computers in both the hardware and software domains. These were defined at an early stage. The communication protocol chosen was dependent on the system classification. Systems in the first class function as a module connected to a standard bus, for instance a module designed for use in a CAMAC system responds to application oriented CNAF commands according to the CAMAC standards. A system which is classified into the second or third category requires a more comprehensive communications protocol.

Typically, in the latter two cases, operational parameters have to be written to the microprocessor systems while data and status words have to be read. It was decided to implement a software protocol in which the central computer remained master and the communication was in the form of messages between the two processors[4]. All communication was effected asynchronously to allow the microprocessor to continue operating should the computer or interface fail. The hardware link between the CAMAC system and the microprocessor Multibus was implemented using a CAMAC I/O register[5] using differential drivers to reduce noise. The concept of master-slave was broadened to include the possibility of microprocessor malfunction, the CAMAC I/O register having the necessary features to perform a hardware reset on the system.

2.3 Operating parameter entry and safeguard

Operational parameters in simple systems can be easily entered into the system by front panel switches. With more complex systems the number of parameters which must be set up is greater. Entry of parameters via the front panel is limited by available space, is time consuming and susceptible to errors. The use of a central computer to load these parameters avoids such limitations and allows parameter verification and protection against accidental modification by unqualified personnel.

It is normally required that the microprocessor system also continues operating with valid parameters after a power fail without computer or human intervention. To satisfy this requirement the parameters must be stored in a non-volatile memory, either in Electrically Programmable Read Only Memory (EPROM), Electrically Alterable Read Only Memory (EAROM) or battery backed-up memory. The disadvantage with having default parameters in EPROM is that the system can only restart at a predefined state and connot continue with the previous values. A more flexible solution is to use either battery backed-up RAM or EAROM. The power fail interrupt is then used to save the operational parameters or merely to set a power fail flag.

2.4 Maintenance

With complex monitoring and control systems being constructed by groups from various laboratories whose commitments excluded continuing maintenance, it was decided to implement a simple method of rapidly localising a hardware fault that could, if necessary, be used by staff unfamiliar with the details of the electronics. The program is written as a number of blocks, each block performing a specific function. At the start of each block, a test is made to check if the user wishes to continue or return control to a VDU. During normal operation no VDU is connected and, by default, the program execution continues. Should a fault occur a VDU is connected and each break-point may be individually enabled, thus allowing the user to step through the program verifying that specific blocks or functions have been executed. Many simple errors can be found and corrected in this manner. Should the first level intervention fail to diagnose the fault, a small resident monitor may be used in conjunction with the break-points to examine variables, processor registers etc. This second level of fault-finding requires a greater knowledge of the system and a detailed program listing.

To resolve complicated hardware faults in a microprocessor system, technicians need to be fully conversant with both the hardware and software aspects of a system. The ideal solution is that the person responsible for the eventual maintenance of the project is involved from the outset with the design. Frequently, however, this is impractical and the maintenance relies on a short period of familiarisation and on good documentation. Best use of this time can be made if the technician has a good basic understanding of the hardware and software concepts of microprocessors. This is provided by the Technical Training Service at CERN and can be backed-up by further in-depth study of the relevant documentation.

3. Examples of Microprocessor Monitoring and Control Systems in EHS

3.1 The matrix scanner

The matrix scanner[6] is built in the SPS-MPX chassis system and interfaces to the MPX bus. The function of the unit is to scan up to 512 isolated contacts and to continuously compare these states with a preloaded status memory. Any change of state is signaled to the monitoring and control computer. The use of a microprocessor in the design allowed many special features to be implemented, for example, the 128 word software FIFO that provides chronological readout of change of state. Minor modifications, found to be desirable during operation, were also made without difficulty.

The module, although internally complex, is treated by the monitoring computer as a standard MPX module using the usual MPX commands.

3.2 The data board and the expansion control projects

Each picture taken by the Rapid Cycling Bubble Chamber must be uniquely marked. Correlation of this information and the information written onto magnetic tape is essential for later analysis. The data board system[7,8] is responsible for writing the roll and frame numbers plus relevant physics data onto the film. The microprocessor system drives a bank of relays which act as shutters. After a delay to allow the relays to settle, their position is checked and a flash is triggered. Light passing through the shutters is transmitted via light guides to the film. Two members of the Rapid Cycling Bubble Chamber team from Rutherford Laboratory were responsible for the design and construction of the project and close liaison with CERN personnel ensured rapid on-site system integration.

In the Rapid Cycling Bubble Chamber the pressure change during an expansion for a given piston displacement tends to be reduced during a burst, due to the build-up of gas[9]. The expansion monitoring and control system (HOBO) forms part of a feedback loop designed to keep the pressure change for each expansion constant. The HOBO system first samples the pressure and displacement values during a burst of expansions. It then transmits the static and minimum values for both the chamber pressure and the piston displacement to the monitoring computer which computes the correction parameters. These correction parameters are then returned to the HOBO system and are subsequently transmitted, at the start of each expansion, to the expansion control electronics.

Since the common language for both projects was PL/M, the same communications package was used. The test programs, written in NODAL on a NORD 10 were consequently based on a common framework, as were the operational programs.

3.3 The ISIS gas control system

The gas monitoring and control system for the ISIS detector allows the user to open and close control valves and monitor status by issuing FORTRAN statements interpretively[10]. The user can then build up a series of statements into a working program. Once finalised, the program can be compiled and the code written into EPROMs. The interpreter is written in FORTRAN and resides in the control system in EPROM.

4. Conclusions

With the coming of LEP the number of outside groups working together in large collaborations such as EHS seems likely to increase. Equipment will be designed and built by these groups and a wide variety of CPU/board systems/languages could be expected. It is impractical, however, for CERN to attempt to support a continously expanding number of combinations. Central support would involve several people full time or a wide range of development systems would be needed. Even with the use of "universal" microprocessor development systems such as the Tektronix or Hewlett Packard which support several families of processors, the problem is far from being solved due to incompatibilities from the software aspect and the high cost of individual processor support.

The answer to this problem is the standardisation of the hardware and software used. However, recent experience in EHS has shown that this is very difficult to enforce. There are now also many valid reasons for not complying with a standard. Among these are:

i) Designers have considerable experience with a particular system.
ii) Many I/O boards have been designed.
iii) Software libraries and modules used in previous projects can be easily adapted.
iv) Large investment has been made in a specific development system.
v) Central support for brand X processor is available in the home institute.

Failure to comply with a standard brings, however, the following disadvantages:

i) To complete on-site installation and final tests, the designer will have to bring to CERN his development system or, if he has used cross software, he must install the compiler on the central computer.
ii) When the development aid returns to the home institute any further modification is impossible. At this time the program running in the microprocessor system must be considered fixed.
iii) No advantage can be taken of software routines, or of hardware, already existing at CERN.
iv) Integration into the central monitoring chain will be more difficult as CERN personnel are unfamiliar with the non-standard hardware and software.
v) Provision of future maintenance and of spare parts stock is the responsibility of the designer.

However, should the designer still choose to implement a non-standard solution, the essential aspect from the project coordination and user point of view is to avoid a proliferation of non-standard interfaces. This can only be achieved if a standard already exists and the designer is assured of support in the form of readily available hardware interfaces and reliable communication routines. Internally the choice of software and hardware of a project will continue to be the responsibility of the designer. The exterior interface must be standard

Acknowledgements

I would like to thank D.A. Jacobs, who defined the basic concepts of the computer/microprocessor communication, P. Bähler, for his assistance in integrating the microprocessor systems into the central monitoring chain and T. Lingjaerde for his advice and support.

References

1) W.W.M. Allison et al., CERN SPSC/75-15/p-42.

2) T. Lingjaerde, "The Basic Structure of the Data Acquisition and Monitoring System for the European Hybrid Spectrometer", CERN/EF/EHS/TE 77-4.

3) R. Rausch et al., "Multiplexeur de Commande et d'Acquisition Digitales et Analogiques pour le Système de Contrôle par Ordinateurs du SPS", CERN Lab. II-CO/73-2.

4) D.A. Jacobs, "Communication Protocol between EHS-MON N10 and I8080 based Microprocessors", EHS Software Notes 02, November 1978.

5) Hytec 440-S-4 Input/Output Register, Hytec Electronics, Court Road, Maidenhead, Berkshire SL6 8LQ, England.

6) P. Bähler, N. Bosco, R.A. McLaren, "An MPX Matrix Scanner", DD Report in publication.

7) J.L. Benichou, D. French, A. Hervé, G. Merighi, "Data Board for RCBC Cameras", CERN/EF/EHS/TE 78-6.

8) J. Alsford, J.F. Ayres, R.A. McLaren, "Microprocessor Control of Film Marking for the European Hybrid Spectrometer Bubble Chamber Experiments at CERN", DD Report in publication.

9) A. Hervé, "Results of Expansion Tests for a Rapid Cycling Bubble Chamber using a Forced Hydraulic Servo-system", CERN/DPH.II/RCBC 75-26

10) A. Romaya, "FORTRAN Subset Interpreter for a Microprocessor Gas Control System", Proc. of the Europhysics Conference on Computing in High Energy and Nuclear Physics, 9-12 September 1980, Bologna, Italy.