EXPERIENCE WITH THE CONTROL SYSTEM FOR THE SPS

M.C. Crowley-Milling
Proprité littéraire et scientifique réservée pour
tous les pays du monde. Ce document ne peut
être reproduit ou traduit en tout ou en partie
sans l’autorisation écrite du Directeur général
du CERN, titulaire du droit d’auteur. Dans
les cas appropriés, et s’il s’agit d’utiliser le
document à des fins non commerciales, cette
autorisation sera volontiers accordée.
Le CERN ne revendique pas la propriété des
inventions brevetables et dessins ou modèles
susceptibles de dépôt qui pourraient être décrits
dans le présent document; ceux-ci peuvent être
librement utilisés par les instituts de recherche,
les industriels et autres intéressés. Cependant,
le CERN se réserve le droit de s’opposer à
toute revendication qu’un usager pourrait faire
de la propriété scientifique ou industrielle de
toute invention et tout dessin ou modèle dé-
crits dans le présent document.

© Copyright CERN, Genève, 1978

Literary and scientific copyrights reserved in
all countries of the world. This report, or
any part of it, may not be reprinted or trans-
lated without written permission of the
copyright holder, the Director-General of
CERN. However, permission will be freely
granted for appropriate non-commercial use.
If any patentable invention or registrable
design is described in the report, CERN makes
no claim to property rights in it but offers it
for the free use of research institutions, manu-
facturers and others. CERN, however, may
oppose any attempt by a user to claim any
proprietary or patent rights in such inventions
or designs as may be described in the present
document.
ABSTRACT

The design of the multicomputer control system for the 400 GeV Super Proton Synchrotron (SPS) at CERN was described in the report CERN 75-20, issued in 1975, before the commissioning of the accelerator. The present report, which should be read in conjunction with the earlier one, describes the modifications made to the system in the light of experience, and how it has adapted to changing requirements. Reliability of the system and of its components is discussed. Taking into account modern developments of microprocessors, etc., the changes that might be made if the system were to be redesigned are examined. Finally, the application of the design philosophy to other fields is discussed briefly.
## CONTENTS

1. INTRODUCTION

2. CHANGES SINCE THE PREVIOUS REPORT CERN 75-20
   2.1 The hardware system
   2.2 The computer system
   2.3 The message-transfer system
   2.4 The timing system
   2.5 The software system
   2.6 The command language
   2.7 Data modules
   2.8 The operating system
   2.9 The operator interface
   2.10 Applications software
   2.11 Protection
   2.12 The alarm system

3. EXPERIENCE WITH THE SYSTEM

4. RELIABILITY
   4.1 Computers and peripheral equipment
   4.2 The message-transfer system
   4.3 The equipment interface
   4.4 The software system

5. CONSIDERATIONS FOR THE FUTURE
   5.1 The programming language
   5.2 Data modules
   5.3 General-purpose consoles
   5.4 The computer network
   5.5 The interface system
   5.6 Autonomous operation
   5.7 Critical timing
   5.8 The impact of microprocessors
      5.8.1 Autonomous function controller
      5.8.2 Graphite display generator
      5.8.3 Function generators
      5.8.4 Data acquisition
      5.8.5 Local servo loops
      5.8.6 Conclusions
   5.9 Application to other purposes

Page
1
1
2
5
8
8
9
10
11
13
18
19
21
22
24
26
27
28
29
30
31
32
32
33
34
34
35
35
35
36
36
36
37
37
38
6. ORGANIZATION AND ACKNOWLEDGEMENTS

REFERENCES

BIBLIOGRAPHY
1. **INTRODUCTION**

The report CERN 75-20 "The design of the control system for the SPS" was issued at the time when the system was just coming into operation, but before the accelerator had been commissioned. Only a very preliminary assessment of the performance could be made, and it was stated that a further report to give an "assessment of the performance of the control system, the difficulties and failures experienced, and changes that seem desirable in the light of experience, will be issued when sufficient evidence is available".

The commissioning of the accelerator took place between May and November 1976 and it has been supplying beams for physics experiments on a regular schedule since the beginning of 1977. Both the commissioning period and the first year of operation, with their differing requirements, have given sufficient experience of the operation of the control system to make it worth while to issue this further report.

The over-all assessment of the quality of a control system must be to some extent subjective. Individual components can be looked at objectively -- do they meet their specifications or have any unexpected limitations shown up? But the answer to the question "Does the control system provide the services needed in the most convenient and satisfactory way?" may well be answered differently by individuals with differing previous experience or prejudices. Therefore this report will deal as much as possible with the objective part of the assessment, and the question posed must be left for others less involved in the design to answer.

The section that follows this introduction brings up to date the changes and developments that have taken place in the hardware and software since the previous report was issued and gives the main reasons for these changes. To avoid repetition, it is assumed that the reader has a copy of the previous report, and Subsections 2.1 to 2.11 correspond to Sections 6 to 16 of CERN 75-20, with identical titles, to facilitate cross-reference.

Further sections discuss the operational experience with the system, the reliability, possible future developments, and what lessons may be learned from the SPS experience that might be useful for other applications.

It should be emphasized that a control system for an accelerator like the SPS never reaches a final state, as might be expected with a normal process control system. The demands on the accelerator are always changing, and these may require developments in the hardware and software to enable them to be met, or for limitations in the existing equipment to be overcome. The basic conceptual simplicity of the SPS system has so far enabled, and hopefully will in the future continue to enable, these changes and developments to be incorporated with the minimum of disturbance.

2. **CHANGES SINCE THE PREVIOUS REPORT CERN 75-20**

2.1 The hardware system

The reasons for using the CAMAC standard for the primary interface to the computers, with the addition of a special serial multiplex (MPX) system to control the majority of the equipment, have been given in the previous report.

To save time in the initial design, the original MPX controllers were separate units which were connected to three standard CAMAC modules to provide the interface to the computer.
When the long-distance version of the controllers was designed, the opportunity was taken to make this in the form of a triple-width unit to plug directly into a CAMAC crate, thus saving the intermediate modules. This is now the standard controller for all extensions and additional installations, as it can be used both for short-range (~200 m) and for long-range use. In the latter case, the maximum range is virtually unlimited if repeaters are inserted at intervals of not more than 1.5 km, using the standard data-link cable and the latest type of repeater developed for the data links. This new controller operates in three modes: programmed input/output, direct memory access and automatic surveillance. Interrupt facilities can be provided, in conjunction with a special station module, but are not normally required.

Most of the CAMAC modules used were standard, but a few special ones were needed. These include an Autonomous Function Controller, designed to fit into the CAMAC crate, which, together with a plug-in memory module, allows a number of data to be acquired and pretreated, either directly by CAMAC modules, or indirectly through the MPX system, without involvement of the computer after the initialization. This facility is used mainly for the beam instrumentation, where it is necessary to acquire a large number of data rapidly at a given point in the cycle. The original design used discrete logic, but the latest model uses a microprocessor.

While serial CAMAC was not available at the time of the design of the accelerator control system, the standards had been agreed and some prototype equipment was available when the control for the secondary beam lines was being considered. Since the requirements for the instrumentation of these lines are very similar to those for experiments, and suitable standard CAMAC modules were available, it was decided to use serial CAMAC for this purpose. After initial difficulties, only to be expected since this was one of the first large-scale applications of the new system, satisfactory operation was obtained by the time beams were becoming available for the first experiments.

The introduction of microprocessors into the SPS system without major disturbance will be facilitated by the agreement on a standard bus system and hardware, known as the CIMBUS, which can accommodate a variety of different microprocessors. As far as the MPX system is concerned, the CIMBUS crate looks the same as a standard station crate, and so can be allocated up to 128 addresses at the data-module level, without the need to introduce any additional subaddressing.

The reliability of the interface equipment is discussed in Section 4.

2.2 The computer system

The 24-computer network, illustrated in the previous report, had already undergone some changes by the time the report was issued, and since then there have been a number of changes and additions to the system in the light of experience.

The first modification was to combine the two computers in BA6 for extraction and beam line to the West Area, since it became evident that the beam-line computer would be under-loaded, and there was some duplication between the two requirements. On the other hand, the computer in the power-supply building in the West Area, which was supposed to deal with the second half of the primary beam line and the secondary beam lines, and to provide services to the experimenters, was soon shown to be overloaded, and so the load was shared
between two computers. Corresponding changes were made in the North Area, resulting in a net increase of one computer.

The next change was triggered by difficulties which were being experienced by failures in the data-link repeaters (see Section 4). If the only links between the remote computers and the control centre are the data links, and the data links fail, it is clearly impossible to find the cause of the failure without going to each of the buildings on the data-link route and performing quite complicated tests, or replacing the repeaters in turn by ones known to be working satisfactorily. The development of the long-distance version of the MPX system opened up the possibility of performing the latter operation automatically, by having a computer at the centre with long-distance MPX links to each building in which data-link repeaters are installed, to allow duplicate repeaters to be switched in remotely by suitable MPX modules. This exchange can be made for each repeater in the data link in turn, until the fault disappears.

This facility for changing repeaters would not by itself have justified the addition of a computer to the system, but there were two other requirements. The computer facilities for the personnel access system had up to then been provided by the Alarm computer, which relied on the network and the general-purpose computers to provide the connections to the equipment. The only possible time to make any modifications to the computer system is when the accelerator is shut down, and this is the time when the activity for the personnel access system is likely to be a maximum. If the access system needed only a single computer for its operation, it would only be necessary to keep that computer in operation during shut-downs, instead of the whole network. The same argument applies to a lesser degree to the acquisition of information from the radiation monitoring system.

These three requirements seemed to justify the addition of an Access computer, with long-range MPX links to the auxiliary buildings, connecting to the access, data-link switching, and radiation-monitoring systems. This is now being implemented.

It will be noted that the addition of these long-range MPX links goes against the simple concept of a singly-connected system, in which all equipment in a building is interfaced to one or more "local" computers and these provide the only connection to the centre. However, the provision of facilities for remote diagnosis calls for some duplication, and this is the simplest way of providing it, without the complication of multiply-connected networks.

A block diagram of the arrangement resulting from these changes is shown in Fig. 1.

Further changes involved the adoption of a two-node network, as described in Subsection 2.3, to assist maintenance and provide for a rapid recovery from hardware faults, rather than from traffic considerations, since the measurements on the message-transfer system show that there is plenty in hand at the moment.

It should be pointed out that it has been possible to make the changes described above -- the addition of a computer here, the combining of two computers into one there -- without major upsets because of the original simplifying decision: that all intercomputer communication should be in the form of interpretable statements or data. Naturally, changes have had to be made in some of the programs, but in many cases these have been limited to changing the computer number assignments.
Another change is becoming necessary for two reasons, which are somewhat connected. As shown in Subsection 4.1, the drums have been becoming increasingly unreliable and, although Norsk Data continue to provide some maintenance service, they no longer support them as a standard product, having changed to moving-head discs for their latest systems. It is considered that the conditions in the auxiliary buildings are not suitable for moving-head devices and, since the mass storage requirement for the general-purpose (GP) computers is relatively modest, it has been decided to replace the drums by semiconductor memory, which has become cheap enough for this change to be made at reasonable cost. This is done by replacing the core memory of the existing computers by semiconductor memory, large enough to serve both purposes. This new memory is of the self-correcting type, which can correct for an error of one bit in a word, and signals errors of more than one bit. In addition to eliminating the drum problems, this change will give a significant gain in speed of operation of the computers. This is because, with the existing system, most of the data modules are on "core-loads", which are swapped between the drum and a reserved area in the main memory as required. This swapping process can take up to 40 msec, and so can be a serious limitation on the number of operations that can be carried out within one cycle of the accelerator. In the replacement system, all the data modules will be stored in the main memory and a call to a data module will only involve switching pointers, taking a few microseconds.

Other changes include the addition of a fourth console computer and the gradual change-over from paper tape to floppy discs for reloading the drums and stand-alone systems. Also, it had not been realized in the original planning that there would be a continuing need for several additional computers, not directly part of the control system, after the accelerator had become operational. It had been planned to keep the time-sharing system, as this is
essential for the development and assembly or compiling of the programs that are not written in NODAL, but it has been found necessary to add others for hardware and software testing. During the construction period, this work was done using computers awaiting installation in the auxiliary buildings. However, this work did not stop after all the computers were installed, since there are continual minor changes and improvements to both the hardware and the software, and these have to be tested under realistic conditions before they can be installed in the control system. To take care of the differing requirements, a total of six additional computers have been acquired, including the mobile spare mentioned in Subsection 4.1, and a fixed spare in the main control building. These computers have links to the message-transfer system, to allow access to the library for the transfer of files. These links are normally closed and, contrary to the case for normal links, an alarm is given in the control room when these links are open, although in fact no interference with the operation of the accelerator has so far been experienced from these "parasitic" computers.

2.3 The message-transfer system

As was pointed out in the previous report, the most difficult task in the design of the message-transfer system was to make an estimate of the inter-computer traffic to be expected, before the requirements for the control of the accelerator as a whole were known. A number of different situations were examined and the resultant traffic assessed. It soon became clear that the number and length of the messages were very dependent on the strategy adopted for the reporting of fault and alarm conditions. Since no solution to the problem of alarm analysis seemed to have been found elsewhere at that time, the specification for the message-transfer system was drawn up to cover the worst-case estimation, with something in hand.

Measurements made soon after commissioning showed that the system easily met the specification, but only recently has it been possible to make measurements on the speed and traffic flow under operational conditions.

A report has been issued[2], which gives a description of the principle and protocol of the message-transfer system, and the results of the speed and traffic measurements.

The specification called for a minimum transfer rate of 10,000 sixteen-bit words/sec for a long file from a process in one computer to a process in another computer, with a total traffic in the message-handling computer of 30,000 words/sec. It should be possible to transfer a short file (32 words) in less than 5 msec including all overheads.

In practice, the maximum transfer rate on a link is 21.4K words/sec, and the capacity of the message-handling computer is 43K words/sec, both figures being for the 64-word data-block size chosen. The total transfer time for a 32-word block is 4.07 msec. Thus, it can be seen that the system developed comfortably surpasses the specification in all respects.

Traffic measurements were made for four different phases of the machine operation: start-up, normal day operation, normal night operation, and "Monday morning" operation. The last is likely to be a busy time, since it is then normal for the people responsible for the hardware to call in at the control room to run their diagnostic and statistics programs.

Several interesting results came from these measurements. The first is that the major activity on the network is its self-surveillance. Each link is checked periodically by the
message-handling computer (MHC), which sends a message down the link if it has not been active for a given period, usually one second. The message must be echoed back correctly, otherwise the message-handling computer closes the link and raises an alarm. In order to avoid confusing the measurements of the real traffic, the surveillance was inhibited during the period of these measurements.

The second result is that the average activity uses only about 5% of the capacity of the message-handling computer, when the surveillance is suppressed, and the variation between the phases is not as great as might be expected, the message-handling computer utilization averaging from 3.5% for the night shift to 5% for the Monday morning shift. The peak utilization, in a 100 msec period, varied from 10% to just over 20%.

Another surprise was that the activity was relatively evenly spread out over the SPS cycle. One of the worries in the design stage was that the requirements to call and run programs might bunch at certain times in the cycle, and it was thought that it might be necessary to restrict some less important programs to run in the "quiet" part of the cycle. Only the "Monday morning" phase showed any great variation during the cycle, where the traffic just after injection was significantly higher, as can be seen in Fig. 2.

When the traffic to and from the individual computers was examined, it was found that, as could be expected, the Library computer was the main customer, receiving about 10% and originating over 30% of all traffic, since it provides the mass-storage facility for all computers without discs or drums. The next greatest contributor was the Display computer, with about 12% in and 15% out, followed by the Alarm computer and the Consoles. At the other end, the RF computer demonstrated the characteristic aloofness of RF systems in accelerators by contributing about 0.1% to the traffic!

![Graph showing CPU activity over time](image)

**Fig. 2** Message-handling computer activity during the SPS cycle (period 06.00 to 12.00 Monday morning)
All these measurements show that the message-transfer system is relatively lightly loaded, and has a substantial reserve capacity. This has been obtained by keeping to a rather simple protocol, minimizing the number of acknowledgements that have to be made, and relying on time-outs to resolve blocking situations.

Before these measurements were made, there was some worry about the rapid increase of traffic between the computers controlling the primary and secondary beam lines and providing facilities for the experimenters, and it was decided to modify the message-handling software to allow the use of a system with more than one node. Although the use of more than one node has been shown to be unnecessary from the point of view of capacity, the system has been reconfigured into a two-node system for a quite different reason. The maximum capacity of 32 computers on one node is obtained by using four groups of eight Direct Memory Access (DMA) controllers for the data links. In the event of a failure in this rather critical electronic equipment, the quickest and simplest recovery procedure is to exchange a complete group. If the number of computers on a node does not exceed 24, then one complete group can be kept as an installed spare. Recovery from failure only requires the change-over of some plug-in cables, and the reloading of a table in the message-handling computer.

The two-node arrangement shown in Fig. 3 has now been in operation since July 1978, and the system could be expanded to take in further nodes, in the future, if, for example, it proves desirable to interconnect the CPS and the SPS control systems.

---

**Fig. 3**  Computer-system layout after addition of second message-transfer system.
2.4 The timing system

There have been relatively few changes to the timing system from the description in the previous report. The double cycle, for which provision was made, has not been used, since the intensity requirements at 200 GeV have been relatively modest, and a greater overall efficiency has been achieved with a single cycle with two flat tops.

However, the system of "events" has proved to be one of the most powerful features of the timing system. Some of these events are programmed to occur at a certain time in the cycle, and the timing of these can be altered by loading a new table into the Master Timing Generator. Others are generated for events whose timing may vary slightly from cycle to cycle. For example, the extraction system may be pulsed at the same time each cycle, but slow extraction is a resonant process, the build-up time being dependent on the emittance of the beam in the SPS, which varies slightly from cycle to cycle. However, a signal from a beam monitor in the extraction line can be used to trigger the event "start of slow extraction".

The total number of events has increased from 23 to 35, as the demand from the users has increased.

In addition to the original timing modules, which give programmable selection of the event and time after that event, some modules giving six independent outputs, hardwired to selected events, have been produced. These are used where it is important that there can be no program interference in the timing, for reasons of equipment safety, and also for certain applications, where immediate response on restoration after a mains failure is more important than the ability to vary the timing.

A duplicate Master Timing Generator has been installed, as a failure of this unit would stop the whole SPS. A change of cycle is loaded into both units, and the failure of one unit causes the other to take over automatically.

2.5 The software system

The reasons for basing the software system on having an interpreter resident in each computer, and restricting intercomputer communication to interpretable statements and data are discussed in the earlier report. In practice, they have been shown to be fully justified. All the advantages that were expected have been achieved, and few of the disadvantages have caused any difficulty. The main disadvantage, which is the main point of discussion each time the system is considered for other applications, is the slow speed of operation of an interpreter. It is true that to interpret a certain piece of code takes many times longer than to run the compiled equivalent, but that is by no means the whole story in a real-time control situation. There are many other things that limit the speed with which a program can run; for example, analogue acquisition and conversion to digital form with sufficient accuracy and hum rejection can take from 20 to 200 msec, and valves or other motor-operated devices can take several seconds to come to the required position. In addition, there are other delays in the operating system unconnected with the use of an interpreter; for example, core-load swapping can take up to 80 msec if all the data base cannot be kept permanently in core and readback is necessary. Thus, the additional delay caused by interpretation can be relatively small. Also, many of the operations in the control of equipment like an accelerator are tied to the reaction time of a human being,
and the interpreter is still fast compared with that. Even for complicated acquisitions, computations and display production, which may take a few seconds, the operator does not feel frustrated if he is given some indication on the screen that his command is being carried out. The greatest delays, which have nothing to do with the speed of the interpreter, are when actions have to wait for a particular time in the cycle, and then it is of the greatest importance to make sure that the operator is not left looking at a blank screen during this wait.

Where there is a genuine requirement for increased speed, this can often be provided by writing functions to carry out some calculations which would take too long in NODAL. Such functions can be pre-compiled and then called by the NODAL programs when needed. In other cases, the required effect can be obtained by writing more complicated data modules, with additional properties, to perform the additional computations. The question of speed is discussed further in Section 3.

It may be that for some future control systems the speed of an interpreter may be the limiting factor, and, as a long-term project, the possibility of speeding up the interpreter is being investigated. This investigation is looking into two possibilities: the compilation of individual NODAL statements and the development of hardware to perform the required list searching at a much higher speed than is presently achieved in software. The part-compilation, which retains the structure of the source program and the interactive debugging and editing which is one of NODAL's great advantages, has been christened Nodiling, and a first version of a "Nodiler" has been written and tested successfully, increasing the speed of execution of a typical NODAL line by a factor of about five.

2.6 The command language

The original design of the language NODAL has provided most of the facilities needed for the SPS control system, and relatively few changes and additions have had to be made as the result of experience. The number of functions available has continually increased, and they now total over 200, with 170 available in the console computers.

It was not realized, until pointed out at a recent conference, that NODAL seems to be the only language available at present that has provision for true parallel computation in a multicomputer system. Since the programmer can determine in which computer he wishes a given statement to be carried out, he can command a number of different actions to be carried out in different computers at the same time, meanwhile continuing with the main program in the computer he is in direct contact with, and he is provided with the necessary commands to acquire and synchronize the results remitted from the other computers.

This is the result of statements such as:

```
FOR I = 1, 6; EXECUTE (I=M) I=N

:  

FOR I = 1, 6; WAIT (I=M).
```

This calls for the execution of six different groups of the main program in the six computers numbered (I=M), and then, after carrying out some other statements, waiting for the results from the other computers, so they can be used in the remainder of the program.
The only restrictions on the groups sent to the other computers is that the code must be executable in the computer concerned, and the group or groups must end with a REMIT statement to return the results to the computer calling for the EXECUTE. Even if these results arrive before the WAIT statement, they can have no effect on the running program until the WAIT is reached.

NODAL is now used in some of the NORD-10 computers used for control and data acquisition for experiments, using the operating system SINTRAN III, and it will be used for the new PS control system. It is also used for the control of the new electron-positron storage ring being built at DESY, Hamburg, but there the name has been changed to POCAL, without, as far as is known, any other changes. A language based on NODAL, but with a few changes, which can be interpreted or compiled is being developed by GEC for their 4080 computer, and is being called GRACES.

NODAL, without the string-handling facilities, is also available on the PDP-11 series of computers. A new NODAL manual has just been issued\(3\).

2.7 Data modules

The original concept of the data module was a rather simple handler and data table for a basic type of apparatus, with the necessary properties accessible through NODAL to perform single control and acquisition actions on the hardware or the data table; all combinations of such actions were to be carried out in NODAL by multiple calls to the data module(s). However, there has been a tendency to do more and more in the data modules, mainly in the case of those for the beam instrumentation system. Here there are the special problems of acquiring data from a large number of devices at a given time in the cycle. The data module is accessed first with one or more properties to set up the CAMAC equipment, to capture the data on the appropriate timing pulse, and to read the data array into the computer memory by DMA immediately afterwards. A real-time program is then scheduled to acquire this array, perform any necessary manipulations, and write it into the data table. The latter can then be read through another property of the data module, when required by the main program. Complication of the data module can also simplify and speed up the NODAL programs. For example, if a surveillance program is required to check whether a power supply current is within tolerance, three calls to the data module have to be made, to obtain the requested value, the actual value, and the tolerance, so that a comparison can be made. If the comparison can be made in the data module by calling an extra property, which returns a "yes" or "no" answer, the surveillance program is only required to acquire the values if the answer is "no".

This technique of complicating the data modules should only be applied where absolutely necessary, for several reasons. Firstly, it is a means of introducing assembly-language applications programs into the system, with all the disadvantages discussed in the previous report. Secondly, the more complicated the data modules become, the greater the requirement for modification as the accelerator is developed. Since assembly-language modules should only be assembled into the system after very complete testing to show that they cause no untoward effects elsewhere, changes can only be made at rather infrequent intervals. Thirdly, the simpler the data module, the easier it is for someone else to understand it, when the original writer is no longer available.
The number of data modules was originally restricted to 64, as it was thought this would be amply sufficient, and it enabled each individual piece of equipment to be identified by a sixteen-bit word, the first six bits giving the data module number, and the other ten bits allowing for up to 1024 units of the same type. The total number of data modules has been kept below 64, but there have been additional requirements for handlers for single pieces of equipment which were originally satisfied by using the general-purpose function mechanism provided by NODAL, so they were called "equipment functions".

Later it became convenient to amalgamate the equipment functions with the data modules, to use the same support routines. Since the equipment number identifier did not find general use, the limit on total numbers could be overcome by using the six-character name of the data module or equipment function as the identifier. Another advantage of this change is that the CPS control system is being rebuilt, adopting NODAL and the data-module concept, using the same message-transfer system, and so the future possibility of coupling the two systems is not limited by the number of data-module identifiers.

Because of the time element, the data modules were written by a number of people, some having no previous experience of assembly-language programming. This resulted in a variety of styles, especially as the concept was new, and no clear guidelines could be laid down until some experience had been gained. Since the initial commissioning of the accelerator, a considerable amount of effort has been put into providing guidelines and appropriate subroutines, and bringing the data modules up to date.

2.8 The operating system

The operating system for all computers consists of the basic SYNTRON II, a considerably modified version of the computer manufacturer's SINTRAN II, as described in the previous report, together with the NODAL interpreter and the message-transfer package. Originally, SINTRAN III was used for the Service computer, since it provided the extra facilities needed, such as time-sharing terminals for program development, a driver for magnetic tapes, and for the possible future requirement to have more than 64K words of memory. However, a number of difficulties were experienced owing to the incompatibilities between the two systems, which could not be resolved simply. For example, SINTRAN III can only have one real-time task active at a time, leading to delays in responding to the message-transfer system and consequent time-outs.

To overcome this difficulty, the facilities of SYNTRON II were gradually extended until it could provide all those required for the Service computer. A version with multiple time-sharing NODAL buffers at the interactive level had already been produced for the control computers in the Experimental Areas, where it was required to give the individual experimental teams control over some of the elements in their secondary beam lines. The addition of the memory paging from SINTRAN III, together with the necessary drivers, gave a version of SYNTRON II with sufficient facilities to carry out all the tasks required from the Service computer for the operation of the control system. It does not provide for assembly, compilation or the building up of systems, as these facilities are provided by a SINTRAN III system running on a separate laboratory computer (TSS). This computer is connected into the control system network for the purpose of transferring files only.
There have been other modifications to SYNTRON II, the main one being an improved library file system. This cuts the time taken by the Library computer for file operations to about one-fifth of that taken previously, and this has eliminated the delays that were beginning to occur at times of peak activity.

The development of SINTRAN illustrates the quandary facing a control system designer; whether to stay with a manufacturer's operating system when it changes and enjoy their full support, even if it results in a non-optimum system, or to retain and possibly modify an obsolete system, which the manufacturer may no longer support, but which gives the facilities required. When the SPS control system was being designed, Norsk Data were offering SINTRAN II, a small, fast executive with restricted facilities, and a separate time-sharing system which provided all the facilities required for programming. These two have been replaced by SINTRAN III which performs both duties, but inevitably, as it covers so many requirements, it is slower than SINTRAN II and requires more main memory plus a moving-head disc. The latter requirement rules it out for use in the auxiliary plant buildings of the SPS, where the necessary dust-free conditions cannot be guaranteed. Thus, in the case of the SPS the answer was clear; continue with the modified SINTRAN II system, as it suited the requirements and the systems programming effort is available to maintain it and make any changes that might be required in the future. For systems where such an effort is not available, one may be forced to stick with the manufacturers standard system, and follow its development, even if the "progress" is in the wrong direction for the particular application. In future the most important parameter in the choice of computer for a control system may well be the properties of the executive available, and the policy of the manufacturer for its further development or replacement.

With the SYNTRON II system in most of the computers, the interpreter has four working areas at different priorities. At the highest level is a small buffer to run "Immediate Execute" (IMEX) calls from other computers. These messages have a maximum size of one block (64 words), and are intended for use only when the highest possible speed of response is needed, or to free blocking situations, such as an endless loop in a program of lower priority. Next in priority is the "Execute" buffer, used for running programs sent from other computers, which can be up to 2K words long, but are normally much shorter. Below this is the "Interactive" buffer, at the disposal of the local teletype or VDU, and at the lowest level is the "Surveillance" buffer, in which run the surveillance and local logging programs. In the time-sharing versions used for the Experimental Area and Service computers, there can be up to 10 resident interactive buffers, the number being limited by the memory size, but these can be dynamically allocated between up to 20 terminals. The computer consoles have a single large (6.5K words) interactive buffer and a "Defined Function" area into which may be loaded special functions to be used by the NODAL programs running interactively.

As well as providing for the running of a number of applications programs, at different priorities, by the re-entrant NODAL interpreter, the current SYNTRON II system allows for the scheduling of programs both in NODAL and machine code according to time or interrupt, provides for the reception and transmission of messages, for the management of the dynamic core and the peripheral handlers, and has comprehensive file-handling and storage facilities. The SYNTRON II system occupies 8K words and the NODAL interpreter 10K words, with all string functions.
2.9 The operator interface

The concept, described in the previous report, of having completely general-purpose control consoles, with only a few interaction devices, was new to most of the engineers and physicists who had designed the accelerator equipment and were called on to write the programs to commission and control it. There was some initial resistance from those who were accustomed to identifying a particular control or display by its position on a control desk or rack, but once some experience of the power of the system had been gained, there only remained some pressure for a few dedicated displays which will be described later.

Some guidelines had been laid down and a few simple simulation programs had been written by the Controls Group to illustrate methods of using the interaction devices, but the equipment designers were left free to use them in the way they thought would best serve their purposes. It would have been foolish to have laid down too rigid rules from the start with such a new system, since the ideas as to the best way to control an accelerator evolve only with experience. This approach has led to some unsatisfactory programs which have had to be rewritten, but the gain has more than offset such a loss. The usual pattern is that an overview of the situation concerning a type of equipment is available on touching a succession of buttons on the touch-screen and that control of a particular item is obtained by further touch-buttons, with a final selection using the rolling-ball to position the cursor to the appropriate point on the display screen. The actual control may be by means of the touch-screen, for on-off actions, or by means of the knob, for variables. In a small minority of cases, preference is shown for using the keyboard to input a digital value.

Doubts have been expressed as to whether the use of colour displays gives any real control advantage, or whether they just look more attractive. For example, in a recent conference, a discussion on air-traffic control came to the general conclusion that anything that could be done by colour could also be done by symbols on a black-and-white screen, and one contributor stated that colour was so valuable a variable it ought to be reserved for some future unspecified use! In the SPS system colour plays a very significant part in informing the operator about the status of values he is reading from a display. If the colour is white, the value has been obtained from the hardware, and represents the present setting. If the operator wishes to change the setting by means of the knob, he makes the selection, using the touch-screen and cursor, and then the value is repeated in green alongside the white figures. If he then turns the knob, the green value will change, green representing the value being set by the operator, but not changing the hardware. If he tries to set the value outside a preset range, the green figures will change to red. When he has set the new desired value, the operator presses the interrupt button alongside the knob, and this new value is sent to the hardware. The green figures disappear and the original white figures change to the new value, but in magenta, to indicate a transition. When the value in the hardware has been changed, and confirmed, the figures become white again. This use of colour to show actual, setting, transitional, and out-of-limits values occupies less of the display screen than if labels had to be attached to all values to show the status, and so allows more information to be provided in a display, without confusion.

The experience with the SPS has shown that colour does make a significant contribution to the ease of control, if used in a standardized way, and that the TV raster display,
using shadow-mask tubes, is the simplest and cheapest method of providing colour display of adequate resolution. A large range of colours is not needed, as the different colours must stand out clearly, even on a badly adjusted or dying monitor, if colour is to be used as a real display parameter and not just to make a pretty picture. Red, green, magenta, and white are available from the character generators, which are used for both alphanumeric strings and simple mimic diagrams, where colour is used as a status indicator as explained above. As described in the earlier report, graphics displays were originally provided by scan converters. Because of the cost of these and the need for regular maintenance, only a limited number of scan converters was provided, in a pool which was shared between the three consoles. It was intended that normally graphs would be in a single colour, but these could be combined with multicolour characters in a single display. It was possible to use more than one colour to distinguish between different graphs on the same display, but that meant reserving more than one scan converter from the pool. A scan converter could be assigned to any one of the colour monitor inputs, red, green or blue. The blue gives a poor contrast and is hardly ever used. Green is the most used colour for graphic display, with the addition of red where it is needed to distinguish between two states.

Owing to the rapid deterioration of the scan converter tubes, and the need for frequent adjustments, these have been replaced by a dot-matrix system using semiconductor memory and a microprocessor. This was developed at CERN and is now manufactured commercially. Such a device would have been far too expensive and complicated using the technology available when the SPS control system was designed, but the recent advances in large-scale integrated circuits have brought the cost and the complexity down to the level where the use of dedicated graphics generators on each console could be considered, instead of the pooled system. This change has been made recently, providing red and green only and, as well as simplifying the switching, it has the added advantage of taking a considerable amount of load off the Display computer, which had become one of the bottle-necks when all three consoles were in use.

One restriction on the use of colour is the need for "hard copy". On the SPS both electrostatic plotters and TV raster-scan copiers have been used, and these both give black and white images, so any information conveyed by the colour is lost. The raster-scan copier is the more convenient to use, and so has been adopted as the standard.

Turning to the interaction devices, the touch-screen has proved to be extremely reliable, and the interest shown in it for this and other applications has resulted in it being marketed by more than one firm. A maximum of sixteen buttons has proved to be adequate, and for most choices a smaller number is used. In such cases, it is desirable to make the button layout different for successive choices, since the operator soon learns to recognize the patterns, and a frequently used sequence can be gone through without having to stop to read the legends on the buttons.

The facilities of the multipurpose knob have not been used as widely as was expected; it is used mainly as a simple incremental encoder. The sixteen-position switch is used by a few programs, but this facility can be equally well provided by the touch buttons. The spring-loaded centre-zero mode is little used and the variable brake only in demonstration programs, but its ability to provide the equivalent of mechanical end-stops finds some operational uses. The rolling-ball is used almost invariably to move a cursor on the display screen. This cursor is displayed in yellow, to distinguish it from other elements of
the display, and can take a number of forms under program control. The most popular forms include a small solid box for selection on a mimic diagram, a horizontal line for selection of a line of text, and cross-hairs for curve modification.

Only brief mention was made in the previous report of the requirements for the observation of analogue wave-forms on normal or storage oscilloscopes. There are approximately 1500 such signals around the SPS, and ideally one would like to be able to select any six for display on the two oscilloscopes on a console. However, this would have required a bigger switching matrix than seemed reasonable, and so the signals were grouped in the auxiliary buildings so that they could be switched onto relatively few lines going to the control room, in such a manner that signals which are required simultaneously are on different lines. A full matrix switch is used at the control room so that any incoming line can be put on any monitor. The bandwidth required is greater than 100 kHz. The switching is controlled from a separate touch-button screen on the console. Initially the selection program ran in the Console computer, which then passed on the switching instructions to the Display computer. Since trace switching involved interrupting the main console program, which caused some inconvenience, the whole operation will in future be carried out by the Display computer, now that the duties of the latter have been eased.

The main inconvenience with having completely general purpose consoles is that it is necessary to tie up some of the resources if it is desired to have a continuous display of some given parameters irrespective of what the console is being used for. It was recognized from the start that such "comfort" displays would be required -- displays which were always in the same position so that the operator could be assured that the accelerator was running correctly at a glance. It was originally planned that such a display should be projected onto the wall of the control room, where it would be visible from any console. However, it turned out that none of the projection systems then available would be satisfactory; they were either not bright enough, or were too expensive and required too much maintenance. This situation is still true today, and large wall-mounted TV monitors are used. In addition, there are a number of displays requiring special transmission techniques, such as low-loss coaxial lines, or transient recorders with TV raster read-out, which cannot be included in the normal analogue wave-form system. The number of these could not be estimated before experience with operating the accelerator had been gained, but one panel on each of the control consoles had been reserved for such displays. At one time, the number of these displays required looked likely to exceed the space available and so when the layout of the control room was changed, as described below, the opportunity was taken to provide a separate desk for this purpose. The number of permanent displays has not grown as rapidly as was feared, but nevertheless it is convenient to have them together on the one desk.

The original layout, shown in Fig. 4a, was chosen following the assumption that the engineer-in-charge would use the central console, and that the top of the console would be low enough for him to oversee the activities of operators working at the other two consoles. However, owing to the size of the available monitors, etc., and the spacing needed between them, the height of the console increased so that only a tall man could see over it when seated, and even his vision was impeded when books, etc., were placed on top of the console, as inevitably happened. Also, it was found convenient for two operators to be able to converse when setting up or making adjustments to different parts of the same system; for
example, if one is adjusting the extraction system and the other a beam line in order to
set up the accelerator as rapidly as possible, co-operation is needed so that the activities
of one do not frustrate the other.

A third reason for changing the control-room layout was that the need for a fourth
control console had become evident. This was mainly because of the continuing pressure to
improve, extend, and modify the applications programs, partly for operational reasons, and partly because the equipment designers saw improved or more elegant ways to control their equipment. Any console not being used for operation of the accelerator would normally be occupied by someone modifying a program. While a considerable amount of the program writing and modification can be carried out on the terminals of the time-sharing system which had been set up in the laboratory and office block from the start, a console is necessary for testing and debugging if displays are involved, which is the case for most applications programs. In addition, it is intended that the secondary beams in both the experimental areas will be set up and controlled from the main control room, rather than the local control rooms used for the initial operation, in which case it was estimated that four consoles could be required at peak periods during a start-up.

The new layout is shown in Fig. 4b. The first stage was to move the old console 2 to join up with console 3. This was a simple matter, involving the re-routing of only a small number of cables, since nearly all the console equipment is driven by modules in a CAMAC crate housed in the console, the CAMAC controller being connected to an extension of the console computer I/O bus.

Since only one alarm screen is needed for the two consoles, and other arrangements were being made for the permanent displays, two modules of the old console 2 could be removed. Four new modules were then built on to the old console 1 and the permanent display unit installed. The table in the centre provides storage for the instruction books, manuals, and maintenance information. An overview of the control room is shown in Fig. 5.
Surprise has been expressed that there is no reservation system to enable an operator at one console to reserve a computer subsystem for his use, to prevent another operator from trying to change some value on this subsystem. However, since all the control programs are run in the consoles, the interpretive system ensures that there can be no interference between different programs or sections sent for execution in the same computer, so that the worst that could happen would be for a value to be set by one operator and then subsequently changed by another. In practice, perhaps because of the possibility of verbal communication between operators, this does not happen with sufficient frequency to become a problem.

2.10 Applications software

It was interesting to see how the physicists, engineers and technicians, who were responsible for the design of various parts of the accelerator, approached the problem of writing programs to control their equipment. Some of them had previous experience of writing FORTRAN programs to calculate various parameters needed for their designs, but a majority had never written a program before. However, once they had been shown how they could type an immediate statement on the demonstration system and see it obeyed, and then write a short program and see the sequence called for carried out, most of them gained sufficient confidence to use the computers in the laboratories and assembly hall to test their equipment.

The experience gained in this application then showed the way to write programs to control their equipment when it had been installed and was working as part of the accelerator. Since each person's experience and method of approach is different, the programs produced varied widely in style and efficiency, but they did the jobs required at this stage. The speed with which they were made to work was largely due to the simple (to the user) error return system used in NODAL. On running a program, if the writer has made syntax errors, or if a call to the hardware brings an error return, the execution of the program stops and an error message is displayed on the interactive screen, such as

NONEXISTENT NAME AT LINE 1.75

or

HARDWARE ERROR AT LINE 6.25.

The use of a control character on the keyboard then causes this line to be displayed, with an arrow showing where in the line the error occurred. The system is automatically put into the edit mode, so a syntax error can be corrected and the program run again.

In view of the different styles of programming and the difficulties this may make for people trying to understand each other's programs, the question has been asked whether it would not have been better to have laid down standards and to have run courses for those concerned to try to achieve some uniformity in the programs. Apart from the fact that most of the people concerned were so busy with the installation and commissioning of their equipment that it would have been difficult to persuade them to find time to attend such courses, too little was then known about how best to program a multicomputer system to control an accelerator like the SPS, to be able to lay down such standards. It was only as a result of the experience of people writing programs in many different ways that the most effective methods of control could be determined. The flexibility provided by having an
interpreter in every computer and unrestricted intercommunication between computers allowed
the more ingenious to use the system in ways that had never been envisaged at the design
stage.

The first set of programs, mainly written by the equipment designers, was attached to
the "tree" described in the previous report, so that they could be run using the touch
buttons. Sufficient programs were available to perform all the actions required during the
commissioning and initial operation of the accelerator. Some of these programs were orien-
tated more towards the testing, rather than the operation of the accelerator, but they pro-
vided for the actions required.

Experience with these programs showed the way to redesign them to meet the operational
requirements, and this redesign is being carried out progressively, mainly by the crews
operating the accelerator. These new programs are placed on an "operations" tree, the old
ones still being available on the original tree, so that they can be used for test purposes
by the groups responsible for the maintenance of the hardware.

Whereas originally the pessimists wrote into their programs extensive checking routines
at each operation, there is now sufficient confidence in the integrity and stability of the
system to write simpler programs without error checking. Should there be a failure due to
a hardware fault, then either a separate hardware diagnostic program can be called, or,
where acquisition and not control is involved, the equipment providing the error return can
be skipped over. Provision is made in NODAL to branch to another group on error return,
which simplifies this procedure.

It has been suggested that, while it is clearly advantageous to be able to modify pro-
grams rapidly and to improvise the control sequences when one is starting up a new device
like the SPS, the ease with which this can be done becomes a danger when moving to the
operational phase. This could be true in a situation where there is no protection scheme
and no discipline. We did suffer to some small extent from the improvements introduced by
the groups responsible for the equipment not always being known to the operators, but this
difficulty has been largely eliminated by the introduction of the operations tree. Programs
attached to this tree can be modified only by the operations group, who have the appropriate
protection key.

The ease of writing new programs, whether for more sophisticated control of the acce-
lerator complex, for gathering and analysing statistics, for documentation, or even just
for playing games to while away the long night hours when the accelerator requires no atten-
tion, means that the demand for disc space on the library increases continuously, but this
seems a small price to pay for all the advantages.

At the time of writing, there are about 1.3M words of program and data files stored in
the library, with over 100 named users.

2.11 Protection

The basic protection scheme, as described in the earlier report, has proved to be
adequate and has not needed appreciable modification.

Logging-in can be carried out either by putting the user's plastic identity card into
a card reader on the control desk, or by typing in "SET USER = (CERN personel number)"
followed by an optional private password. Contrary to expectation, the latter, more clumsy way of logging-in seems to be the most frequently used, perhaps because the plastic card is often in the pocket of a jacket hung up on a coat-rack the other side of the control room!

The logging-in program then looks up a table that gives the protection code attributed to the user. This protection code is in two parts, the section number and the capability word. The section number refers to a small group of people responsible for a particular type of equipment. This forms the only protection for files on the library. Files can be declared as free, section-protected, or read only. Free files can be written into by anyone, section-protected files can be written into only by those with the correct section number. The change of status of a file, such as from read only to section match, can only be carried out by someone with the correct section number. Part of the identifier for a file is the index number and each index is attributed to a section. A section can have more than one index.

The protection scheme for access to the hardware operates at the data-module level. It is often desirable that some entries in the data tables, such as maximum values, addresses, etc., should only be changed by the people responsible for the equipment. In such cases, the data-module property is protected, and requires a section match for a write operation. Read operations remain unrestricted as in general throughout the system. In the case of other properties, such as to set a required value for a power supply, the access must still be restricted, but not so tightly, and for this the capability word of the protection code is used.

Each bit of this word represents one major subsystem, such as RF, vacuum, etc. If the appropriate bit is set in a person's capability word, he can control all the equipment in that subsystem. The operations staff must, of course, have full capability.

It is difficult to assess how far one should go in providing access protection in such a control system. It should not be necessary to try to protect against anyone deliberately trying to sabotage the system, since there are so many simpler ways that could be used by anyone determined to wreck things. It may, on the other hand, be necessary to protect against unauthorized actions which may give individual benefits. For example, an experimental team may be allowed to vary some of the parameters of the secondary beam-line feeding their experiment, but should not be allowed to vary the splitting-ratio between the targets. Since it would be difficult to keep protection codes for all the individuals involved in experiments in the system and up to date, the experiment itself is given a section code, and the VDU or teletype installed in the experimental barrack is permanently logged-in with that section code. This system works well and no difficulties have been reported.

The other thing a protection scheme can be called to do is to prevent a program error from causing the operation of the wrong equipment. A typing mistake in an equipment name usually stops the program with the comment "NONEXISTENT NAME!", and the chances of the typing error producing a valid name must be small. More likely is the use of an incorrect serial number causing the wrong unit of the given type of equipment to be operated. This was the most frequent error in the commissioning period, mainly because of the failure of the documentation to keep up with changes in the allocation of equipment numbers. This type of error is not normally detected by the present protective scheme. Provision is made in the system for differential protection to be applied to the individual units controlled by one data module, which might help in some cases.
On the whole, it seems worth while to have a protection scheme for the hardware in addition to that normally provided for the operating system and the files. That used on the SPS is thought to be a reasonable compromise. It should be noted that, by putting the protection at the level of the data module, the simplicity of the distributed data-base philosophy has been maintained. The penalty paid for this is that if a program is called by an unauthorized person, it will run until it tries to make a protected access and will then abort. To have introduced a scheme calling for checking whether all actions were authorized before starting to run the program would have introduced unacceptable complications and delays, as well as requiring duplication of part of the data base.

2.12 The alarm system

One problem which was not discussed in the previous report is how to deal with the faults in the hardware and software that must inevitably occur in such a complex system. Faults that occur during the execution of a program and cause it to fail, because the hardware cannot respond in the correct fashion, or because of software errors, are normally reported on the interactive screen on the console. However, it is not necessarily desirable for a simple fault to stop the program, particularly in the acquisition of a series of values, and so the facility to branch on a fault was included in the NOBAL language.

The main problem is not the faults that show up during the execution of a control program, but those that may occur at any time and jeopardize the operation of the accelerator. To cover these, surveillance programs run periodically in the remote computers, checking the actual status and values with the demanded status and values stored in the data-module tables. When a discrepancy is found, a message is sent to the Alarm computer, which composes a line of text to be displayed on a reserved colour screen on each console. A new message appears in black letters on a solid red background to make maximum visual impact. The operator can acknowledge the message, in which case it changes to red letters on a black background, or cancel it, causing it to disappear. A cancelled message will reappear the next time the surveillance program runs if the fault persists, and error messages are automatically cancelled when the program finds that the required conditions are restored.

One of the main problems with any alarm system is what to do about consequential alarms. Obviously, if a pump supplying cooling water to several equipments fails, it is unnecessary to tell the operator that the water flow in each of these equipments is inadequate. Originally, it was thought best to keep the surveillance programs as simple as possible, so that they would report every single defect, and devise programs for the Alarm computer to analyse the information it received according to tables of consequentials. When the difficulties of this approach were realized, requiring the analysis of large numbers of situations which might never occur, the possibilities of "self-learning" systems were explored. The idea here was that for each occurrence, at first the operator would be supplied with all the information available. He would then analyse the situation, locate the primary cause, and then indicate to the system the information he would like to see displayed if such a combination of alarms occurred again. The system would thus gradually build up a store of situations that actually occur, and the operator's task would be progressively eased.

Owing to the pressure of other work, this interesting approach was not pursued. Subsequent experience has shown that there is a better way and that is to attack the problem
at the other end; the surveillance programs running in the remote computers. It has been shown that, by a relatively small complication of the surveillance programs, the simple and obvious consequentials occurring within one type of equipment can be sorted out, reducing the number of alarms reported to the operator to a sufficiently low level as to make further effort to eliminate consequent messages unnecessary. This has now been extended to consequentials involving more than one type of equipment, in more than one computer.

The surveillance programs are written in NODAL and are normally scheduled to run at fixed time intervals, but these intervals must not be too short, or one may not have run to completion before the next is scheduled, since they normally run at the lowest NODAL level.

This restriction was eased considerably by the introduction of the Fast Alarm Scan Program (FASP). The hardware of the MPX system makes provision for scanning through a table of MPX addresses in the computer memory, successively acquiring the summary status at each of these addresses and comparing it with that in the table. This carries on autonomously until a discrepancy is found, which causes an interrupt. This interrupt can then be used to call the associated surveillance program to identify the fault and send the appropriate alarm message. By this means, frequent scheduling of the surveillance program is avoided, without losing the rapid response to a change of state.

3. EXPERIENCE WITH THE SYSTEM

This section will be relatively short, not because of the lack of experience, but because a lot of the experience with individual subsystems has already been described in Section 2 when discussing the reasons for the changes that have been made, or will be described in Section 4 in connection with the reliability of the equipment. Therefore, this section will concentrate on the experience with the system as a whole.

The computers with a stand-alone software system, including the NODAL interpreter, were available early in the program and they were used in the laboratories and workshops for the development and acceptance testing of various parts of the accelerator equipment, speeding up these operations considerably. The simple programs required were written by all grades of staff, and no special software support was needed.

The next stage, when the computers had been installed in the service buildings, connected up to the message-transfer system, and were being used to commission the large assemblies of components, caused some headaches, mainly because of the sheer volume of equipment to be connected up and "debugged". Owing to the short time scale, the fifty data modules required had to be written by quite a number of people, most of them not professional programmers, and since a mistake in any one of these can crash the system, it is understandable that there were some difficulties. However, it is clear that these difficulties were minute compared to those that would have been experienced if all the applications programs would have had to be written in the conventional manner. The applications programs were written and tested, mainly as a part-time activity, by about 50 to 60 people, most of whom had no experience beyond batch FORTRAN, and many had never written a program before. The insulation provided by the interpreter allowed these activities to go on without much mutual interference.
As has been described in Subsection 2.10, the original programs written by the hardware groups were, in many cases, orientated more towards testing the equipment than operating it, but progressively they have been modified, mainly by the operations staff, and transferred to the operations tree.

While the need for the control and surveillance programs is self-evident, it was more difficult to persuade the groups to put a reasonable priority on the provision of statistical and documentary programs. The number of these is now increasing and one development of interest is a scheme for providing displays which explain the possibilities available at each branch of the tree. These are called by pressing a "HELP" button, which presents the appropriate display according to the current set of touch-buttons.

It was also found that some of the personnel responsible for maintenance of the equipment were not getting full information on the state and alarm history of this equipment, because they were diffident about going to the control room and asking to use a console or terminal to call up the information. This has been overcome by providing a few VDU's in the laboratory and office buildings, which are connected to the Service computer. A special program runs in a single dedicated NOMAL buffer at the interaction level, which provides a tree of choices in the same manner as the touch-buttons, except that it is necessary to type in the number of the branch, etc., chosen. The leaf programs gather the required data and display the results, which can be the present state of some parts of the equipment, a history of faults, a histogram of power consumption, etc. Since all the terminals share a simple buffer, there can be some delay in response, but this is adequate for the purpose required.

One of the main criticisms of the decision to base the software system on an interpreter was that it would prove to be too slow. This aspect has already been discussed in Subsection 2.5, but it is worth repeating here that the facilities provided for incorporating assembled or compiled subroutines as functions which can be called by the interpreter, for the autonomous collection of time-critical data in the CAMAC system and for the scheduling of "real-time" assembled programs from the data modules, have satisfied the requirements encountered so far. This was confirmed in a recent meeting in which users of the SPS control system were asked to quote cases where it had not been possible to provide the necessary speed of operation, and none were forthcoming. However, it was said that, while it may be satisfactory for a machine like the SPS with a cycle time of several seconds, it would not be fast enough for the CPS, where the minimum cycle time will be under one second. Since the limiting factor seems to be the virtual memory system of the NORD-10, presently used for the data modules, which imposes a delay of about 40 msec at each exchange, this should be solved by the proposal to equip the remote computers with sufficient semiconductor memory to enable all the data modules to be resident, in which case the delay will only be a few microseconds.

On the subject of data modules, the original concept was that these should be rather simple, just providing the means for accessing the individual items of hardware, and so could be quite stable once they have been "debugged". Therefore no special facilities were provided for easy loading of modified data modules, which normally requires system regeneration. In practice there have been many changes, for which there are two main reasons. The first of these was a consequence of the way the data modules were originally written in
many different styles. As previously mentioned in Subsection 2.7, in many cases these were rewritten to follow the subsequent standards, and to include the protection where this had not been included originally. The second, and more long-term, source of change in the data module was its elaboration to make it do much more than was originally expected. This included the provision of extra properties to simplify the surveillance programs and the ability to transfer arrays in addition to simple values, as well as scheduling real-time programs to acquire data at a particular point in the cycle.

4. RELIABILITY

As was stated in the earlier report, a large accelerator like the SPS can be expected to have a "down time" of the order of 10% of the scheduled operation time, owing to the complexity of the equipment and the pressure to operate at the limit of what is attainable. Naturally, we would like the control system to make a negligible contribution to this down time, since failures in this area are particularly frustrating to the operators, but in practice the system can be considered sufficiently reliable if it is responsible for less than 1% loss of scheduled operation time. It would not then be worth while expending a large amount of money and effort to reduce this figure still further.

After the commissioning period, the SPS started regular operation for physics in January 1977, and so statistics are available for about twenty months of operation, at the time of writing this report. In this period, the average number of protons accelerated each cycle has risen from about $4 \times 10^{12}$ to about $1.5 \times 10^{13}$, as the specialists learnt the reasons for the limitation on performance, and the operating crews found how to put the theories into practice.

The way the accelerator down time, and that attributed to the control system, has varied from January 1977 is shown in Fig. 6. It can be seen that, on the above criterion, the control system had reached a satisfactory state of reliability after the first year of operation. The particularly poor reliability in period 5 in 1977 was due to a combination of circumstances. A large number of thunderstorms in this period caused the difficulties and damage described below, at a time when many of the experts were on vacation, and so the time to diagnose and repair the faults was increased. In addition, a number of changes and additions had been made in the two-months shut-down between periods 4 and 5, and some of the troubles resulted from these. In contrast, fewer changes were made in the shut-down at the beginning of 1978, very extensive testing was carried out at the end of the shut-down, and precautions had been taken against mains voltage fluctuations. In considering these figures, it should be borne in mind that the control system for the SPS includes more than just the interface to the equipment to be controlled, and some of the failures of "the computer system" would in earlier accelerators be attributed to "the power supply system", "the extraction system", etc.

Although the amount of time the accelerator is out of action because of control system failures is satisfactorily small, the failure rate in the control system itself is not negligible, but many of the failures do not stop the accelerator immediately, and it is often possible to rectify a failure before it affects the operation of the accelerator. This is the result of the decision made at the start of the design that equipment should, whenever possible, be designed to "continue to do what it has been told to do until it is
told to do something different". In practice, this called for buffering the demanded values in the interface as close to the equipment as possible, the use of discrete function generators loaded with tables of values and triggered from the timing system, the use of a fully distributed data base, and means for recovery from a failure without disturbing the system. At the time this was decided, the cost of the extra memory required was not negligible, and in some cases single buffering was used where double buffering would have saved complexity in the software. Now that the hardware costs have fallen so dramatically with the widespread use of large-scale integration, there need be no such restrictions for future applications.

As well as designing the system to try to minimize the effects of failures on the operation of the accelerator, a considerable amount of effort was also put into ensuring
that the mean time between failures (MTBF) for the individual parts of the equipment was as high as possible. This was done by selection and pretesting of components, running them well within their ratings, and extensive "soak testing" of the completed units.

The other factor in the over-all reliability, the mean time to repair (MTR), was also taken into account, by modular construction and built-in test facilities. Since the commissioning, there have been some changes of components where the MTBF has proved to be too short, but the major improvements in reliability have been due to the effort put into reducing the MTR. Some of this effort has been directed to speeding up the exchange of parts of the system suspected to be faulty, but the major part has been put into developing comprehensive surveillance and diagnostic programs to locate the source of faults, some of which can be very difficult to diagnose in such a complex system. Some of this work is described in the subsections which follow.

4.1 Computers and peripheral equipment

Although it is common to blame "the computers" for any failure in the control system, the failure rate of the central processor units has been reasonably satisfactory, after the "teething" trouble had been overcome. Initially, most of the failures were in the ROM control store and core memory modules. The former were replaced by an improved type and after a period of "natural selection" for the latter, there has been little further trouble with these. Subsequently, the failures given under the heading "Computer" in Table 1 have been mainly in the peripheral drive and I/O cards, and recently there have been failures because of a faulty batch of semiconductor memory.

The next difficulty that showed up was the sensitivity to thunderstorms. When the control system was designed, assurances had been given that the mains supply, by means of a direct 380 kV link with a node of the French grid system at Génissiat, 35 km from the site, would be both secure and stable, so no special precautions were taken with the supply to the computers and interface equipment. In practice, it was found that, during thunderstorms, the voltage on one or more phases could drop to as little as 50% of its nominal

<table>
<thead>
<tr>
<th>Unit</th>
<th>Average number in service</th>
<th>Number of failures</th>
<th>MTBF per unit (months)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>37</td>
<td>75, 105</td>
<td>106</td>
</tr>
<tr>
<td>Drum</td>
<td>21</td>
<td>29, 60</td>
<td>18</td>
</tr>
<tr>
<td>Disc</td>
<td>18</td>
<td>14, 13</td>
<td>28</td>
</tr>
<tr>
<td>Other peripherals</td>
<td>160</td>
<td>41, 50</td>
<td>60</td>
</tr>
</tbody>
</table>

Notes: This table includes all computers used by the SPS, either as part of the control system or for testing and development in the laboratories. An intermittent fault may contribute to more than one failure, if it has been incorrectly diagnosed on the first occurrence.
value, for a period of about 50–100 msec. This was due to flash-overs on one of the other overhead lines leading from Genissiat, and the action of the automatic clearing and resetting mechanism. The first action of these dips in the mains supply was to activate the power-fail shut-down procedure in the computers. The second was more serious, the dip and subsequent overshoot causing the computer power supplies to fail at an alarming rate when the thunderstorms were frequent.

To overcome these and other difficulties which were caused by transients in the mains supply, it was decided to fit "no-break" power supplies, of the rectifier-battery-inverter type, to feed the computers, the CAMAC and the MPX system. Not only did this solve the problems due to mains variations, but it also allows the computer system to continue to operate during the few seconds needed to change to the Swiss supply from the Meyrin site, should the French supply fail.

Although failure of a computer is not very frequent, the cause of a failure can often be difficult to diagnose, especially if it is an intermittent one. It is then best to replace the computer by one known to be in good order, and search for the fault: at leisure in the laboratory. Owing to the relatively large size of the NORD-10 computers, such an exchange is not trivial, but it has been eased considerably by the fitting of special junction boxes to allow rapid changeover of the I/O bus and by the provision of a spare computer and drum unit, mounted on wheels, which can be taken by a special vehicle to any of the auxiliary buildings. This computer is normally connected to the network, so that it can be surveyed to make sure that it is in working order when it is needed.

Of the peripherals, it is a little ironic that the sealed fixed-head drums, chosen on the grounds of reliability, have been a major cause of lost time. Not only have failures been more frequent than expected, but their diagnosis has been difficult and time-consuming. When the failure rate seemed to be increasing towards the end of 1977, and the computer manufacturer decided not to support these drums in future, the decision was made to change to moving-head discs for the computers in the central building and to semiconductor memory for those in the auxiliary buildings, as described in Subsection 2.2.

Other failures that have occurred in the conventional peripheral equipment, particularly in the magnetic tape units, cause annoyance but they do not stop the accelerator.

4.2 The message-transfer system

The message-transfer system hardware, which provides the interface between the data links and the computers, has been reasonably reliable once the system was "run in", as can be seen from Table 2. However, failures are difficult to diagnose and, owing to the fragility of the interconnections, the replacement of a card to cure one fault may introduce another. This is particularly true on the message-handling computer, where the units are tightly packed to minimize the delay introduced into the DMA cycle. This is the main reason for the introduction of the two-node network as described in Subsection 2.3, so that a complete group can be exchanged, simply by changing over the data-link plugs. In addition, all new units fitted to the system use the latest printed-circuit techniques, which reduce the number of interconnections required.

An early source of trouble in the message-transfer system was the data-link repeaters, and most of this was due to a single type of integrated circuit. This is a differential
receiver unit, which was being operated within the manufacturer's rating, but which showed a high rate of failure. No alternative was available and the only temporary solution was selection of the integrated circuit units after extensive soak testing. The repeaters have been redesigned to use other integrated circuits, and the old ones are being progressively replaced by the new version.

If a data link fails it may take a long time to find which repeater is faulty, especially in the case of the longer links, since, at present, it is necessary to visit each auxiliary building through which the link passes, until the faulty unit is found. This will not be necessary in future, when the provision for remote switching of duplicate repeaters, as described in Subsection 2.2, is implemented.

4.3 The equipment interface

The equipment interface discussed in this section includes the CAMAC, the MFX system, and the timing system. It does not include the function generators and other specialized electronic units which are considered as part of the equipment being controlled. Detailed statistics have been kept since June 1977, and a summary of these is given in Table 2.

The CAMAC is used mainly as the interface to the MFX controllers and for the acquisition of data from the beam-monitoring system, where a large number of variables have to be read in a short period of time. The modules used are mostly manufacturers' standard lines and the crate controllers were made by the computer manufacturer to a CERN specification.
The MPX system was designed at CERN and the modules manufactured and tested, mainly by two firms, to CERN drawings and test specifications. This was also true for the timing modules. An attempt was made to keep the whole system simple and robust, and a well-proved connector is used to make the connections when a module is plugged into a crate. This care taken in the design has been rewarded with a MTBF per unit considerably better than that of the CAMAC units, as shown diagrammatically in Fig. 7. It can be seen that in both cases the control units, which are the most complex modules, are also the least reliable.

Despite this good MTBF, there are so many modules in use that there are on average one or two failures per week, and this is sufficiently frequent to make it worth while to reduce the MTR to the minimum. This has been done by fitting responders in the last station on each branch of the serial highway, and by writing a series of programs which test each station of each branch in turn. By this means, faults down to the individual module level can usually be located from the control room.

4.4 The software system

The operating system SYNTRON and the NODAL interpreter were developed early in the program and were soon in regular and stable use in stand-alone computers used for testing components of the accelerator. Not unnaturally, since it was the first tryout of a new concept, difficulties started to appear when the multicomputer network was set up and used by the equipment designers for writing and testing their applications programs. Some of these problems were due to circumstances that had not been foreseen, and to blocking situations. The latter were largely cured by the introduction of the time-outs, which, in turn, caused difficulties for the users when some of their programs, which had previously executed satisfactorily, were aborted.

The message-handling computer was the only one which did not have the NODAL interpreter, as it had been thought to be unnecessary on this single-purpose computer. However, during this difficult development period it became evident that this was a mistake, since when
things went wrong in the network, the cryptic error messages appeared only on the teletype connected to the message-handling computer, and it needed an expert to decipher them. Therefore NODAL, and the extra memory needed, was added to this computer, and the interface to the message-transfer programs designed to make it possible to control the links and to obtain plain-language error messages at the consoles, as well as to gain access to the diagnostic and statistic programs which produced the traffic measurements reported in Subsection 2.3.

The difficulties were sorted out in time for the start-up of the accelerator and now the network system is very reliable, the message-handling computer often running for several weeks at a time without reloading.

The stability of the software system in the individual computers depends, as in all systems, on the frequency with which changes are made, and a large part of the effort in the last year has been put into facilities to allow changes to be made without reduction in the reliability of the system.

For the reasons given in Subsection 2.7, most of the changes in the non-NODAL parts of the system have been in the data modules, and a spare computer, with sufficient interface equipment to test the actions called for by the data modules, has been used for some time as stand-alone data-module test facility. Whenever an alteration is made in a data module, it must be tested extensively on this test set-up, before the new version can be loaded into the control system.

By this and other means, it has been possible to keep the accelerator down-time attributed to system software troubles to a negligible proportion of the total. The only place where some difficulties are still experienced is in the Experimental Area computers, where the demands on the computers are heaviest and the changes most frequent.

5. CONSIDERATIONS FOR THE FUTURE

This final section is written in answer to the question which is often asked: "If you were starting again now, with the experience gained and the subsequent developments in hardware and software, what changes would you make in the design of the system?"

One thing is certain, and that is that the basic philosophy would not be changed. To be quite clear, it is worth repeating the main elements that go to making up this philosophy. These are:

- The use of a single, powerful, high-level language that provides facilities for multi-computer programming and debugging in a fully interactive manner, so that the equipment designers and builders can write the control and test programs, and the operations staff can modify and add to them as required.

- The distribution of the data base amongst the computers and the splitting of it into data modules, each of which contains the data table and specialized driver for all the units of a given basic type of equipment, irrespective of the use to which it is put. No other working copies of the data tables exist within the system.

- The use of general-purpose consoles which are in effect computer peripherals. This has the requirement that all the equipment of significance to the operation of the machine must be computer controlled or surveyed.
- The use of a network of similar minicomputers to provide all the facilities, including those which would normally be provided by a medium-sized central computer.
- The use of a standardized interface system, as simple and cheap as possible and well adapted to the control of a large process.
- The use of accelerator equipment that is designed to continue to operate under the most recently demanded conditions, even when failures occur in the control system.
- The avoidance of the computers being involved in actions where timing is very critical by the provision of a timing system which can be programmed to activate external hardware at the appropriate time.

There are many variations possible in the detailed implementation of a control system which keeps to this same philosophy, and some of these are examined below in an attempt to answer the original question.

5.1 The programming language

NODAL was designed to be interpreted, and to provide the facilities needed to program a multicomputer control system, in a simple and flexible manner. There is no doubt that it is successful. The questions that have to be asked here are: "Is there a better language?" and "Is it necessary to use an interpreter rather than compile the application programs?"

None of the standard languages can offer comparable facilities for multicomputer programming, and there would be no incentive to move to another non-standard language unless it had much more to offer, and such a language does not seem to have been devised so far. On the second point, if debugging is to be carried out easily in an interactive manner, it seems essential to keep at least some of the structure of the source programme in the code that the computer obeys. Up till recently, this was only possible with an interpreter, which uses the source programme directly, or to a lesser extent with an incremental compiler. The development of the Nodiler (see Subsection 2.5) has provided a method of partial compilation which retains nearly all the advantages of an interpreter.

A criticism has been made that NODAL is not a "modern" language, lacking such things as flow control to reveal structural errors in a program and the possibility of creating new variable types. Flow control can be carried out in a compiler, but NODAL is interpreted line by line, so structural errors can only be revealed when the program is run. However, the error return and simple editing facilities described in Subsection 2.10 allow such a simple recovery from mistakes that the lack of flow control causes little inconvenience. The ability to create new variable types would only be of interest for applications of NODAL outside the control field, as there are sufficient variable types to cover present requirements. Type checking is carried out in NODAL.

Thus the answer to the questions is that certainly NODAL would be retained and it would be interpreted, and there seems to be no strong case for any other high-level language, since all foreseeable requirements can be met by NODAL, supplemented by functions, if the other elements of the philosophy are retained. These functions can be written in NODAL, nodified if they run too slowly, or written in assembly language or equivalent, where the highest speed is needed.
5.2 Data modules

No great change is foreseen in the data module concept, but the implementation would gain from the experience with the SPS. Standards and guidelines could be laid down early, and subroutines provided to do the work common to most data modules. Even with these it may well be better to have the data modules written by a few professional programmers, if they are available, than as a part-time activity by the larger number of engineers who wrote those for the SPS.

Elaboration of the data module beyond the "one action at a time" concept, with NODAL used to couple these single actions into a sequence, is not per se undesirable, as long as it is restricted to providing more powerful properties, and is not used as a means of introducing assembly language applications programs.

In the existing system, most of the data modules are held in core-loads, which are swapped in from the drums when required, and this introduces a delay of the order of 40 msec when a program calls for the use of a data module different from that in core at the moment. This delay will be eliminated when the drums are replaced by semiconductor memory and, with the continued reduction in the cost of such memory, any future system would have all the data modules resident.

5.3 General-purpose consoles

When used correctly, the combination of touch-buttons, a rolling ball to move a cursor, and one knob seems to give all the facilities required for the operation of a large accelerator like the SPS, although a case could probably be made for a second knob when operating a fast cycling accelerator (cycle time < 0.1 sec) when two-handed optimization of non-orthogonal parameters is feasible. Since these devices are not specific to the operation of an accelerator, there seems to be no reason why an identical console could not be used for any type of process control. For small systems, where the cost has to be kept to a minimum and some restrictions can be accepted, the display screen and the touch-buttons could be combined, using a plate with the capacitance-sensitive areas over a portion of the display screen.

Although the present system works well with separate generators for the character and graphics displays, these would be combined into a single microprocessor-controlled display generator for a future system, taking into account the trend towards reduction in price and increase in speed for semiconductor devices and memories.

The present arrangement, of having all the interface equipment for the console devices mounted inside the console, would be retained, as it allows rearrangement of the consoles with the minimum of cabling changes, as described in Subsection 2.9, and the disadvantage that maintenance may be a little more difficult can be overcome by suitable mechanical design.

5.4 The computer network

The use of separate minicomputers to carry out the different tasks at the control centre requires a fast transfer of information between the computers and this has been provided by the special packet-switching message-transfer system developed for the SPS.

Although the network possibilities now offered by some computer manufacturers would need to be examined closely for any future control system, the impression obtained is that,
in order to achieve generality, the software overheads are too great to provide the speed required for this type of service. If starting the design now, the SPS message-transfer system could be made simpler and cheaper using the latest integrated circuits, with buffering in the interface to reduce the DMA requirements on the message-handling computer. An investigation would also be made into the possibility of using one of the several protocols now being proposed as standards. Fibre-optical transmission systems might be used to eliminate the need for intermediate repeaters on the longer links, as the extra cost of the link may be offset by the savings on the repeaters, and problems with electrical noise pick-up would be absent. Taking advantage of the increased bandwidth of such links\(^a\), the number of them round the ring could be reduced by changing the topology to a ring-connected, multi-node system\(^b\). The main difficulty with optical links is their sensitivity to radiation, which would preclude their use in the tunnel.

There has been some questioning of the advisability of relying on a single library computer as the main file storage, and the possibility of blocking if the demand for files from different computers at the same time becomes excessive. It is true that the library is the major source of traffic in the network, but this is well within its capability, as can be seen from Subsection 2.3. Delays which were experienced at peak activity in the early running were due to deficiencies in the filing scheme of the operating system, and not to the speed of the message-transfer system. Changes to the filing system increased the speed by a factor of five and the difficulties disappeared. The alternative of having mass storage on all computers, or a mixture with mass storage on some computers and a library computer for the rest, suffers from a number of disadvantages. The normal method of operation of the SPS multicomputer system is for the applications programs to be loaded into the console computers. Thus the only way to cut down the library traffic appreciably would be to put mass storage on the console computers. However, since the consoles should be general purpose and completely interchangeable, this means keeping as many copies of each program as there are consoles, with the usual difficulties of ensuring that all copies are updated to the same stage, apart from the cost of providing the extra storage.

5.5 The interface system

The interface system for the SPS uses a mixture of CAMAC and the specially developed MPX system because there was no standard system available that satisfied the criteria for a process control interface, which can be summarized as follows:

a) It should be computer independent.
b) It should be mechanically and electrically robust, and use industrial type plugs and sockets.
c) It should provide complete isolation, including the earth circuit, between different pieces of apparatus connected to it.
d) It should allow the connection of a considerable number of stations, each containing a number of interface modules, on to a common bus system. This bus must be extendable to at least 200 m, and preferably further.

\(^a\) Bandwidths of 10M and 140M bits/sec have already been achieved with repeater spacing of 12 and 6 km, respectively, and further progress is anticipated\(^b\).
e) The control unit should incorporate parity and other checks into the system to minimize the effects of interference pick-up.

f) It should have built-in read-back facilities for remote fault location and diagnosis.

g) It should be inexpensive.

h) It should be an international standard with a large variety of interface modules available from a number of manufacturers.

Would the situation be any different if one was starting afresh today? Serial CAMAC is a workable system, but it does not satisfy criteria (b) or (g) and most of the modules available do not satisfy (c). For small systems, where the development effort to produce a special interface cannot be justified, CAMAC seems to be the only system where sufficient different types of modules can be bought "off the shelf" to satisfy most requirements, although not many of them are specifically designed to carry out process-control operations in the most economical way.

If starting an SPS-sized project today, it would still be worth while to develop an MPX system, to gain advantage of the lower cost, simplicity, and improved reliability (see Subsection 4.3) compared with serial CAMAC. However, a new MPX system might take a rather different form, as this is the field in which microprocessors could make their greatest contribution. At present, CAMAC is used in the accelerator control system for interfacing the MPX system and for the rapid data acquisition. In a future system, the MPX bus controllers would incorporate microprocessors and interface directly to the minicomputer I/O bus. The fast data acquisition would be accomplished by MPX crates containing microprocessors triggered from the timing signals to acquire the data when required and then to pass selected items through the MPX system to the minicomputer on request. This possibility will exist with the present MPX system in the near future, using the CMBUS system. Further applications of microprocessors are discussed in Subsection 5.8.

In the case of the secondary beam lines and in the experimental areas, the use of serial CAMAC seems still to be justified for the beam monitors and allied equipment, since this is the type of duty for which CAMAC was designed. For the future, however, the picture may change, since a replacement for CAMAC, to allow faster acquisition and treatment of data, is now being discussed and some initial standards are being laid down.

5.6 Autonomous operation

For equipment to continue to operate after a failure in the control or interface system requires that the demanded status and values, etc., must be stored in the equipment. The continuing fall in the cost of semiconductor devices removes the cost penalties which have had to be suffered in the past, particularly when double buffering is employed to simplify changes of conditions which are time critical. This subject is discussed in the next two subsections.

5.7 Critical timing

As part of the original design philosophy for the SPS control system, actions and closed loops were divided into three categories, according to the order of response times required: microseconds, milliseconds, or seconds.
The microsecond range would be covered by hardware alone, which could be parametrized through the computer system, but would otherwise operate in an autonomous fashion, triggered where necessary from the timing system described in Subsection 2.4. In the millisecond range, a single computer could be involved (e.g. the main magnet power supplies), but triggered hardware should be used where possible, as in the case of the function generators. The multicomputer system would be expected to operate in the order of the reaction time of an operator; from a fraction of a second upwards.

Starting again at the present time, the same policy would be followed, but the term "hardware" would include devices incorporating microprocessors performing single stream operations with their programs in read-only memory. These devices would be used for all time critical operations or control loops, and triggered from a similar timing system to that used at present. The interface to the computer system would then be required to provide the parameter-setting information or acquire the resulting data in the time scale of the operator.

5.8 The impact of microprocessors

The advent of the microprocessor can have considerable impact on the design of control systems, as in other fields, but they do not bring any change in basic principles, allowing decentralization, which was already taking place in control systems, to proceed at a higher rate and lower material cost. In the enthusiasm to incorporate microprocessors in all kinds of equipment it is sometimes forgotten that the development costs can be considerable, and the rate of their introduction into control systems may be limited by the effort available, particularly for the software development. This is particularly so if the introduction of microprocessors is used to complicate the basic system rather than simplify it, which should be the real aim if it is to be reliable and understandable by the operating staff.

Since the term microprocessor now covers such a large range of devices right up to those that have the capabilities of a small minicomputer, we must define the type of devices that are being considered in this section. They are effectively intelligent process controllers, performing a single stream of sequential processes according to external trigger. They can be divided into two types according to use; general-purpose devices which will perform different actions according to the control programs loaded into their memory, and special-purpose devices, performing a fixed set of operations and often built into the equipment they are controlling, preferably with the control programs in read-only memory.

To give an idea of the possible duties which might be carried out by such devices in the SPS system, without departing from the basic principles, some examples are given below of applications which are being, or could be, implemented and which allow simplification of the data modules or applications software.

5.8.1 Autonomous function controller

Once the associated memory has been loaded from the computer system, this device, described in Subsection 2.1, performs a series of operations on the CAMAC dataway independent of the computer. The necessary arbitration is provided so that the highway is made available to the computer when required for other purposes. The latest unit incorporates a microprocessor and it has been renamed "Auxiliary Crate Controller" to line up with recent ESONE decisions on nomenclature.
5.8.2 Graphics display generator

Equipment of this type has already been installed, as described in Subsection 2.9, in the form of microprocessor-controlled digital-storage graphics controllers. These act as peripheral devices to the console computers, which channel the raw data to these controllers rather than to the Display computer, which was becoming overloaded at times. The standard unit provides a resolution of 384 \times 288 dots, which is adequate for normal purposes on the colour screen, and a high precision model is being developed for special requirements to give a resolution of 768 \times 576 dots. This requires higher speed electronics and four times the memory, but if the present cost trends continue, this may become the standard for the future, as it gives an improved appearance to the curves.

5.8.3 Function generators

The SPS uses about 300 function generators, mainly to supply reference voltages to power supplies which provide currents which have to vary cyclically. Each function generator is loaded, through the local computer and MPX system, with a table of up to 64 vectors defining the required pulse shape, and must supply the appropriate variation of reference voltage each cycle, without further attention from the computer. The existing function generators were built up from discrete logical units, and use single buffering, so that changes can be made only at certain times in the cycle. If these function generators were to be designed again, it would be an ideal application for microprocessors, which could result in simplification and reduction in size of the hardware. Some of the computation now carried out elsewhere, such as conversion from pulse height and time into slope and duration could be done in the microprocessor, but the temptation to put too much of the software into such a device should be resisted, as then flexibility may be lost, since all the program should be in read-only memory.

Similar devices could be used when it is required to change a number of parameters each cycle, to give a repeating pattern of cycles with different characteristics.

5.8.4 Data acquisition

Data acquisition can be made in two basic ways: The system can be designed to obtain a piece of data directly from the hardware when, and only when, it is required, or all the data can be acquired automatically at fixed intervals (e.g. once per cycle) whether required or not. The design of the SPS system was on the basis that normally data would only be acquired when required, as it was clear that to acquire and store all the data at frequent intervals would be unnecessary and would also put an unacceptable load on the computers. Instead, means were provided to acquire and transmit small blocks of data around the system at the highest practicable speed. However, there are two types of acquisition where problems occur. The first is where analogue signals have to be digitized and, owing to the accuracy required and the rejection of mains pick-up, the conversion time of the analogue-to-digital converters is relatively long. In such cases the acquisition has to be made in two stages, the computer first initiating the acquisition and conversion and then going back for the result after the appropriate interval. In the second case, it is necessary to acquire data from several devices simultaneously at a given point in the cycle, mainly from the beam instrumentation system. The solution adopted here is to use CAMAC modules that can acquire and store data when triggered from the timing system, and
then transfer the data into the computer memory via DMA. Originally, it was intended that this should be carried out only when needed, the DMA being set up for a limited number of cycles, but it was found unnecessary to go to the complication of a 'booking' system, as the load on the computer was not excessive, for the limited data required, if it was acquired every cycle.

With the addition of microprocessors to the interface equipment, it would be possible to acquire all the data from the analogue measurements every cycle, storing the results in the microprocessor's memory. When some part of the data is required for a program, the local computer could then obtain it from the microprocessor in a single action, rather than the multiple actions now required, and this could speed up some programs appreciably, particularly where the data modules are resident in the local computer memory.

The need for this increase in speed is not yet evident in the SPS system, but it may be useful for other systems where the cycle time is shorter, and it would probably be used for any future system, irrespective of the need for speed, to simplify the software at the data-module level.

5.8.5 Local servo loops

The control of local servo loops seems to be an ideal application for microprocessors, provided that the interface to the rest of the system can be kept simple. For example, the local computer could send a value representing a power-supply current to the microprocessor, which could then be left to set up the reference voltage, check the actual current periodically and compare it with that required, and then change the reference voltage if needed to keep the error below a set tolerance. This type of work is done by a small Siemens computer connected to the CAMAC on the West Experimental Area computer, but the same service could be carried out by a microprocessor.

It is likely that, for future projects similar to the SPS, the various pieces of equipment, such as power supplies, pulsers, RF amplifiers, and vacuum equipment will themselves incorporate microprocessors to carry out the logical and regulatory duties now carried out by discrete electronic and electromechanical devices. It is important that the design of such equipment should take into account the requirements for standardized interfacing to the control system, so that not only can parameters be set and data acquired, but also diagnostic information can be obtained, so that the location and nature of faults can be made available to the control system.

5.8.6 Conclusions

Summing up, it can be seen that the main impact of microprocessors on a control system like that for the SPS should be to take some of the load off the minicomputers, particularly that part where timing is critical, and at the same time allow some simplification in their software. For example, instead of a surveillance program having to acquire the status and measured values from a piece of equipment to check with the data tables, it will be able to ask the microprocessor incorporated in the equipment "Are you alright?" and get the answer "Yes" or "No, status ABl2 is incorrect and value Xy89 is out of tolerance".

The price to pay for any reduction of problems in the computer system, and for any additional facilities for self regulation and fault finding that may be incorporated into
the equipment, is the software that has to be developed and maintained for the microprocessors. Means can be provided to ease this work, such as one already available on the TSS (time-sharing system) computer in the SPS system, which include facilities for writing programs in a high-level language, interactive debugging, cross assembling, and loading of the microprocessors at special terminals provided in the development laboratories. Even with these facilities the amount of work is not negligible, and must be taken into account when assessing whether it is worth while to make a change involving the addition of microprocessors to the system. The gain is greatest where the number of identical units involved is large, such as the case of the function generators, and when the control programs, once debugged, are fixed and held in read-only memory.

5.9 Application to other purposes

The subsidiary question sometimes asked is on the following lines: "The system seems to work fine for a large slow cycling accelerator like the SPS, but how would it work for other control purposes where a faster response time is needed, and what would be the advantages of using it?".

It is hoped that this report answers the first part of the question, having shown how various parts of the process and procedures could be speeded up if required, while keeping to the basic principles listed at the beginning of this section.

The advantages to be gained from using the SPS system for other control purposes are obviously greatest when the same type of hardware can be used -- the same make of computer and message-transfer system, the CAMAC and MPX system, and the general-purpose control console -- since then most of the existing system’s software could be used unaltered, only requiring the applications programs and any additional data modules to be written. Even where this is not possible, there seem to be considerable advantages to be gained in following the same principles.

Whether the system uses a large multicomputer network or a small single computer, one can benefit from the facilities for equipment designers being able to write the control programs and the users to modify them in the light of experience without needing the services of professional programmers, and from the partition of the data base into separate, simple modules which correspond to the properties of the hardware rather than to the use to which it is put.

In the multicomputer case, however, there are the additional advantages of being able to program the complete system in the same simple way, which allows a complex system to be got working satisfactorily in a far shorter time, and with fewer programmers than would seem possible in any other way.

6. ORGANIZATION AND ACKNOWLEDGEMENTS

The work described in this and the previous report was mainly carried out by the Controls Group of the SPS during the construction period, and by the Computer Control and Operations Groups in the commissioning and operation periods.

The Controls Group staff reached a maximum of 73, of which 24 were concerned with the hardware and software described in these reports, 16 were developing and commissioning
beam-instrumentation equipment, and 27 were providing for the electronics services (drawing, prototypes, etc.), and for installation of electronic equipment and low-current cables for the whole of the SPS, with the assistance of contract and temporary labour. Extremely important was the work done by the five link-men, who belonged to the Controls Group but who worked in the five SPS Equipment Groups which had the major control problems. It was their job to assess the control requirements, keep the Controls Group informed of the problems, and to help to "sell" the new concepts of complete computer control to sometimes reluctant customers, as well as writing some of the data modules.

After the completion of construction, a reorganization of the personnel was carried out to provide an Operations Group, to which six members of the old Controls Group were transferred, to deal with the maintenance and improvements in the main control room equipment and assist in the applications work.

The Controls Group was then split into three new groups, responsible for the beam instrumentation, the electronics services, and the computer system. The last group is responsible for the maintenance and new developments in the computer network and interface system, and for the system's software and data modules. The present staff totals 23, five of these being full-time programmers.

The author wishes to express his gratitude to all who took part in this project for their enthusiasm, ideas, and hard work that enabled the control system, incorporating many novel features, to be completed on a tight timescale, with such satisfactory results.
REFERENCES


5) J.E. Midwinter, Optical communication, today and tomorrow, Electronics and Power 24, No. 6 (1978).


BIBLIOGRAPHY

Detailed descriptions of the various parts of the Control System can be found in the following CERN Reports.

COMPUTER NETWORK

Real time network for the control of a very large machine.
J. Altaber (SPS-ACC/Reprint/77-10).

Hardware description of the SPS message transfer system.
G. Beetham (Lab. II/CO/CC/Int. Note/GB/75-34).

Measurements on the control network of the SPS.

Maintaining the SPS computer control system. Present problems and future plans.

INTERFACE SYSTEM

Système de multiplexage de données du SPS. L'unité de contrôle série et son utilisation.

Handling of interrupts in the SPS-MPX system.

SPS general purpose multiplex and timing systems. Standard equipment list.

Auxiliary CAMAC crate controller using a 16-bit micro-processor.
A multicontroller approach to CAMAC for the SPS control system.

Cost comparison between serial CAMAC and the SPS MPX system for control applications.
R. Rausch (SPS-CO/Int./CC/76-6, August 1976).

SOFTWARE

Principles of the operating system for the SPS real-time control network.

The file manager for SYNTRON II -- Users' guide.
J. Altaber (SPS-CO/Comp. Note/76-15).

The NODAL system for the SPS.
M.C. Crowley-Milling and G. Shering (CERN 78-07).

Data module manual.

The FASP scanner and the FASP function user package.

Programming the new displays.
P. Anderssen (SPS/AOP/CO/Note 77-49).

OPERATOR INTERFACE

The design and construction of a control centre for the CERN SPS accelerator.
F. Beck (SPS-CO/76-1).

Two devices for operator interaction in the central control of the new CERN accelerator.
F. Beck and B. Stumpe (CERN 73-6).

The SPS microprocessor TV graphics system.
P. Anderssen, G. Shering, S. Summerhill and B. Stumpe (CERN/SPS/AOP/Note 78-29).