SLAC – PUB – 4126
November 1986
(A)

RADIATION INTERLOCKS – THE CHOICE BETWEEN CONVENTIONAL HARD-WIRED LOGIC AND COMPUTER-BASED SYSTEMS

K. F. CROOK
Stanford Linear Accelerator Center
Stanford University, Stanford, California 94305

ABSTRACT

During the past few years, the use of computers in radiation safety systems has become more widespread. This is not surprising given the ubiquitous nature of computers in the modern technological world. But is a computer a good choice for the central logic element of a personnel safety system? Recent accidents at computer controlled medical accelerators would indicate that extreme care must be exercised if malfunctions are to be avoided. The Department of Energy (DOE) has recently established a sub-committee to formulate recommendations on the use of computers in safety systems for accelerators. This paper will review the status of the committee's recommendations, and describe radiation protection interlock systems as applied to both accelerators and to irradiation facilities. Comparisons are made between the conventional (relay) approach and designs using computers.

INTRODUCTION

Designers of radiation protection interlocks have been gradually moving away from all-relay systems to solid-state logic systems and, more recently, to computer-based systems. Computers were initially used for monitoring, display and data-logging functions; the actual control of the safety system was still performed by hard-wired logic. In the last few years, however, computers have been used for both the monitoring and the control of safety interlock systems in medical accelerators, irradiation facilities and nuclear power stations.

The Department of Energy (DOE) has recently set up a sub-committee to make recommendations on the use of computers in radiation safety interlock systems for large accelerators. Many of these recommendations should have relevance to the use of computers in safety systems for other types of machines. These recommendations, which are presently in draft form and are under review indicate, in essence, that computers may be used in safety systems for accelerators when extreme care is exercised and only if high reliability for both hardware and software can be achieved and demonstrated. The computer should be dedicated solely to safety functions and should be fault-tolerant and fail-safe to the maximum extent possible. The recommendations are discussed in more detail in the body of this paper.

INTERLOCK LOGIC TYPES

A simplified block diagram of a safety interlock system is shown in Fig. 1. Typical inputs are shown on the left side of the diagram. These may include pressure and flow signals,

*Work supported by the Department of Energy, contract DE–AC03–76SF00515.

Presented at the Health Physics Systems of Radiation Generating Machines
Mid-year Symposium, Reno, Nevada, February 8–12, 1986

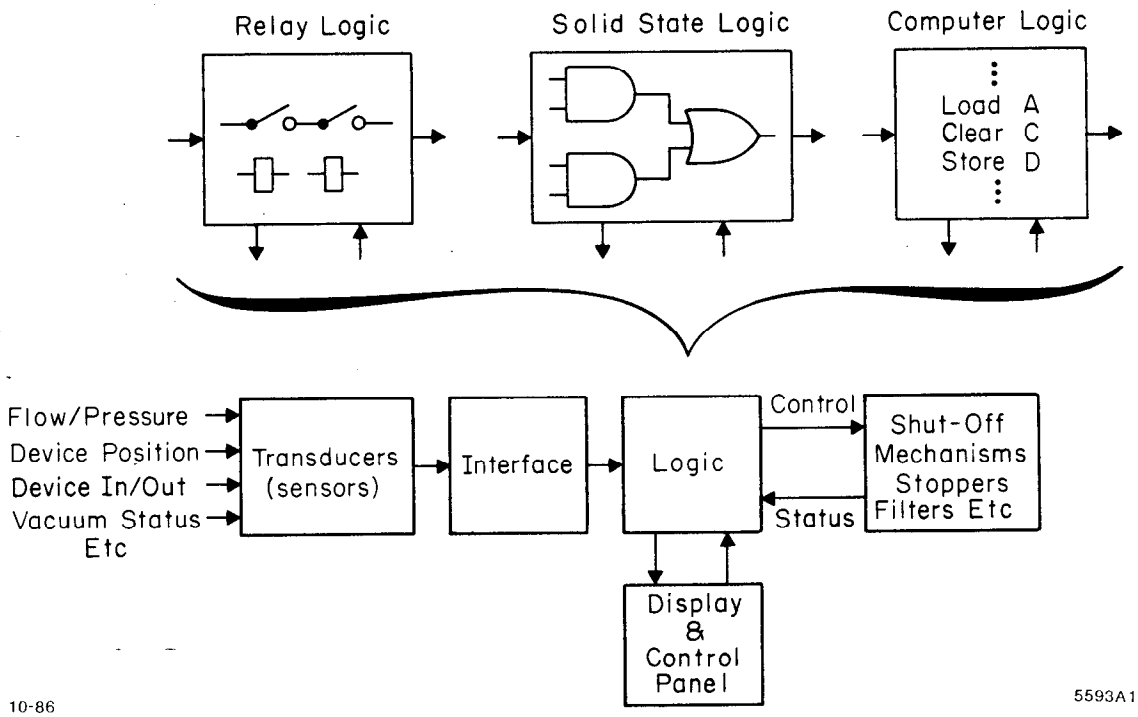

Fig. 1. Safety System Block Diagram

the position of radiation safety devices such as shutters, filters or stoppers, the open/closed condition of doors, beam current and vacuum status. Transducers or sensors convert these inputs into electrical signals, and interface modules change the signals to voltage levels that match the logic used.

The logic block has three main functions:

1. Providing all the decision-making logic.
2. Driving the display and control panels.
3. Sending signals to the shut-off mechanisms to either permit radiation or to shut down the machine.

As shown in the figure, the logic block may utilize relays, solid-state circuits or computers. These are discussed in more detail below.

Relay Logic

A simplifed circuit for interlocking a beam stopper is shown in Fig. 2.

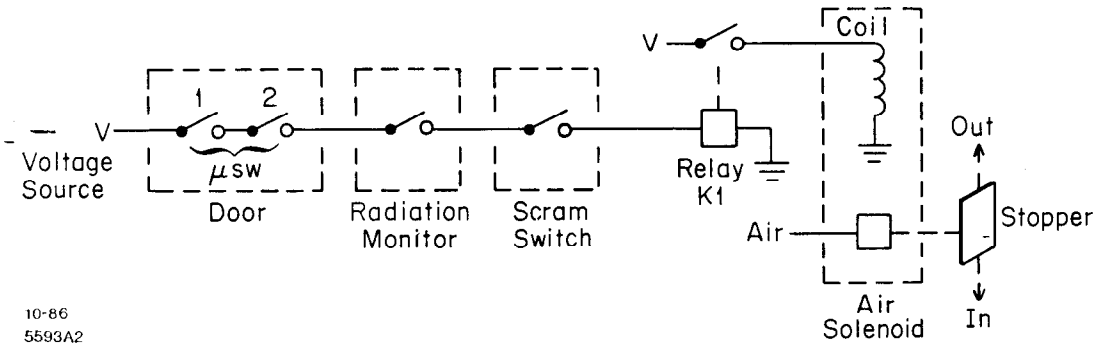

Fig. 2. Simplified Beam Stopper Interlock

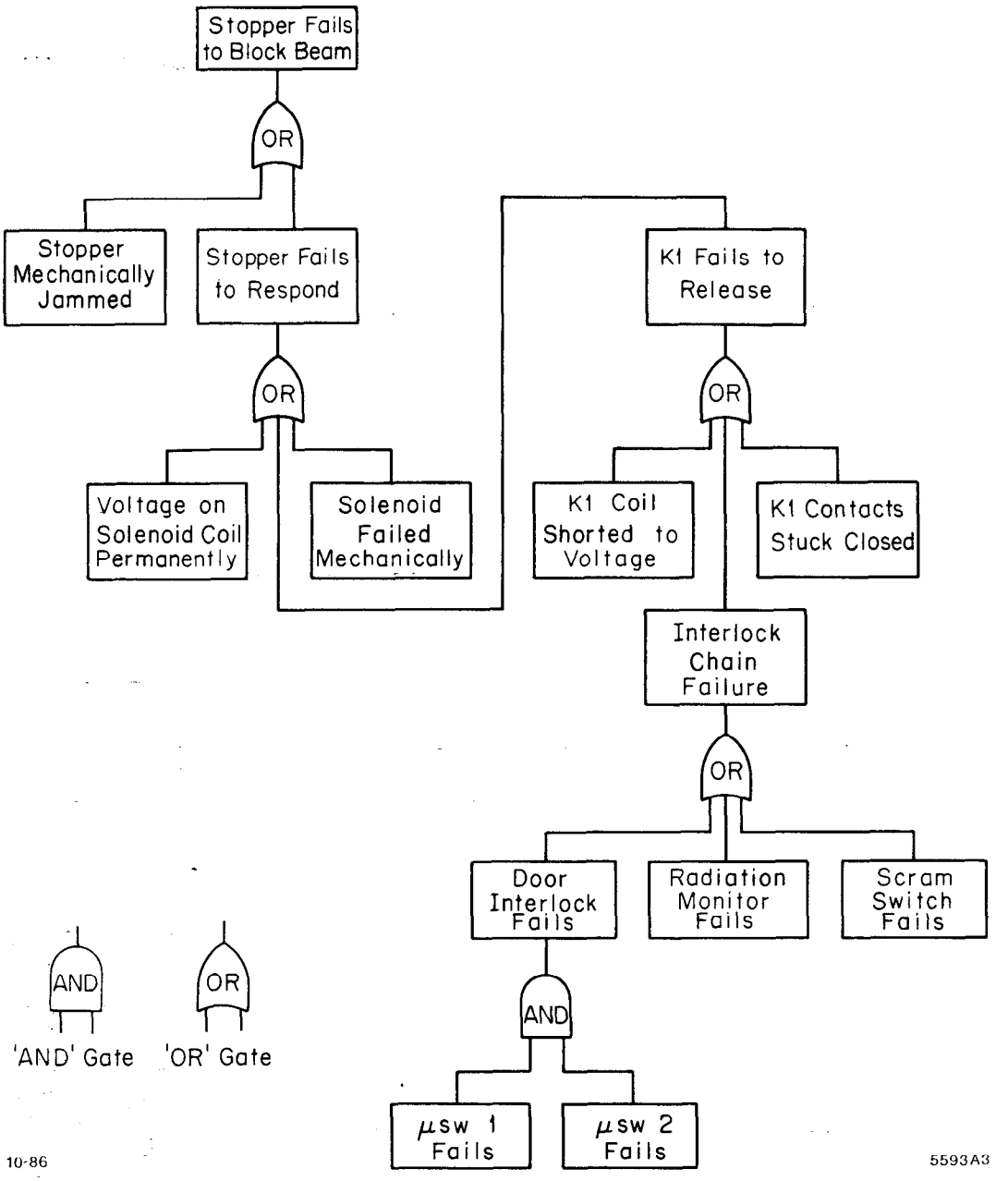

Fig. 3. Fault Tree Analysis for Circuit in Fig. 2

In the normal operating condition, all contacts are closed, relay K1 is energized, and air is applied through the air solenoid, holding the stopper in the "out" or "operating" position. If a door opens, if the radiation monitor exceeds a pre-set limit, or if the SCRAM switch is pushed, relay K1 de-energizes, the air pressure is reduced and the stopper falls into the beam line. Note that the circuit is also "fail-safe" in the sense that if either the control voltage V or the air supply to the solenoid fails, the stopper drops to the safe position.

The only redundant components in this system are the door microswitches. To render the circuit inoperative, if the door were opened, both microswitches would have to fail. A circuit such as this can be analyzed for failure modes using Fault Trees (1) as shown in Fig. 3.

Starting at the bottom, each possible component failure is shown as an input to either and "AND" or an "OR" gate. It can be seen that the only "AND" gate in the tree is the

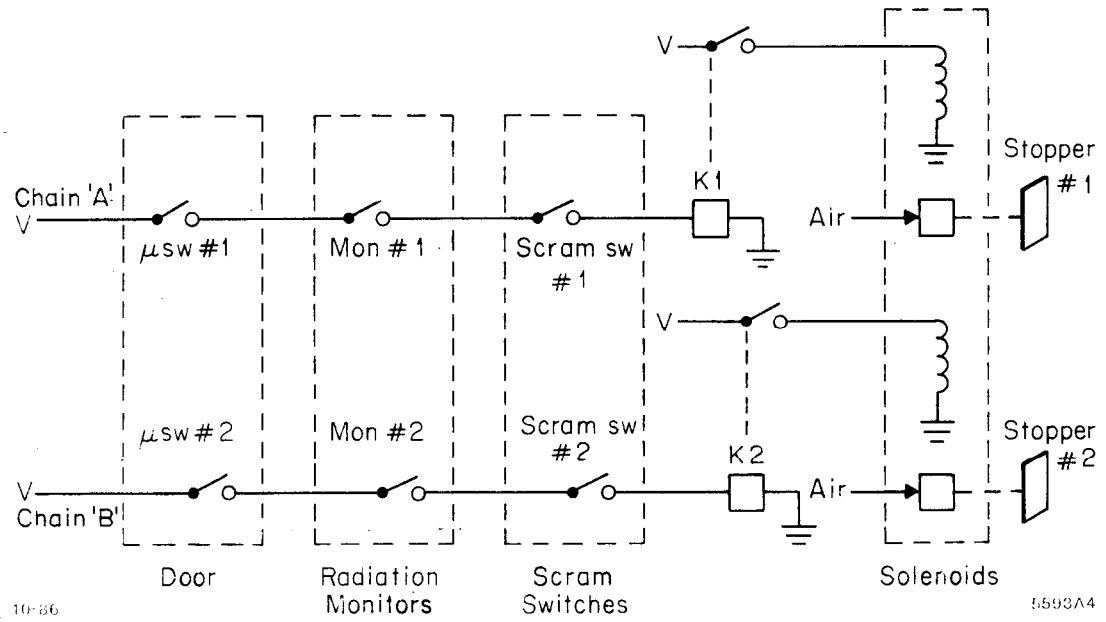

Fig. 4. Beam Stopper Interlock with Redundancy

one associated with the door microswitches. Designers of safety interlock circuits attempt to incorporate redundancy at each critical point in the system. A safer version of the circuit is shown in Fig. 4 where each component has been duplicated, including all of the interconnection wiring and the stopper mechanism itself. In some applications, triple redundancy is desirable.

Combined with frequent testing, redundancy as illustrated in Fig. 4, can achieve significant improvement in safety when measured by the probability of an accident in a given time span.

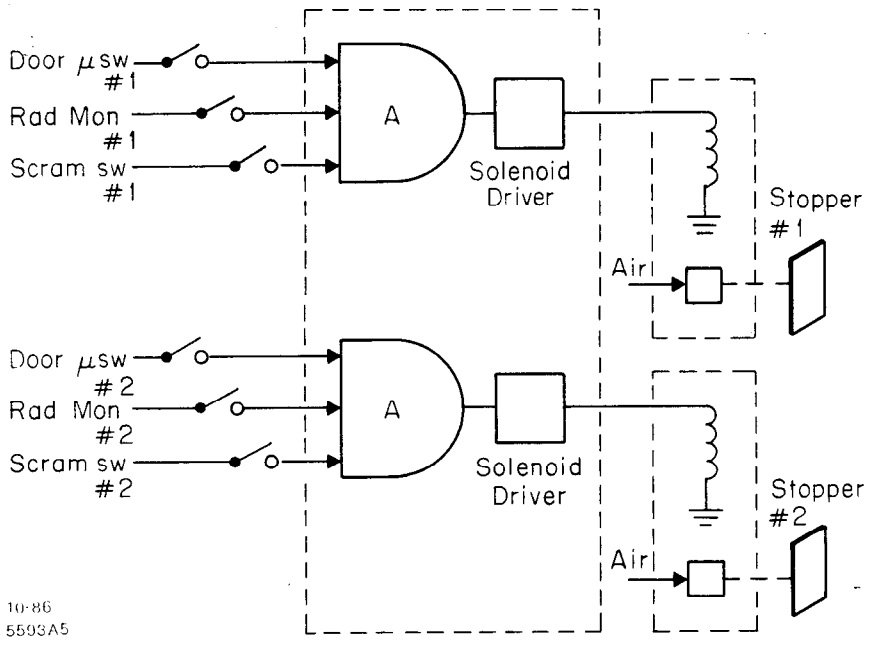

Fig. 5 Solid State Beam Stopper Interlock

Solid State Logic

Fig. 5 shows the same simple interlock circuit implemented in solid state logic. The external devices are connected to a logic AND gate which operates the air solenoid through a power transistor driver.

Solid-state circuits for safety systems have disadvantages when compared to relay circuits. Whereas a relay will most frequently fail in the open or de-energized state, solid-state components may fail in either the open or short-circuit state. Redundancy and frequent checking are therefore essential to reduce the probability of catastrophic system failure. In some circuits, it may be possible to inject pulse signals that automatically check the system integrity from input to output. (2) One such test method makes use of a repetitive "housekeeping" pulse train that is fed to the input of the circuit and is required to be present at the output within specified voltage limits. An alternative approach is to simulate a test fault by injecting a fault signal at the input and requiring the output circuit to detect the fault. During this fault-test interval (typically 1 microsec – 1 millisec), the output is inhibited from causing a system trip.

Another technique that is used frequently to reduce the probability of electrical breakdown in solid-state logic circuits is to provide isolation from transients in external circuits by the use of optical-isolators, with appropriate low-pass filtering, at all of the input and output connection points.

Computer Logic

There are a number of hardware options available if a computer is selected as the central logic element in a safety system. Regardless of the computer system chosen, high reliability and a long mean time between failure (MTBF) are essential hardware requirements.

Hardware options include:

1. Personal computers, such as the IBM PC.
2. CAMAC-based systems.
3. Programmable logic controllers.
4. High quality control computers from vendors with established reputations for producing high reliability equipment (MIL specifications).

In choosing a computer system, consideration should be given to the following:

1. Are high reliability input/output interface cards or modules available for the particular computer CPU selected? Reliability of the interface is at least as important as the reliability of the CPU.
2. Does the manufacturer have published test data on the MTBF of all hardware?
3. Does the computer manufacturer provide suitable programming languages that are fully documented?
4. Is professional customer support available?
5. Is there a second source for all hardware?

In the implementation of a computer-based system, the following points should be considered:

1. Redundancy of computer components as in relay and solid-state systems is highly desirable. This includes all elements of the system, from input to output.
2. Protection of input/output circuits from damage by external transients is as important with computer components as it is with solid state logic.
3. External watch-dog timers that continually check that the computer program is cycling are a valuable contribution to system reliability.
4. Safety interlock functions should be handled in a computer devoted to safety functions only.

SOFTWARE CONSIDERATIONS

Apart from considerations of hardware reliability in computer- based systems, there is a paramount requirement for reliable software. In real-time programming applications, inputs may change in parallel, not in sequence. The program must function correctly, to meet shut-down deadlines, in spite of these multiple input changes. Testing of programs to demonstrate high reliability and bug-free operation is difficult. Analytic techniques are sometimes employed but generally either "simulated" testing, or "real-world" testing is used. Clearly, it is difficult to simulate all possible variations on a test bench. It is equally difficult, in real-world testing, to create all of the conditions that might lead to software failure. Reference (3) describes techniques for verification and validation of real-time software. It is interesting to note that despite intensive efforts to write error-free code, 50% of all problems in Bell Labs computerized exchanges are due to software problems, while 20% are hardware and the remaining 30% due to operations and maintenance errors (4).

EXAMPLES OF CURRENT PRACTICE

Large Accelerators

Most large accelerators use relay systems for access control and alarm systems. A few use solid-state circuits for interlocks, but only one, to the author's knowledge, uses a fully computerized safety system. This is the Pulsed Beam Fusion Accelerator (PBFA II) in Albuquerque NM. This facility, which started operation in early 1986, uses a single Hewlett Packard computer for safety interlock and access control functions.

Gamma Irradiators

Whereas relay systems have been used extensively in the past, there is certainly a trend towards the use of computers–particularly programmable logic controllers (PLC)–for safety systems. This is understandable given the widespread use of PLCs for control of the conveyer systems of large irradiators. It may be tempting, for reasons of economy, to incorporate the safety functions into the same PLC that is used to control the conveyer system. This would be highly undesirable. In fact, a strong case should be made for redundant safety computers (as noted previously) that are quite independent of the conveyer computer.

Medical Accelerators

The recent accidents involving AECL medical accelerators have focused attention on the issue of reliability and safety of computer controlled safety systems for these types of machines. While it is clear that a computer can be a valuable adjunct to a medical accelerator for record keeping, data logging and display functions, it is not as clear that

the computer should be the sole device to determine if shut-down is necessary, or that it should be the sole path to the shut-down mechanism. It could be argued that external, independent hardware should be used in parallel with the computer as a back-up to the computer shut-off. Alternatively, one might consider using a second, redundant computer, with independent, isolated sensors, interfaces and shut-off mechanisms.

Nuclear Power Stations

One of the more surprising developments in safety system design is the use of computers in shutdown systems for Canadian CANDU reactors (5). The normally conservative nuclear industry has been using computers in power plants for a number of years, but until recently they were used only for tasks such as data monitoring and logging, CRT graphics and equipment control.

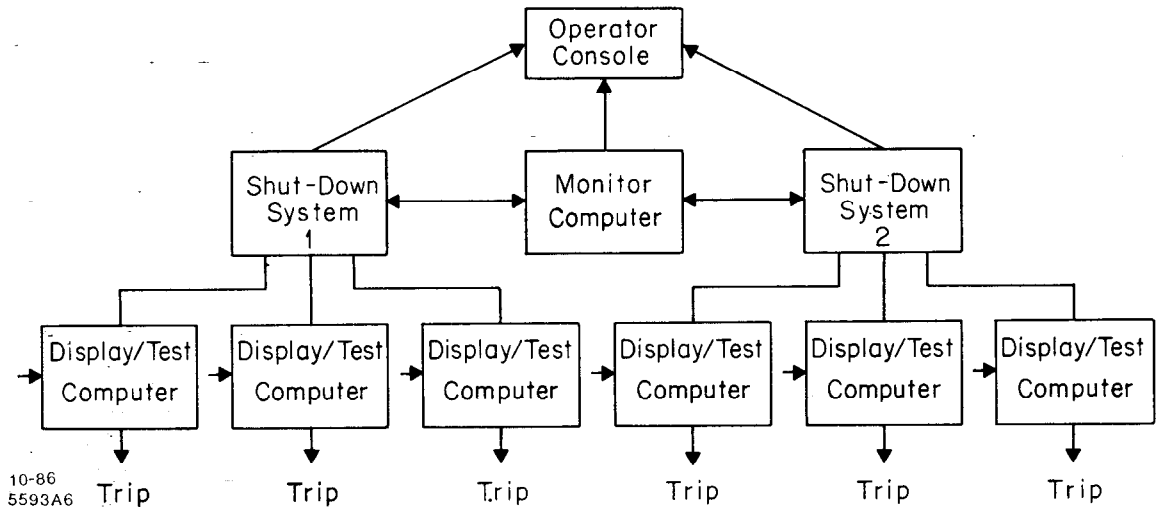

Fig. 6. Fully Computerized Shut-Down System Nuclear Power Plant

Starting in 1987, CANDU reactors will have a fully computerized shutdown system. In this design, there are two independent computer systems, each with three independent and redundant channels of instrumentation as shown in Fig. 6. The outputs from these three channels are combined in a two-out-of-three voting scheme which is used to initiate shutdown. A total of fifteen computers is used, each isolated from one another by fiber optic links operating at 19.2 K bits/sec. The anticipated benefits offered by these computerized systems include:

- lower costs
- greater flexibility
- improved testing
- improved operator interface
- lower down-time

DOE SUB-COMMITTEE RECOMMENTATIONS

DOE has set up a committee to write a code of good practice for accelerator health physics (6). A sub-committee under the main group has been studying the use of computers in safety interlock systems and has prepared recommendations which are currently under review. Following is a summary of the sub-committee's recommendations:

1. It is permissible to use computers in safety systems only if high reliability can be achieved. An unavailability of 0.001 has been suggested as being reasonable, but this number may be changed. (.001 unavailability means that the safety system must be designed and routinely tested to insure it will operate 999 times out of 1000 attempts.)
2. Choose computers for safety systems only if there is an "good" reason. Examples include:

- if the system is sufficiently complex that a very large number of conventional relays or solid state components would be required
- if speed of response is important
- if operational flexibility is an important factor.

3. Use computers only if full-time professionals are available for the design and testing of the hardware and software.
4. The designers should be familiar with the accelerator operation, be sensitive to the requirements of safety systems and of course, have competence in computer technology.
5. High reliability is essential. Redundant hardware is encouraged.
6. Dedicate the computers in safety systems to that function alone.
7. Programs should reside in Read Only Memory (ROM) or Programmable Read Only Memory (PROM).
8. Use software of high quality. Programs should be simple, modular, testable and fault-tolerant.
9. Use hardware watchdog timers to check on continued cycling of sequential programs.
10. Carefully control program changes by using passwords or keys. All changes must be thoroughly documented and tested.
11. Utilize non-identical, but functionally equivalent programs, when parallel, independent computers are used.
12. Test all elements of the system regularly.
13. Provide hardwired operator over-ride switches.

CONCLUSIONS

Computers are here to stay and designers of safety interlock systems must adjust to their presence sooner or later. The decision to incorporate a computer into the safety logic should not be made lightly. Good reasons for moving away from proven, reliable hardware techniques should exist. If the decision is made to use computers in the safety system, careful consideration should be given to the selection of reliable hardware, and to the degree of hardware redundancy that is appropriate to the application.

The software aspects of the system must be given very high priority. It would be dangerous to embark on computer interlocking unless proficient and experienced designers of real-time software are available for the duration of the project. The programmers must be familiar, not only with real-time techniques, but with the hardware and operating characteristics of the radiation-producing machine itself. Finally, computer control of safety interlocks requires a high degree of collaboration and cooperation between hardware and software designers if the project is to be successful.

References

1. J.M. Joller, "Constructing Fault Trees by Stepwise Refinement," IEEE Transactions on Reliability. Vol R-31 No. 4 Oct. 1982.
2. T.N. Constant, K. Crook, and D. Heggie, "Operational Experience with SLAC's Beam Containment Electronics," IEEE Transactions on Nuclear Science. Vol NS-24, No. 3, June 1977.
3. W.J. Quirk, VERIFICATION AND VALIDATION OF REAL-TIME SOFTWARE, Springer-Verlag, 1985.
4. W.N. Toy, "Fault-Tolerant Design of Local ESS Processors", Proceedings of the IEEE Vol 66 No. 10, October 1978.
5. R.S. Gilbert, "Control and Safety Computers in CANDU Power Stations", IAEA Bulletin, Autumn 1985.
6. "A Guide to Good Practices for DOE Accelerator Health Physics", R.C. McCall, Chairman (in preparation).