Fermi National Accelerator Laboratory

FERMILAB-Conf-93/280

Techniques for Increasing the Reliability of Accelerator Control System Electronics

Jeff Utterback

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

September 1993

Presented at the *1993 Particle Accelerator Conference*, Washington, D.C., May 17-20, 1993

Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CHO3000 with the United States Department of Energy

Disclaimer

Techniques for Increasing the Reliability of Accelerator Control System Electronics

Jeff Utterback
Fermi National Accelerator Laboratory*
POB 500, MS 307, Batavia, IL 60510 USA

Abstract

As the physical size of modern accelerators becomes larger and larger, the number of required control system circuit boards increases, and the probability of one of those circuit boards failing while in service also increases. In order to do physics, the experimenters need the accelerator to provide beam reliably with as little down time as possible. With the advent of colliding beams physics, reliability becomes even more important due to the fact that a control system failure can cause the loss of painstakingly produced antiprotons. These facts prove the importance of keeping reliability in mind when designing and maintaining accelerator control system electronics.

I. RELIABLE HARDWARE DESIGN

In order to produce reliable circuit boards, the boards must be designed with reliability in mind. Components with high rates of failure must be avoided and good hardware design practices must be followed.

A. Chip Sockets

It is wise to avoid using chip sockets unless absolutely necessary. At Fermilab, chip sockets have been known to corrode after a few years in service. The corrosion is not visible from the exterior of the socket, but can be seen on the legs of the chip when removed. The reason for the corrosion is not definite but it may be caused by a reaction between two dissimilar metals.

When corrosion occurs, the socket loses contact with the legs of the chip causing an intermittent failure which is hard to diagnose. The failure will suddenly clear up if the chip is physically disturbed because it reestablishes contact with the socket. This type of thing is very confusing to a repair technician who can't understand how he suddenly fixed the board by simply touching some pins with a scope probe.

The faulty contact usually occurs on the ground pin of a chip socket because the ground pin carries the most current. The corroded joint acts like a resistive connection. Voltage drops of up to 3 volts have been observed between the chip's ground pin and the board's ground plane. Obviously, I.C. chips will not function properly when their ground pin is not held at ground potential.

To avoid socket unreliability, it is wise to solder all chips directly to the circuit board unless the chip is subject to change, such as an EPROM or a programmable logic device. If a socket is necessary, only a high quality socket should be used. It is better to spend some extra money on a high quality socket than to have reliability problems in the future.

B. Polarized Capacitors

Polarized electrolytic capacitors are a high failure item. They should be avoided unless absolutely necessary. After a few years in service a polarized capacitor may develop an internal short. At Fermilab, shorted capacitors have been know to cause blown fuses, power supply failures, and in some cases circuit board fires.

Many designers use polarized capacitors as a rule of thumb in the power supply section of a circuit board. The rule of thumb should be closely examined to see if an alternative component, such as an I.C. regulator, or an unpolarized capacitor can be substituted. Of course in some designs there is no viable alternative to using a polarized capacitor. In these cases the designer should spend some time finding capacitors which have a good history of reliability.

C. Components which Need Calibration

In order to produce a reliable circuit board, it is wise to avoid components which need calibration. Components which need calibration are a maintenance headache and if calibration is not done on a regular basis, the performance of the circuit board will degrade or the circuit board may fail in service.

Some designers use variable resistors to set timing or voltage thresholds. After a period of time, the resistor may need to be calibrated to keep the circuit performing well. Also, there is some danger that the position of the variable resistor may be disturbed by someone mishandling the board.

If possible use components which need no calibration. For timing circuits substitute "delay line" I.C.s in place of "one-shots". For comparators, use zener diodes in place of variable resistors.

D. Protected Inputs and Outputs

A circuit board often produces an output which drives an external device or relies on some input signal from another circuit board. The board designer can not trust the external peripherals to perform correctly. The board must be designed with protected inputs and outputs.

In order to perform reliably the inputs should be able to withstand an overvoltage and the outputs should be able to withstand a short circuit. Otherwise, the board may have a high rate of failure, especially when being newly installed with untried external peripherals. There are several devices available which provide protection, such as optically coupled input I.C.s and short circuit protected output drivers.

At Fermilab, a circuit board is quite often designed for one purpose and ends up being used for several different purposes. The inputs and outputs should be protected for all contingencies because the board may be used in a situation that it was never designed to be used in.

E. Component Specifications

In order to produce a reliable circuit board, it is wise to study and heed the component minimum and maximum specifications. Specifications such as maximum input frequency, maximum operating temperature, and minimum

*Operated by the Universities Research Association under contract with the U.S. Department of Energy.

input voltage are determined by the manufacturer and should not be taken lightly. Sometimes designers ignore the specifications during the prototype phase only to find out that the components fail when subjected to adverse conditions in service.

It is wise to avoid components that are brand new on the market and have not yet been proven reliable. It is more fun to use the latest and greatest thing, but designing for reliability dictates the use of tried and true components. Sometimes a circuit is designed around a component that is so new, it only exists on a marketing advertisement. This is know as a "Vapor Component", and can cause frantic redesign efforts when it fails to materialize.

F. Built in Diagnostic Aids

When designing a circuit board it is wise to include built in diagnostic aids. Things such as LEDs, test points, and terminal ports can be invaluable to a repair technician when diagnosing a failure in the repair lab or in the field.

Sometimes a systematic failure will occur in the field. Diagnostic aids on the front panel of a circuit board can be the difference between a one minute simple solution or a half day ordeal.

LED indicators are a cheap, simple, and valuable way to indicate the state of the circuit board. They can be used for indicating the presents of input signals, the state of the CPU, and any other status information. Test points are valuable for checking the presence of voltages on the board and for looking at important signals with an oscilloscope. If the board contains a CPU, it may be wise to include a front panel RS-232 port so that diagnostic information can be exchanged with the CPU using a terminal.

II. RELIABLE SOFTWARE DESIGN

Many circuit boards designed for use in accelerator control systems have on board microprocessors. Several techniques can be used when designing the microprocessor software to improve the reliability of the circuit board.

A. Self Tests

Self tests should be incorporated into the software so that the circuit board can proclaim its own readiness to be put to use. Self tests can improve the reliability of circuit boards by finding components which have failed, and by finding components which are soon to fail. Self tests are fast, accurate and complete. They make life much easier for the repair technician.

Self tests are usually performed upon power up or reset. If the self test finds a problem it must have some way to proclaim the error. This can be as simple as sounding a beep, or lighting a light. The self test should not only proclaim the error but also give some information about the nature of the error. One simple way to do this is by flashing a code on an LED. For example, flash pause flash pause means the RAM is dead. Flash flash pause flash flash pause means the I/O chip is dead. It is also a good idea to proclaim the results of the self test in a status register. This will allow diagnosis from a remote location provided the status register is not the broken component.

When the self test routines are written, keep in mind that the self test must be able to run with a minimum of working components. The self test can not assume that the RAM is working, so it must be able to run completely from PROM without using the stack. The self test should be designed to run with only the CPU and the PROM working.

Self tests should be performed on all components that the CPU has access to. RAM tests can be done using some systematic scheme which writes to and reads from every address location. RAM tests should check for bad data bits as well as bad address lines. It is wise not to write the same data to every location because a stuck address line could be overlooked.

The PROM itself can be tested by doing a check sum self test. When the PROM is burned, the check sum can be manually entered and burned into the last address. During the self test, the CPU adds up the contents of the entire PROM and compares it with the last location.

Other miscellaneous components should also be tested. Command/Status registers can be tested by writing some data and reading it back to confirm. DAC/ADC pairs can be tested by writing a value to the DAC and reading it back on the ADC. Some items such as a "write only" register are hard to test. The software designer should work with the hardware designer to make the hardware self testable.

B. Signs of Life

Hardware and software should be designed as reliably as possible, but in the event of an unforeseen glitch or bug, the CPU may malfunction. These malfunctions can cause the CPU to go senile, (lose the data in memory) go crazy, (execute data instead of instructions) become compulsive, (execute an infinite loop) or die (cease to run at all). It is wise to add special hardware to detect a malfunctioning CPU and to show the state of the CPU to the outside world. This hardware can then be used to find problems and improve the reliability of the hardware and software.

One way to determine the health of a CPU is to check its pulse. At Fermilab, it is common practice to include a heartbeat circuit on smart circuit boards. The CPU must periodically access the heartbeat circuit in order to keep the heartbeat LED on and the heartbeat status bit asserted. If the heartbeat circuit times out, it indicates that the CPU is malfunctioning.

A similar scheme to the heartbeat circuit is the watchdog circuit. The watchdog detects the death of the CPU when it fails to be periodically fed (accessed). When the watchdog gets hungry it bytes (resets) the CPU. When designing a watchdog it is wise to make it a finicky eater. The periodic access should be a specific data code written to a specific address. If the watchdog ignores the data and depends only upon a chip enable, the CPU may still be able to randomly feed the watchdog even if it is in a "crazy" state.

Another diagnostic aid used at Fermilab is called Age. The CPU keeps a running total of the time elapsed since it was last reset. The age can then be read from a register by anyone interested. Age is useful for determining reliability. If a smart circuit board keeps resetting every day or two, steps should be taken to improve its reliability. A reliable smart board should be able to run for several months without resetting. Anything over 6 months is hard to achieve due to the inevitable lightning glitches and AC power failures.

C. Exceptions and Interrupts

All CPUs provide a way for external hardware to request service from the software. This is know as exception processing and is used quite frequently in control system electronics. Due to its complicated nature, incorrect exception processing is the source of a great many reliability problems in smart circuit boards.

Most smart boards do not utilize all of the CPU's available interrupt and exception vectors. The software designer should not leave the unused vectors blank. This may cause the CPU to go astray if one of the unused exceptions unexpectedly occurs. It is wise to provide a graceful return from each and every exception vector in order to improve the reliability of the circuit board. Another good idea is to somehow make it known when an unexpected exception occurs. This will allow the designers to track down problems with the design.

Most CPU's provide a nonmaskable interrupt input. If this interrupt occurs, the program must stop what ever it is doing and service the interrupt. It is wise to use this interrupt only for catastrophic events such as impending power failure. It is not a good idea to use this interrupt for mundane tasks such as servicing a timer. The software becomes very complicated when dealing with nonmaskable interrupts and this complication leads to bugs and unreliability. The complications are caused by nonmaskable interrupts occurring at a point in the program when the CPU can not afford to be interrupted. It is best to use maskable interrupts in most situations so the software can choose to delay servicing them if the CPU is busy in a critical part of the program.

D. Good Programming Practice

In order to produce reliable software, it is important to use good programming practices. With the advent of high level languages, this is just as important as it was with assembly language. Document routines with comments to make the purpose and operation known. Split the program up into small routines with a specific purpose. This makes the program less complicated and enables code reuse. Use good naming conventions for routines and variables. Long names are encouraged in high level languages to make the program self documenting.

III. MAINTENANCE

After the hardware and software design is finished and the circuit board is in use, the reliability of the board depends on efficient trouble shooting and good failure tracking. If the circuit board has a high rate of failure, attention should be devoted to making changes for improving the reliability.

A. Tracking Failures

It is a good idea to keep a record of failures for electronic circuit boards. The record can be used to track reliability and to aid in repairing boards with similar types of failure. The record should include all pertinent information such as the date of failure, location of failure, and the board's serial number. The record should also include the failure symptoms and the diagnosis.

There are several commercial database programs available which are ideal for keeping records of failure. They have customizable data entry screens and hardcopy report generation. The database programs also have nice information searching facilities which work great for finding failure trends. The database operator can do queries such as "Show me all the boards which failed in January of 1991." or "Show me all the boards which have failed due to a shorted capacitor."

B. Computerized Test Station

In order to produce a circuit board with a high degree of reliability, the board must be tested comprehensively during the design phase. Also, failures which occur while in service must be diagnosed quickly. A computerized test station is a good way to meet these goals.

Most modern control system circuit boards reside in a commercial bus crate such as CAMAC, VME, or Multibus. The circuit boards are designed to send and receive data to and from a bus master over the bus backplane. In order to test a circuit board, there must be some way to to talk to the board over the bus backplane.

At Fermilab, it is common practice to connect a PC computer to a test crate via a PC to bus interface. Interface hardware can be obtained from commercial companies who specialize in this area, most notably BiT3 Computer Corp.* Programs, which run on the PC, can then be written to test circuit boards. PC computers are the obvious choice because they are readily available, relatively inexpensive, and have a wide selection of compilers and programming aids to choose from.

Programs written to test a circuit board should be flexible and easy to use. The tests should generally keep repeating until halted by the operator. Constantly repeating tests are valuable for providing stimulus when using an oscilloscope for diagnosis.

Sometimes a circuit board may develop an intermittent failure. This type of failure is very hard to diagnose because it may take hours or days for the board under test to fail. The computerized test station is very valuable for finding intermittent failures because it can perform tests, unattended, 24 hours a day, and create a log of failures for the repair technician.

Sometimes when a board fails in the field it may seem to work fine when tested in the repair lab. If this happens the tests must be made more rigorous in order to find the problem. Long term computerized tests may be needed. Technicians should not be content to put a board back into service without first finding and fixing the reported error.

---

*BiT3 Computer Corp. 8120 Penn Avenue South, Minneapolis, MN, 55431-1393 Phone (612) 881-6955