COMPUTER CONTROLLED VACUUM CONTROL SYSTEM
FOR SYNCHROTRON RADIATION BEAM LINES

S. M. Goldberg, J. Yang, and C. Wang
Stanford Synchrotron Radiation Laboratory
SLAC Bin-69, P.O. Box 1349
Stanford, CA 94305

Summary

The increasing number and complexity of vacuum control systems at the Stanford Synchrotron Radiation Laboratory has resulted in the need to computerize its operations in order to reduce lower efficiency of operation. Status signals are transmitted through digital and analog serial data links which use microprocessors to monitor vacuum status continuously. Each microprocessor has a unique address and up to 256 can be connected to the host computer over a single RS232 data line. A FORTRAN program on the host computer will request status messages and send control messages via only one RS232 line per beam line, signal the operator when a fault condition occurs, take automatic corrective actions, warn of impending valve failure, and keep a running log of all changes in vacuum status for later recall. Wiring coats are thus reduced and monitoring conditions can be monitored without adding excessively to the complexity of the system. Operators can then obtain status reports at various locations in the lab quickly without having to read a large number of meter and LED's.

Hardware

The receiver/transmitter card uses a 24-bit dual word length serial data protocol. There are 16 bits of data and address or control command and 8 bits of protocol overhead including a 4-bit checksum for hardware errors checking. There is an over error checking as it will catch many multi-bit errors. The receiver/transmitter card will then pass all messages to the host computer via a single RS232 line running at 9600 baud. Since a single receiver/transmitter card can handle up to 256 remote cards connected serially via twisted pairs, then a single twisted pair which formerly sent only one input/output status signal can now handle up to 256 times 16 signals. In practice only a few addresses are needed per beam line, but this still gives us a substantial savings in wiring costs.

The host computer is a DEC 11/34 with a 8-line D2-11 RS232 serial line multiplexer. This is used for both connecting terminals to the computer and for communicating with the serial digital party line communications not described above.

Software

Since there are many beam lines at SSRL and because we wanted the polling section of the program separated from the polling sections and logging sections of the program, we used a multitasking operating system, DCE's RSX-11M. Since the few truly highly dependant features are still handled in the vacuum electronics hardware itself, all code was written in FORTRAN and used the regular RSX terminal driver to interface to the RS232 lines. Most importantly all messages to the receiver/transmitter card were in 8-bit ASCII format. The WITIO system was used in connection with the RSX standard full duplex terminal driver. The read-after-prompt with read-all (8-bits) and read-no-echo function code was used in addition to the write-all (8-bits) function code with the RSX terminal driver. In addition the terminal characteristics were set to slave and nowrap on the terminal lines used as data lines to keep out unprocessed input and spurious carriage returns.

For a given beam line there are two separate major programs. One continuously polls the main status panel to see if there are major changes. These include checking to see if there are fault conditions, seeing if the manual override is on, seeing if a remote control panel has been activated by an operator, and noting that a valve has opened or closed. In addition at set times of the day the polling program wakes up the master control program. The master control program then takes the appropriate action which could be determining what caused the fault condition, resetting the control panel after a fault, automatically opening a beam line if closed, and if a manual override is engaged, allowing an operator to open a beam line at a remote panel instead of the local one, logging causes of faults and changes in valve states, and routine reading and logging of pressures. The master control program also keeps in memory the last status conditions so that when

© 1983 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to others or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
awakened it will be able to note all differences from the control panel's last recorded state.

The remote panel control by the operators is actually a semi-automatic operation. The computer is actually opening the beam line vacuum system but executes each step only after the operator throws a switch. But if the operator throws the wrong switch the computer will continue to wait for the correct one, allowing an additional element of safety. The operator must open the line correctly but can do so at any speed needed in order to watch for any possible sudden failures for which the opening of the line may be aborted. These remote panels can also be located far from the local control panel since all the panels are actually doing is sending status messages over the serial digital communications net. This allows operators to open and close beam line at a convenient geographic location such as the main control room for the synchrotron ring.

The polling program is kept small and fast. The master control program is large and though it is always alive it is hibernating most of the time via the RSX stop directive. The polling program then unstops the master control program only when there is a change in some status condition. Thus when the master control program is not actually doing something it can be checkpointed and other programs can be allowed into memory in the 11/34. Both programs when run for the first time read an input file created by any standard text editor that gives English-like commands for determining how often to poll, which valves are at what address and at which bits within that address, what the proper startup sequence is for a beam line, etc. Thus one can easily rearrange the cabling or change the way a beam line is opened simply by editing a text and restarting a program rather than actually rewriting the program. The file also includes lists of possible fault conditions to check once the hardware notes that there is a fault. This flexibility makes the adding of additional beam lines much simpler and allows minor changes to be performed without the need of a programmer.

Other programs will be written that will take the logging files and make useful statistics out of them. Such statistics will include noting how often valves have been opened or closed in order to predict when they are ready to fail. The fault statistics can be analyzed to determine what types, how often, and when faults have happened. The pressure readings can also be analyzed to determine how the vacuum changed over time.

References
