IEEE TRANSACTIONS ON MAGNETICS, VOL. MAG-17, NO. 5, SEPTEMBER 1981

CONTROL SYSTEM FOR A SUPERCONDUCTING RECTIFIER USING A MICROCOMPUTER

Herman H.J. ten Kate, Dick A. Kamphuis, Meindert Caspari,
Louis J.M. van de Klundert and Zweitze Houkes

Abstract - Within the scope of a research program of superconducting rectifiers software is being developed to take care of the control of such systems. The hardware architecture which interferes with the in and output signals is based on a LSI-11/2 microprocessor with sufficient mass storage for data logging, console and printer. The flexibility inherent to this hardware configuration is desired for optimalisation of the rectifier concerning maximum current, power, efficiency and quench stability. The paper describes the structure of the program and the interaction between both computer hardware and software and the superconducting rectifier. However, because the reliability of computersystems is unsatisfactory an additional hardware protection system still handles the most important alarms.

INTRODUCTION

During a development program for high current (10-100 kA) superconducting rectifier-fluxpumps attention has to be paid not only to the rectifier itself but also to an adequate control system. Moreover, the magnetude of the stored energy pumped into a coil by the rectifier, obliges a fully reliable system to protect the rectifier and its surroundings in case of a quench. It is even possible that the control and guarding system can be designed so that it protects the very sensitive superconducting systems to quench in case of human failures. Furtheron it has to be possible to use several pumping scenarios such as charging, decharging, keeping the load current constant for a moment or even for years etc. A microcomputer can take care of these and many other functions. In order to get more flexibility a microcomputer has been prefered to investigate the performance of such a system in connection with real superconducting devices. In this paper the interaction of hardware and software with the rectifier will be presented.

SUPERCONDUCTING RECTIFIERS

The principle of the superconducting rectifier, known for more than twenty years [1], is shown in figure 1. A (special shaped) a.c. current is fed into the primary coil of a transformer. By periodically opening and closing the switches S_1 and S_2 at proper times, a current is induced in the load coil which increases every half cycle. A maximum is set by the parameters of the system or by the critical current of the conductors. This is an analogue of a normal conducting rectifier where instead of the switches S_1 and S_2 diodes are used. Between two pumping periods both switches are closed for inductively commutating the already pumped current from one secondary branch to the other. This way of operation, called the inductive commutation mode permits the best efficiency of the rectifying process but forces special shaped control signals. Two additional components of the secondary circuit protect the rectifier switches and the secondary coil windings against damage in case a quench occurs. Then the protection switch $S_{prot.sw.}$ disconnects the rectifier part from the coil and so forces the current through the dump resistor R_{dump} where the greater part of the magnetic energy content of the system is dissipated [2].

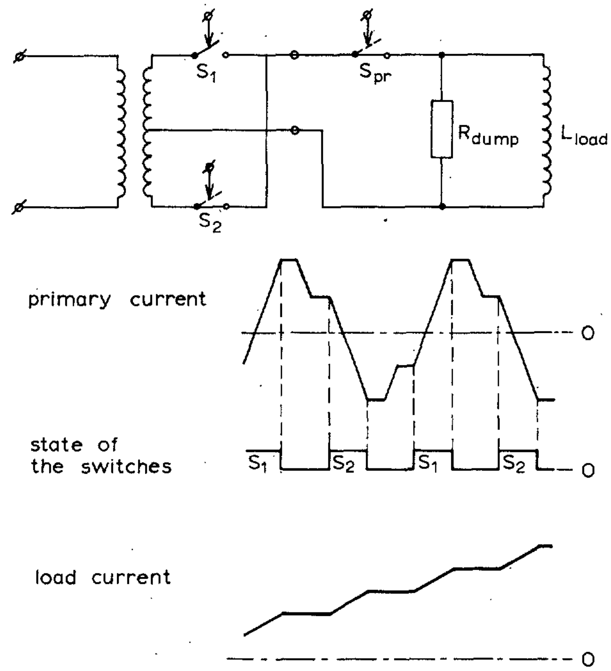

Fig. 1. Full wave superconducting rectifier with its input primary current, state of the switches and the output current in the load coil.

FUNCTIONS OF THE MICROCOMPUTER

For optimal operation of different types of superconducting rectifiers many complicated functions have to be realized. The investigation character also requires continuous administration and registration of many variables being input and output. For these reasons a realization in modular programmed software has been chosen. An exception has been made for the protection task, which has to be fail safe or sufficiently redundant hardware. In a more definite situation the software realized functions will be stored in PROM. These software functions are:

- generation of the signals to control the primary current and the rectifier switches.
- measuring the load current to determinate commutation step, switch status, voltages in primary and secondary circuit, temperature in switches and at other places, etc.
- alarm handling: detection of helium level, power up, power back up, quenches etc.
- data logging and administration.
- continuous visualisation of the most important parameters and visualisation of logged data on request.
- direct data processing to show pump performance concerning load power, load time, efficiency, stored energy etc.
- pumpscenarios.
- initial tests.

Work supported by the Foundation for Fundamental Research on Matter, Utrecht, The Netherlands.
The authors are with University of Twente, Department of Applied Physics, Z.H. is with the Department of Electric Engineering, P.O. Box 217, 7500 AE ENSCHEDE, The Netherlands.

All these functions are achieved with an interrupt based software structure which will be discussed below. An important advantage of the software realization is the initial checking of the well operation of the system and its initial conditions before start up, including a check of the authorization of the operator.

HARDWARE CONFIGURATION

The computer configuration shown in fig. 2, consists of a LSI-11/2 microprocessor, an eight-quad backplane, 32 k of RAM, two 8" double density floppy disc units, a programmable clock, a video-display with keyboard, a printer and a plotter.
The programmable clock determines, in a normal functioning rectifier-system, the real-time sequencing of the program. The floppy disc units are used for program and data store.
Video-display and keyboard form a flexible medium for computer-operator interaction and communication. Interaction between computer and rectifier is realized by 12-bits AD- and DA-convertors for the analogue signals and by a 16-bits parallel-interface for the logic signals.
All computer-inputs and outputs are protected against over-voltages by isolation-amplifiers, or opto-couplers.

The various hardware components for controlling and testing the rectifier-process are shown in fig. 3. They will be described in some detail below.
The main part is the power-amplifier governing the primary current. It is controlled by the analogue signal which is generated by the computer-program. This will be discussed in more detail in the paragraph on software.
A switch, controlled by the primary circuit quench detector, interrupts control of the power-amplifier in case of a quench. The quench detector of the secondary circuit triggers the power supply of the protection switch. Both quench detectors and the protection-switch supply are battery supplied, in order to secure operation in case of a main-failure.

The rectifier-switch-supplies 1 and 2 are controlled by the computer.
Proper operation of these switches is verified by the computer using the logic signals generated by the switch status detectors.
Remote controlled switches provide turn on and off by computer of the power supplies.

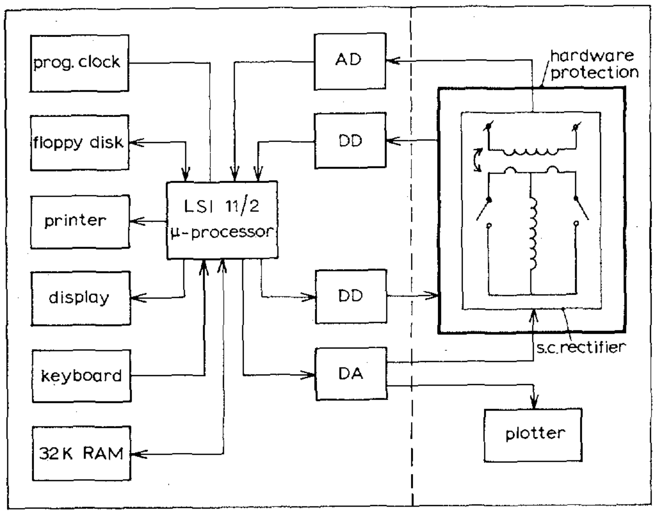

Fig. 2. Computer hardware configuration.

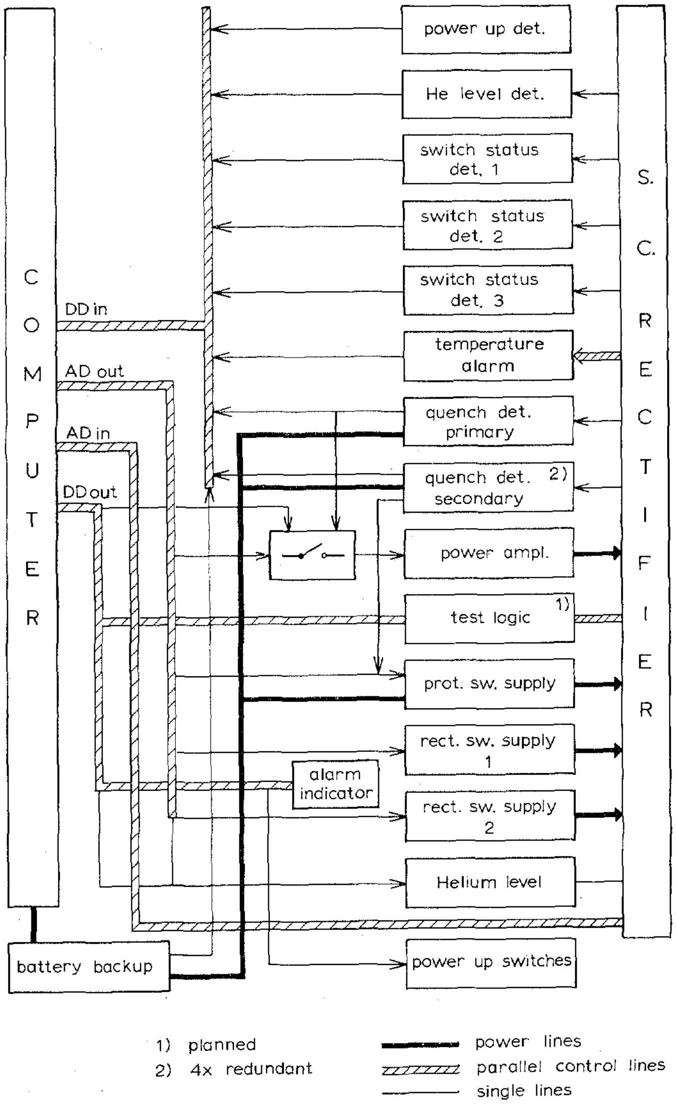

Fig. 3. Signal conditioning between computer and rectifier.

Too low He-level, too high temperatures in the circuit or quenches in the circuit including the load coil, generate logic signals which interrupt the computer-program.

SOFTWARE

The software structure which is typically determined by interrupt based programming is shown in figure 4. The software can be divided into three parts: main program, subroutines and interrupt routines. The main program which is written in FORTRAN and consists of two loops called 'preparation' and 'proces'. During 'preparation' the operator sets system parameters. This has no consequences for the rectifier because in this phase of the programme the power supply is disconnected.
After a start-command the set parameters and the presence of alarms is tested. When the test results are positive, which means correct parameters and no alarms, the program comes into the 'proces' phase. All control and guard functions are active now. The dialogue between the operator and the system is restricted by the computer. The program cannot be stopped until all stop conditions are fulfilled including that all energy has been removed from the system. The subroutines,

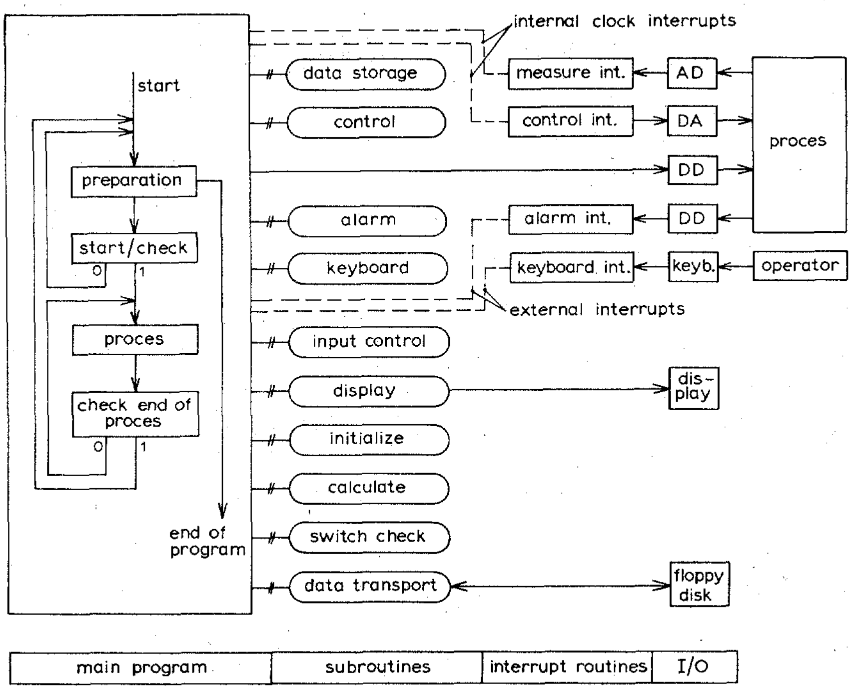

Fig. 4. Sturcture of the software typically determined by interrupt based programming.

written in MACRO-11, actually perform the mentioned computer functions like guardance of proper operation, data logging etc. A subroutine is called by the main program when its flag has been set. It's flag is set by an interrupt routine or by a subroutine. Interrupts are generated by the clock, the parallel interface (DD) in case of an alarm or by the operator. The latter two are the so-called external interrupts. The interrupts are processed immediately by the (MACRO) interrupt routines which results in transport of the concerning data and the setting of a flag.

Keyboard and display

With some restrictions the operator communicates with the program via the keyboard. After a valid keyword has been accepted the operator can watch and direct the proces by means of three screen-pages. The main page shows the important general system and program parameters and a set of allowed commands. The operator can choose the clock frequency and the ratio between the number of clock interrupts used for signal generation and for measuring. On request the actual measuring values are presented. After the wanted load current is determined, 'proces' can be started and stopped. When stopped, the program makes the wanted current zero and the pumping cycle is reversed so that the energy is removed slowly. The administration commands are collected on the second page. Time can be set and requested. Administration blocks on floppy-disc can be created, requested and send to display or printer. These administration blocks contain all by operator established parameters and the state of the alarms etc. In case of an alarm the program creates an administration block automatically. Blocks of already logged data can be displayed or printed. Operator comments upon the behaviour of the system are printed. The third page presents the shape of the control signals, the concerning parameters as well as the commands how they can be changed by the operator. The parameters which determine the shape of the primary current and both switch control signals are shown in fig. 5. The operator can select the length A and C of the time intervals with no current change, and the heigth E of the signal. The slope B of the commutation step and the slope of the pump step D can be prescribed. As mentioned before the length of the commutation step is governed by the load current. Parameters G and H determine the switchcontrol. During 'preparation' these parameters can be varied freely. Within 'proces' thus during the actual loading proces the commands are restricted to those who fit the conditions of entry determined by the program in order to prevent failure modes.

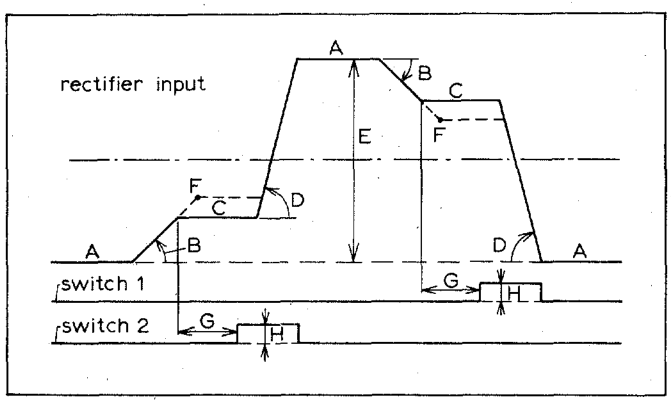

Fig. 5. Selectable parameters of the rectifier input being the primary current of the transformer and both rectifier switch control signals.

Alarmhandling. In case of an alarm the computer gets an interrupt request from the parallel interface. Depending on the kind of alarm the program is forced to decide how to continue the proces. This decision may result in standby operation, in a normal and slow discharge of the load coil via a reversed pumpcycle or in fast dumping of the energy in the dumpresistor. In case of a quench in the secondary circuit the energy is dumped controlled by a parallel reliable hardware protection system. The computer creates an administrationblock on floppy disc with information about type of alarm, system parameters etc. It also makes the desired current zero and triggers an audible and visible alarmsignal.

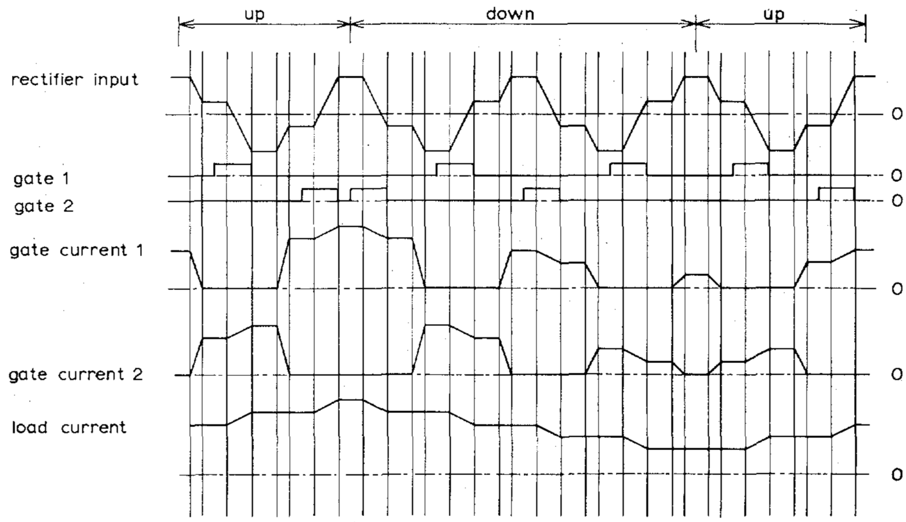

Fig. 6. Relation between rectifier input, both gate currents, and the load current. The reversal of the pumpcycle is shown.

If a quench occurs in the primary circuit, the computer disconnects the power supply of the transformer. The alarm can be reset by the operator. Too low helium level is followed by a reversed pumpcycle i.e. slowly discharge. Figure 6 illustrates how a reversal of the pumpcycle is achieved. When pumping down the commutationstep is retained to the second part of the half period, so exhibiting a complete timereversal of the control signals. Now the phase of both switch control signals is shifted 180^{o} related to the situation of pumping up.

If the switch status is not correct the primary control signal is held constant and the load current is maintained. The proces continues if the failure can be reset, otherwise the energy is dumped.
If a temperature alarm is set because the temperature at the conductor switches or joints are too high, the pumpproces is interrupted until reset.
An alarm set by the power up detector causes standby until reset or dumping. In case of power back up failures the energy is dumped. The communication with the computer by keyboard can be overruled by hand triggering of an alarm or triggering of the protection unit.

The priority of the interrupts are not programmed but they are determined by the electrical distance between the gate and the μ-processor. The delay between an interrupt request and granting the interrupt routine is determined by the maximum length of the routines in which the interrupts are disabled. In the actual program this delay is maximal 350 μs.

FINAL REMARKS

The actual compiled program fills 12 k byte of the 32 k RAM of the processor. The maximum frequency of the generated control signals is determined mainly by the execution-time of the control and measure routines, and by the required number of signal samples per period. At this moment the program is able to scan 9 multiplexed analogue channels each with 1 kHz and is able to generate the control signals with a frequency of 10 Hz with 100 signal samples per period. The latter frequency can be increased further by reducing the measuring frequency. Another possibility is multiprocessing. Dedicated processors handle the separate routines for measuring and signal generation. The latter is inevitable if high frequency (50 Hz) rectifiers have to be controlled. Until now the present software and hardware can easily attend the thermally switched low frequency (0.1 Hz) rectifiers.
The described computer control- and watching-system operates today succesfully in combination with a 9 kA s.c. rectifier [1], also designed and constructed at Twente University of Technology. The computer control system covers an important part of our research program for high current and high efficient s.c. rectifiers. We have recognized that the potentially attractive s.c. rectifier can only be introduced if a fully reliable and safe control system is present. Such a system has to execute a lot of complicated tasks as explained above, but operating such a system still has to be simple and ergonomically justified.

The presented hard ware configuration has been chosen as a software development and testing system. If the software fits all conditions modified by a special rectifier plant, a smaller hardware configuration will be developed with the program stored in (E)PROM. The price of such a definite μ-processor control system can be reduced substantially compared to the 40.000 Dutch florins of the actual system.

The authors wish to thank Prof.Dr. L.C. van der Marel for stimulating this work.

REFERENCES

[1] L.J.M. van de Klundert and H.H.J. ten Kate, "On fully superconducting rectifiers and fluxpumps". A review. Part I: "Realized methods for pumping flux," in Cryogenics, 21, 1981, pp. 195-206. Part II: "Commutation modes, characteristics and switches," in Cryogenics , 21, 1981, pp. 267-277.

[2] H.H.J. ten Kate, P.B. Bunk, H.A. Steffens and L. J.M. van de Klundert, "A thermally switched 9 kA superconducting rectifier," IEEE MAG. 18, 7th Int. Conf. on Magnet Techn., Karlsruhe, BRD, March 1981.