EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN-EP/88-23
February 29th, 1988

On Line Monitoring of the BCDMS Spectrometer Performance

K. Deiters[1,a)], J.P. Fournier[4,b)], V.Y. Karzhavin[2)], M.Y. Kazarinov[2)],
J.C. Michau[4)], J.F. Renardy[4)], A. Staude[3)], J. Strachota[1,c)], and K.M. Teichert[3)]

Abstract

The development and operation of an online test system for a large high energy physics experiment are described. The system is based on computer generated test signals and clearly defined rules of access to the readout and control electronics of the experiment. The rules of access are implemented through standard test software subroutines. They provide for maximum independence of tests and easy test programming. A subset of test programs runs automatically during the data taking periods and guarantees fast problem reporting.

Submitted to Nuclear Instruments and Methods in Physics Research

---

1) CERN, Geneva, Switzerland
2) Joint Institute for Nuclear Research, Dubna, USSR
3) Sektion Physik der Universität, München, Federal Republic of Germany; funded in part by the German Federal Minister for Research and Technology (BMFT) under contract number 024MU12P6
4) DPhPE, CEN Saclay, France
a) On leave from Institut für Hochenergiephysik der AdW der DDR, Berlin-Zeuthen, GDR
b) Now at ARL, Ecublens, Switzerland
c) On leave from Institute of Physics CSAV, Prague, Czechoslovakia

As the high energy physics experiments grow more complex, computers have been used not only for the data acquisition but also for the equipment control and checking. One approach to the equipment performance monitoring is to apply a set of consistency checks to the real experimental data. This approach is not suitable to test equipment components which are only rarely activated. Checks on an individual event basis are limited since there is normally not enough redundancy in the data. A different approach is to generate and process artificially produced detector signals. In this approach one must organize these active tests in a way that they do not disturb the physics data acquisition or each other. Their interference can only be reduced but not removed by using more computers, since any test has to act on the same equipment as used by the physics data acquisition. Moreover, different tests may need access to the same detector components.

In this paper, we discuss a system of active online tests which has been used to monitor the large spectrometer of the Bologna-CERN-Dubna-Munich-Saclay collaboration (BCDMS). This spectrometer is described in ref. [1]. It was designed to study deep inelastic muon scattering with high luminosity. The main physics goal is the measurement of nucleon structure functions. The detectors of the spectrometer include more than 60000 multiwire proportional chamber (MWPC) channels and some 600 trigger system photomultipliers (PM's). A two-level trigger system is distributed along the apparatus to match the 40 m long target. A CAMAC system is used to interface the experiment to a NORD 100 and a NORD 10/50 computer. A REMUS subsystem [2] does the fast readout of CAMAC modules which are read for each trigger. All other CAMAC modules are operated under the standard CAMAC protocol which is slower but more flexible. The computer connection is fully symmetric as shown in fig. 1. By convention, the NORD 100 system is dedicated to physics data acquisition and monitoring; the NORD 10/50 runs the tests described here and some other tasks.

1. Test Hardware

In order to monitor the trigger counter system, all the trigger counters were equipped with Light-Emitting-Diodes (LED) to measure the timing and pulse height of the PM output and the efficiency of the trigger logic. The MWPC tests are measurements of noise and efficiency and also checks of correct data encoding and readout. For the efficiency measurement, a test pulse can be injected simultaneously to all preamplifiers of a chamber. All tests are done during the 12 s time interval between two beam pulses from the accelerator.

A Trigger Processing Unit (TPU) plays the central role in the test hardware. It is a set of CAMAC modules interconnected by an ECL bus. The TPU defines two time slots of programmable length, which we call, respectively, "beam extraction" and "test". In normal operation, the slots are started by timing signals from the accelerator. They also can be generated internally, when the muon beam is not used. For both slots, the TPU generates a gate signal extending over the full length of the slot, as well as a "start" and an "end" signal. The "end test" switches the REMUS routing unit (fig. 1) to the data acquisition computer. Then the "start beam extraction" is sent to this computer, which responds with a system clear and reset sequence. The physics data acquisition takes place during the "beam extraction" slot. The "start test" switches the REMUS routing unit to the test computer and is also sent to the computer itself to enable the test event acquisition. The TPU has three priority ordered trigger inputs which can be individually enabled by the computer. There are three first level physics triggers in the experiment, which are connected to these inputs. In addition, for each time slot, the TPU can be programmed to generate (with the lowest priority) test events of one type at a time. The test events are processed the same way as the physics events. Any event may occur only when the TPU is ready and then will block the TPU for further events until an "end readout" or second level trigger rejection signal is received. When an event is not rejected, accept signals are sent to the readout system, their pattern depending on the event type, and an interrupt is sent to the computer. An event type tag is latched to the REMUS master crate register to be read as the first data word. A pattern of inhibit levels is applied to the REMUS branch drivers in order to suppress the readout of data irrelevant for this event type. At the end of the direct memory access readout of the event data, the TPU sends reset signals to make the experiment ready for the next event.

The TPU has one test pulse output, which is connected to a dedicated high speed ($\beta = 0.99$) transmission line installed along the spectrometer. Test pulses are generated by the TPU automatically on request of a test event. They propagate downstream the spectrometer with the same speed as scattered muons in the spectrometer. They are picked up at regular distances and distributed via programmable fan-out units to the LED drivers, MWPC strobe fan-outs and MWPC test inputs.

2. Test Software

The test software provides for automatic hardware tests and also, in interactive mode, for efficient in-depth debugging. The tests are implemented as independent Real-Time (RT) programs written in Fortran 77. All RT features needed to execute a test event are concentrated into one standard subroutine TESTEV. Similarly transparent software tools are available for the write access to the test pulse fan-outs. A common loading procedure for the test programs provides their correct incorporation into the RT environment. Because of these facilities, all RT features are transparent to the programmer, who can concentrate on his equipment test as if he had exclusive access to the readout and control system.

2.1 Standard Routine TESTEV to Get a Specific Test Event

Any test program generates test events and reads their data uniquely by calling TESTEV. A semaphore of the NORD Sintran III operating system is used as the Event Access Semaphore (EAS) to organize the queue of test event requests. Only one RT program at a time can reserve this semaphore. The reservation queue respects the program priority. When the access is granted, the priority of the calling program is increased above all other tests. TESTEV generates the requested event, and if the event data become available in the expected time, they are read out. If the data are not available in time, which, normally, does not happen, a recovery procedure is executed. In any case, the hardware is restored to the correct initial state in a well defined time, after which the EAS is released and the priority is decreased to its normal level. Some tests need special sequences of CAMAC functions to prepare the test event and thus take more real-time to get it. According to this time, test events are classified in two groups: fast events (up to 20 ms) and slow events (up to 500 ms). In case of a slow event, TESTEV checks whether there is enough time before using the EAS-granted access. This check is done by trying to reserve without waiting a Slow Event Semaphore (SES), which becomes unavailable 1000 ms before the end of the "test" time slot. If the reservation is not successful, the EAS and priority are given up and a new attempt to reserve the EAS is made only after one second.

The two access semaphores are controlled by the test synchronization program SYNCT. This program is started at the same time as the operating system, and has a higher priority then any test program. It starts with reserving both the EAS and the SES for itself. On each interrupt by a TPU signal "start test", SYNCT releases the EAS and the SES and reserves them again 40 ms respectively 1000 ms before the end of the "test" slot. This method guarantees, with safe margins, that the hardware is available in time for the physics data acquisition.

2.2 Test Programs

We have three LED test programs, which control the LED outputs of the Test Pulse Fan-outs (TPF) and analyse data from ADC's, TDC's and pattern units. Using different patterns for test pulse distribution, we check efficiency and timing of all trigger counters and the first level trigger logic. The MWPC test programs control the MWPC outputs of the TPF and test noise, efficiency, encoding (bit pattern sent to the LRS PCOS II [3] front-end shift register), and the second level trigger logic (based on the fast OR of each chamber).

Most test programs need many events with the same TPF setting. Since the remote control of the TPF is slow, it is efficient to do the switching only once for all these events. To avoid interference between different tests, a CAMAC semaphore is assigned to the TPF outputs for LED and to the TPF outputs for MWPC. A test program has to reserve the corresponding semaphore before it can modify the TPF. This use of the CAMAC semaphores is analogous to the use of the EAS, whereas the programs using them do not have to execute on the same computer.

The test programs were first implemented as interactive tools for in-depth debugging and full status reporting of the hardware subsystems. The programs use standard CERN packages for commands, histogramming and graphics. Later, the programs were used for routine tests: All the programs have a common command to run a full status test with standard values of parameters. Also common to all the programs are some features for experts: There is a standard error processing package for all errors detectable in a single event. A short message is displayed on the first occurrence of an error type, then only the error statistics is updated. The user can flag any subset of error types for a dump of the relevant part of the event with a pointer to the first faulty data word. Furthermore, a common command OSCILO initiates an infinite loop of test events without any data processing, in order to obtain a high event frequency for inspection with an oscilloscope.

For the routine tests during data taking periods we have defined an Automatic Test Set (ATS) of 8 programs. This set consists of the ADC-, TDC-, Pattern-Unit- and MWPC-Pattern-program executing regularly according to a fixed hour-of-day schedule (1 or 2 hour intervals) and four fast programs which execute more often. These "watch dog" programs are a one-event, four-bit-pattern MWPC test which executes every minute to detect low-voltage power supply failures, a 100-event MWPC noise test to detect high-voltage power supply failures (every 15′), a trigger setting check (every 30′) and a magnetic field measurement (every 30′). Short reports from the ATS programs are displayed on the central status TV monitor. More information is available as printed output or data on disk. Any of the ATS programs can be executed immediately without interrupting its schedule, or taken out of the ATS to be used in interactive mode.

3. Conclusions

Two components have been specifically designed for the online test system described here: a hardware processor, which automatically generates all control signals required for a specific event type, and a program which centralizes the management of all requests for access to the system from different users. These tools provided the necessary flexibility to develop and to add entirely independent test programs during the construction phase of the experiment, allowing many different users simultaneous access to the system. By just adding a scheduler for the data taking phase, it was straightforward to assemble a routinely executed test cycle, which proved very helpful in running the apparatus over extended periods of time with good reliability.

Acknowledgement

We are indebted to M. Cribier, J. Feltesse, R. Kopp, and G. Smadja for their help and useful discussions of problems encountered during the system development.

We appreciate greatly the technical support of H. Herbert, P. Klemm, P. Marelli, G. Miller and H. Willaredt.

We wish to thank all our colleagues from the BCDMS collaboration for important remarks and suggestions which helped to improve the system performance. We acknowledge the stimulating discussions with R. Voss when writing this paper.

References

1. A.C. Benvenuti et al., NIM 226(1984)330
2. P. Ponting, CERN/EP Electronics Note 80-01(1980)
3. PCOS II, Proportional chamber operating system, LeCroy Research Syst. Corp., Spring Valley, New York

Figure Caption

Fig. 1: Schematic layout of the BCDMS data acquisition system

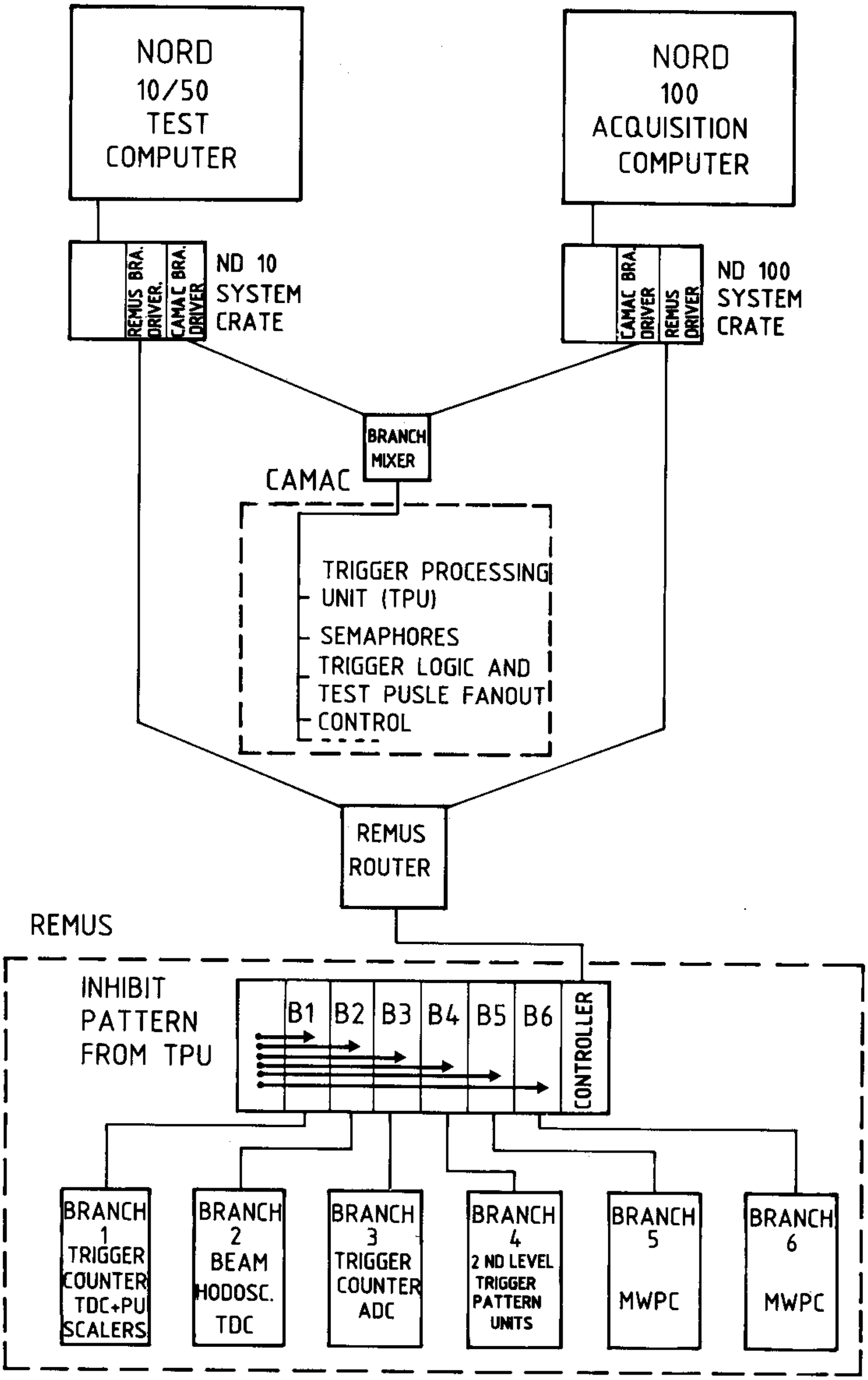
NORD
10/50
TEST
COMPUTER
NORD
100
ACQUISITION
COMPUTER
REMUS BRA. DRIVER.
CAMAC BRA. DRIVER
ND 10
SYSTEM
CRATE
CAMAC BRA. DRIVER
REMUS DRIVER
ND 100
SYSTEM
CRATE
BRANCH
MIXER
CAMAC
TRIGGER PROCESSING
UNIT (TPU)
SEMAPHORES
TRIGGER LOGIC AND
TEST PUSLE FANOUT
CONTROL
REMUS
ROUTER
REMUS
INHIBIT
PATTERN
FROM TPU
B1
B2
B3
B4
B5
B6
CONTROLLER
BRANCH
1
TRIGGER
COUNTER
TDC+PU
SCALERS
BRANCH
2
BEAM
HODOSC.
TDC
BRANCH
3
TRIGGER
COUNTER
ADC
BRANCH
4
2 ND LEVEL
TRIGGER
PATTERN
UNITS
BRANCH
5
MWPC
BRANCH
6
MWPC

FIG. 1