EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN-EP/86-162

13 October 1986

**THE ELECTRONICS AND PARALLEL READOUT SYSTEM
FOR THE UA1 LIMITED-STREAMER TUBE MUON DETECTOR**

A. Bettini¹⁾, G. Bovo¹⁾, G. Busetto¹⁾, S. Centro¹⁾, S. Cittolin²⁾, M. De Giorgi^{2,a)},
T. Markiewicz³⁾, A. Meneguzzo¹⁾, D. Pascoli¹⁾, A. Placci²⁾, P. Rossi^{2,a)}, C. Rubbia²⁾,
K. Sumorok²⁾, L. Zanello⁴⁾ and P.L. Zotto¹⁾

ABSTRACT

A muon position detector based on limited streamer tubes has been built for the UA1 experiment at the CERN $p\bar{p}$ collider. The tubes are arranged in chambers equipped with electronics to allow for analog readout of the induced signals on strips. The electronics module for charge amplification and conversion (STAR), the data-acquisition system based on the VME standard, and the performance of the detector will be described.

(Submitted to Nuclear Instruments and Methods
in Physics Research)

a) Present address: Università di Padova, Italy.

1) Dipartimento di Fisica dell'Università di Padova and INFN Sezione di Padova, Italy.

2) CERN, Geneva, Switzerland.

3) Department of Physics, University of Wisconsin, Madison, USA.

4) Dipartimento di Fisica dell'Università di Roma and INFN Sezione di Roma, Italy.

1. INTRODUCTION

The muon position detector, based on limited-streamer tubes, is part of the UA1 upgrading program. The basic characteristics of the electronics and readout system have been defined to satisfy a number of demands, i.e.

- a large number of channels (more than 50,000) with a good spatial resolution (better than 0.5 mm) on a large surface (more than 850 m²);
- readout time in the range of 10 ms to match the rest of the UA1 detectors;
- large data reduction in real time;
- autonomous test, monitoring and calibration so as to allow easy integration of the new detector in the existing, complex readout system;
- reliability of operation.

As far as the front-end electronics is concerned, these constraints have led to a design choice of current integration from the strips with a time-invariant double integrator mounted directly onto the detector. In order to keep the cost per channel low and to minimize the number of converters and digital connections, analog-to-digital conversion is multiplexed.

For the data acquisition, the short time available has imposed the solution of parallel readout. The use of cheap and fast microprocessor-based central processor units (CPUs), in the framework of the VME bus standard, allows local data compression during data acquisition, and relatively sophisticated monitoring, calibration, and hardware testing, independently of the rest of the data acquisition system.

Figure 1 shows a schematic view of the system organization. The streamer tubes analog readout (STAR) Controller (SC) interfaces the local bus (STAR bus) connecting up to 120 STAR modules with the VMX bus going to the acquisition CPU. In practice, we have up to 90 modules on each STAR bus. The same CPU is connected to a maximum of four SC (six boards is the limit imposed by the VMX standard).

These CPUs are VME modules, based on the Motorola 68010, residing, together with other types of boards, in a VME crate whose contents are shown in detail in fig. 2 and described in section 2.5. One of the CPUs acts as supervisor, collecting all the data after the parallel readout and sending them to the main Data-Acquisition System through a VMX connection to a memory board situated in another VME crate. The box labelled SS (Simulation System) contains hardware capable of simulating the STAR modules, used for software development and test, and a counter, whose value is distributed to all the SC, to mark each portion of the same event with the same label and check for trigger synchronization problems.

2. THE HARDWARE

2.1 Detector structure

The operation of limited streamer tubes has been discussed in several papers [1, 2], as has been the possibility of detecting induced pulses on electrodes through resistive cathodes [3-5]. It is, however, the first time that an analog readout system has been operated on a very large scale.

Extensive tests have been performed before the construction, both with an X-ray source and with an electron beam. The complete detector, which consists of 108 chambers, has been described elsewhere [6], so we recall here only the structure of the chambers.

One chamber has a surface area of 4.8×1.65 m² (see fig. 1) and is constructed from 144 long tubes and 432 orthogonal short tubes, both having a square cross-section of 1 cm². These tubes are grouped in modules of 16 elements each. The long tubes are read through 352 short strips, the short tubes are read through 112 long strips. The strips are 10.2 mm wide and have a 12.7 mm pitch.

The readout electronics is mounted along two sides of the chamber. Each STAR module serves 32 channels, so eleven modules are needed for the long side and four modules for the short side

(where eight channels are not used both in the first and in the last module). The strip planes have a standard female connector at one side to allow for direct insertion of the STARs. The STARs were made only 9.5 cm wide to minimize the dead space that amounts to 7.5% of the area of the detector. All the STARs of one chamber are bus connected to a 34-wire flat cable (STAR bus). Many chambers are connected to the same bus.

Each STAR performs the following functions:

- i) current integration and shaping,
- ii) sampling and holding of the peak value,
- iii) analog-to-digital conversion of the held charges,
- iv) separate gain correction for each channel,
- v) storage of the digital data in a double buffer,
- vi) data selection and filtering through Analog Threshold, Mask, and Data-Ready Flag.
- vii) interfacing to the bus.

On each board there is also a charge generator for calibration and equalization, through gain correction, of the different channels.

2.2 Analog front end

The analog front end collects the induced charge on the strip, and shapes and holds it. The maximum charge per channel is about 15pC, while the strip capacitance is 380 pF/m; which means 1.8 nF for the longest strips. The scheme of the front end amplifier is given in fig. 3. A simple double integrator with a common base transistor as input stage has been chosen in order to have a correct shaping and a reasonable noise figure.

The signal at the output of the second integration is

$$V(t) = (qR_1R_3/R_2)[1/(\tau_2 - \tau_1)][e^{-t/\tau_2} - e^{-t/\tau_1}] ,$$

where q is the input charge, $\tau_1 = 726$ ns is the time constant of the first integrator and $\tau_2 = 7.26$ μ s is the time constant of the second integrator. The output voltage has its maximum at

$$t_M = n/(n - 1) \tau_1 \log(n) = 1.86 \mu\text{s} ,$$

where $n = 10$ is the ratio of τ_2 over τ_1 . Zero-pole cancellation has been avoided because its inherent critical adjustment is not compatible with the number of channels involved.

The peak voltage as a function of the input charge at t_M is

$$V_M = 64.5 \text{ mV/pC} .$$

The sampling error due to possible jitter dt of the hold command is

$$\Delta V/V_M = 0.25 (dt/t_M)^2 .$$

Taking into account the amplification factor (4.5) in front of the 8-bit analog-to-digital converter (ADC) (5 V full scale), we get a resolution of 67 fC per count. The noise indexes of the filter are

$$\int_{-\infty}^{\infty} [h'(t)]^2 dt = 1.22/t_M$$

for the serial noise, and

$$\int_{-\infty}^{\infty} [h(t)]^2 = 1.86 t_M$$

for the parallel noise, $h(t)$ being the transfer function.

The computed equivalent noise charge (ENC) of the circuit is 4.1 fC. The contribution of the front end transistor is 2.2 fC, while the double integrator contribution is 3.56 fC. The measured ENC for the integrators is 2.2 fC, while we measure 14.8 fC for the transistor; this large noise is mainly due to the simple biasing network. The total measured ENC is then 15 fC, which is however 1/4 of the ADC least count.

A quad double integrator followed by a sample-hold circuit has been realized in hybrid form. From each hybrid the sum of 4 channels is available (Σ) and tying all these outputs together to an amplifier's virtual ground an analog signal, the sum of 32 input charges, is obtained. A clear signal (CLR) is also provided to discharge the second integrator before the next bunch crossing if we do not get the command to perform analog-to-digital conversion. The emitter biasing resistor (0.2%) of the common base transistor is also used to inject a known amount of charge during test and calibration.

2.3 Signal multiplexing and analog-to-digital conversion

The hold command (STOR in fig. 3) is synchronous with the $p\bar{p}$ collider bunch crossing and it is broadcast to all STAR modules via the controllers. If the first-level trigger processor does not recognize a good event, a CLEAR command is sent to all STAR modules before the next bunch crossing (CLR in fig. 3). When a good event is recognized a convert command is sent to all STAR modules, again through the controllers. Upon reception of such a command the 32 held values are scanned through a multiplexer and converted through an 8-bit successive approximation ADC.

The digital values are stored in one of two digital buffers of the Data Memory on the STAR board. During this operation the 32 analog values are compared with a threshold (programmable and common to all STARS connected to the same controller) and if one of them is greater than the threshold a Data-Ready Flag is set.

At the end of the conversion ($\approx 110 \mu s$) each controller addresses its STARS (up to 120) in sequence and reads the whole data buffer only if a Data-Ready Flag is set. During the conversion process a control byte for each channel is read from the control memory; seven bits are used to adjust the ADC range in order to equalize the channels while one bit is used as a mask to inhibit setting of a Data-Ready Flag for noisy or wrong channels.

Gain parameters are written into the Control Memory after the calibration procedure that is performed through the programmable charge generator on each STAR module. Both Data and Control Memory are RAM accessed in auto-increment mode and can be fully exercised via the controller.

In fig. 4 the block diagram of the STAR is given while the details of the gain controlled ADC is shown in fig. 5. The reference current of the ADC internal multiplying digital-to-analog converter (DAC) is varied by $\pm 10\%$ through DAC08 as a function of the gain parameter coming from the Control Memory. To make the internal-multiplying DAC fast enough, the standard compensation has been changed (R_c, C_c).

In applications requiring a non-linear conversion scale the resistance R can be used to make the ADC response of the type

$$n_{ADC} = (255/5[V])(V_x/1 + \alpha V_x),$$

where

$$\alpha = 1/RI_{\text{ref}}$$

assuming $I_{\text{ref}} = 1 \text{ mA}$. In our application $\alpha = 0$. The ADC response curves and the equivalent resolutions and ranges are given in fig. 6.

2.4 The Star Controller

The STAR Controller interfaces the readout CPU to the STAR bus. The SC is a VME module connected to the CPU through the VMX bus. The STAR bus interface is one of the major parts in the SC. It merges the commands generated by software-controlled operations with the ones hardware generated by the NIM signals coming from the trigger processor. Moreover, all the signals originated by the trigger processor can also be software generated, allowing testing and calibrating the STAR modules with known signals. The block diagram is given in fig. 7. Every SC controller has a 4 Kbyte p RAM to store pedestals and to perform pedestal subtraction. A set of registers is reserved for control operations and status definition.

The SC also generates two analog signals through a couple of DACs:

- i) Analog threshold (8 bits) (see section 2)
- ii) Analog test (8 bits with 10-bit linearity) to drive on each STAR the calibrator that generates the test charge

$$q_t = [0.09 \text{ ATEST} + q_0] \text{pC} ,$$

where ATEST = 0—255 is the digital value driving the DAC and q_0 is the spurious charge due to the digital noise in the charge generator circuit.

Apart from those initiated by the trigger processor, all the operations performed by the SC correspond to either a read or a write CPU instruction to an address in the SC window. This characteristic makes the SC very flexible. On the other hand, the lack of capability of autonomous operations demands some care in writing the control software in order not to waste time, mainly during data acquisition (see later).

An important feature of the STAR system is the double buffer capability. The SC takes care of the dual buffer handling making it software transparent. In dual buffer operation, in order to derandomize events, a new conversion is allowed as soon as one of the two data buffers in the STAR modules is available while still reading out from the other buffer. The Hardware Sequencer (HS) takes care of the access to the appropriate buffer both during analog-to-digital conversion (writing) or during SC access (reading).

Since the Data Memory is a single port RAM the HS also takes care of temporarily suspending the readout process while analog-to-digital conversion takes place. The HS is stepped by software instructions (EOC = end of conversion, EOR = end of reading) and goes through the states, as shown in fig. 8, handling the double buffer addressing. In status C and E both buffers are full, but in C Buffer 1 has to be read first, while in E Buffer 2 has to be read first.

2.5 VME crate modules

As building blocks of our readout system, we utilize several different types of VME boards, some of which have been built under UA1 specifications. In this section, a short description of each type of module is provided.

- a) CPUA1 (Data Sud Systemes): MC68010 CPU, 8 MHz clock, VME bus master (arbiter), VMX bus primary-secondary master, NS 16081 floating-point processor, RS232 serial port, front panel

signal I/O, 256 Kbyte Dynamic Memory, 8 Kbyte Static Dual Port Ram, Timer, VME interrupter.

- b) Dual Port Memory (DPRX, Data Sud Systemes): 128/256 Kbyte static memory, 400 ns access time, VME/VMX access, write broadcast mode via address modifier select.
- c) Crate Interconnect Module (CI): VME multicrate connection, 10 Mbyte per second DMA transfer, 64 Kbyte window memory mapping.
- d) Battery Backup Memory (BB) : SYS68K/RR-1 (Force), 128 Kbyte.
- e) Graphic Interface (Data Sud Systemes) : DSSE512CHROMA8, 512 × 512, 8 colours Graphic VME board, MC68121 IPC with dual port RAM, EF9365 Graphic Display Processor, 192 Kbyte of image memory.

2.6 Communication between CPU and SC

As is well known, the VME bus architecture is based on memory-mapped communication between modules residing in the same VME crate: in other words, each module, be it a CPU or a memory board, can be configured so as to respond to a preset range of addresses in the 4 Gbyte VME address space. A given CPU will therefore see the Dual Port Rams of the other CPUs or the memory boards present in the crate as parts of its own memory. Software instructions for normal memory-to-memory transfers can be used also for CPU-CPU communication.

The same is true for the VMX bus, the local bus that allows data transfers between modules residing in the same or in different crates, in parallel with VME bus activity. Only the use of different address windows will discriminate between VME and VMX bus transactions.

The SC occupies, in the VME addressing space, a bank of 64 Kbytes whose base address can be selected by jumpers. The address of the control and data registers within this bank is shown in fig. 9. The interaction between a data acquisition CPU and the SC is realized by means of read/write operations to the appropriate locations of the SC address space. The absolute address of the first location of the SC window and the position of the registers relative to this base address are kept in a table which is part of the system configuration description. This scheme allows a simple and flexible handling of the SC-CPU communication and provides software independence from hardware modifications of the SC structure.

3. THE SOFTWARE

3.1 Software general structure

From the point of view of the general organization of the software, our system can be considered as composed of:

- a) one 'supervisor' CPU which we will call SUP,
- b) a variable number (0 to 14) of 'data acquisition' CPUs, called ACQs,
- c) one 'monitor' CPU, indicated by MON.

Note that the presence of the MON CPU is not mandatory for the data acquisition process.

Let us now briefly summarize the basic tasks of the different types of CPUs listed above:

SUP: handles communication with the central Data Acquisition System, broadcasts commands to the ACQs, collects their data in an orderly fashion; sends it to the event builder unit; writes data into a FIFO memory for monitoring purposes.

ACQ: reads data from the STAR modules through a set of 1 to 4 SCs; stores it in a dual port buffer (DPR) also accessible through the VME bus; when data acquisition is not active, interacts with the STARs connected to it for calibration, pedestal measurement, hardware tests, etc.

MON: during data acquisition, performs event sampling from the FIFO memory, event display, histogramming, etc., under user control; when data acquisition is not running, handles STARs calibrations and tests.

3.2 CPU communication.

The basic point of the software structure in a tightly coupled multiprocessor system is a clear and safe definition of a communication mechanism between the different CPUs.

In our system the unit of communication is a *command*: it defines an action to be taken and it may request specification parameters; it is sent from a 'master' to a slave with a procedure described in detail in the next section. In our application, command handling relies on two basic principles:

- i) Commands must be executed.
- ii) A master CPU must know what it is doing.

In other words, all the checks that a command is appropriate in the current system situations are the command sender's responsibility. The slave CPU that receives the command does nothing but consult a dispatch table and transfer control to the subprogram that handles the command. So, for instance, the SUP CPU will not, in general, order an ACQ CPU to read an event if the run has not been initialized. If, however, it were to do so for some reason (testing or an other reason), the ACQ CPU would simply go off and read one event, storing the data in the normal way.

3.3 Command handling

For each CPU, a static Dual Port RAM (DPR) of 8 Kbyte is available. This memory is accessible both internally from the CPU and externally from the VME bus. Two 16-bit words, occupying the first four bytes of the DPR, are used for CPU communication and command handling: LSTAT (status word) and LCOMD (command word).

LCOMD = 0 indicates that a CPU is idle and waiting to accept commands. When a master wants to send a command to a slave, it writes the corresponding command number in the slave's LCOMD. When the slave has executed it, it clears LCOMD.

LSTAT indicates the CPU status apart from the fact that it is idle or executing a command: an idle CPU can be waiting for run start, for event readout, for buffer emptying etc.

The master needs to check the CPU status in order to know whether the command it wants to send is legal. Access conflicts due to almost simultaneous testing of LCOMD from two masters have been prevented through the use of a semaphore in the following way. While a master is looking at the command word of a slave and, if zero, is setting it to the command number, the semaphore is *set*, indicating that no other master can do the same thing.

The semaphore's status itself is tested and set with a single instruction of the M68K, TAS, to avoid reproducing the same access conflict at the semaphore's level. The TAS instruction cannot be used directly for the command word because it would change its value so that the previous contents, if different from zero, would be lost. In order to avoid deadlocks, setting and clearing the semaphore and handling of the command word is done inside a utility subprogram. Calling this subroutine is the only legal way to send commands.

The communication scheme we have described is completely general in the sense that it allows any CPU to control the actions of any other CPU. In our application, however, only the supervisor acts as a command sender, since the system organization does not require, for the time being, any direct communication between ACQ CPUs.

3.4 Software organization.

The code contained in the SUP and the ACQ CPUs is identical and is entirely written in M68000 assembly language. The code used by the MON CPU is different and written in RT FORTRAN.

Each CPU has a unique hardware identification number (ID) that determines its address window in the VME bus address space; the CPU can find its own ID by reading the contents of location 1010, hexadecimal. A table, which is part of the system configuration description, contains the identification number for all CPUs and is globally accessible, being stored in the BB memory.

Each CPU looks for its own ID in the table, assuming then as a logical CPU number the corresponding address (0, 1, ..., 15). If this address is 0 (i.e. if the CPU finds its ID as the first entry in the table) it assumes the role of SUP. If it is equal to 15 (last entry in the table), it assumes the role of MON; for any other address, the CPU is an ACQ. The same table is then used by the SUP to determine the present system configuration (number of ACQ CPUs, presence or absence of MON, etc.).

The SUP/ACQ code can easily be divided into two main parts: the first, executed during data acquisition, deals with the readout of event data from the STAR modules; the second, used when data acquisition is *not* running, allows the user to perform interactively the STAR modules' calibration, pedestal measurement, hardware tests, etc. By convention, actions related to the first part correspond to *negative* commands, while those for the second part have *positive* commands.

The general structure of the software can therefore be summarized in the scheme shown in fig. 10.

3.5 System startup

The executable code is normally stored in BB3 and system configuration tables in BB1 and BB2. Pushing the reset button of the SUP CPU causes a reset interrupt on all the CPUs; this is handled directly by the resident Monitor, which transfers control to an interrupt service routine (ISR). This routine reads a fresh copy of the executable code from BB3 into the CPU and starts its execution. Each CPU performs its private initialization procedure, filling tables from BB1 and BB2 contents. Moreover, the SUP CPU does the general system initialization. Every CPU then goes into a waiting loop until a command arrives.

3.6 Data acquisition

Readout from the STAR modules is organized as follows. A run can be activated either by hardware (setting a level into one of the front-panel inputs of the SUP CPU) or by software. After the run has been started, each CPU is ready to accept an event.

When a trigger arrives, it is distributed to all the SC (TRGIN signal) and each controller broadcasts it to each STAR module connected to it on the STAR bus ($\overline{\text{STOR}}$ signal). Each STAR module will hold the charge, perform analog-to-digital conversion, and store the values of 32 channels on a digital on-board memory.

For a given SC, two hardware flags are set: the first, 'Conversion in Progress' (CIP), is hardware reset as soon as the conversion is over; the second stays on, and inhibits further conversions. Once an event has been read out, this second flag is cleared by software enabling the trigger for the next event. The status of these two flags can be monitored by software, testing the least significant bit (LSB) and the most significant bit (MSB), respectively, of the control and status register of the SC.

Once the CIP flag has gone to 0, each STAR in turn is tested. If the 'Data Ready Flag' is set, this indicates that at least one channel had a pulse height larger than the current threshold. All the 32 channels of the module are then read, with automatic pedestal subtraction, if enabled.

The procedure we have described corresponds to 'single buffer operating mode'. Since each STAR module is equipped with a dual digital memory, a 'double buffer mode' is also possible: in this case, the hardware flag is cleared as soon as the presence of the event is acknowledged and before the actual readout; acceptance of a second trigger is therefore allowed, and the Hardware Sequencer will automatically control storage of data in the currently free buffer.

At the end of data read, each CPU sets its status to 'readout terminated, data in memory'. When the main readout requests an event, the SUP collects the data and writes it into the indicated location of a VME memory.

3.7 The dummy stars and the trigger counters

A set of 8 special modules, occupying the highest addresses on the STAR bus (120₁₀–127₁₀) are included in the system for testing purposes. To the software, they look exactly like any other module; for those with addresses from 120₁₀ to 127₁₀ the data content is fixed and preprogrammed so as to give a regular and recognizable pattern useful to trace readout problems. The last one (address 127₁₀) is called the *trigger counter*. It is a 4-bit counter which must be reset by software and which is incremented by the trigger signal. The purpose of the trigger counter is to monitor the internal synchronization of the different parts of the detector: the portions of the event read by each SC must all bear the same trigger counter value. Moreover, this value must match with the trigger number known to the Central Data Acquisition. The trigger counter value is always read and stored in the event buffer, even for a SC with no data.

3.8 Purpose of the monitor

The Monitor Program (MP) has been designed so as to accomplish two basic functions:

- i) during data acquisition, perform event sampling, analysis and display;
- ii) outside data acquisition, control hardware tests, calibration, pedestal measurement, etc.

Event sampling is completely asynchronous with respect to data taking: the SUP CPU writes events into a first in first out (FIFO) memory, the MON CPU reads these events whenever it can. There is no provision for a minimum fixed sampling rate, and therefore no way in which the monitoring activity can slow down data acquisition or interfere with it.

3.9 Architecture of the monitor

The MP occupies approximately 80 Kbyte of memory in the MON CPU and is written in RT FORTRAN [7]. The VMX bus is used to access a dual port memory module (128 Kbyte EPROM), containing the Fortran Run Time Library and Graphic Library, and a dual port memory module (FIFO) used as an event buffer. RT FORTRAN is an extended version of FORTRAN 77, specifically designed for real-time applications. While retaining all the advantages of the high-level language, it embeds typical data-acquisition program features such as: access to absolute memory addresses, direct control of registers' contents, easiness of interrupts, connection and servicing, optimized bit/byte operations and block moves, etc. Access to absolute addresses can be achieved either by the use of 'pointer variables' which can be EQUIVALENCED to the wanted address or by mapping labelled COMMON blocks onto fixed- address areas.

During acquisition, the MP communicates with the rest of the system only through the FIFO memory and the BB. There is no direct dialogue with the ACQ CPUs. The BB module is used to get current values for run-control parameters, such as threshold, flags, trigger type, etc. When in calibration mode, on the contrary, the MP decodes the user's command, which is expressed in terms of the physical STAR modules, determines which CPU corresponds to the request, and communicates the command to it through its DPR, accessible from the VME bus. The user can therefore ignore the actual sharing of STAR modules among the CPUs and SCs.

The 128 Kbytes of the FIFO memory are divided into 9 event blocks of 10 Kbytes each and 16 message blocks of 256 bytes each. These message blocks, one for each ACQ, will contain error messages generated during data acquisition. The MP outputs these messages on the display together with the relative CPU identification. Notice that this technique of message buffering is very valuable for system debugging, especially when the problem refers not to the single CPU code in itself, but to multi-processor features of the system such as synchronization between CPUs, timing, deadlocks and similar problems. In these cases, the standard debugging techniques are of no great help as they tend to modify the relative timing of the CPU–CPU interactions.

The event display shows in a bidimensional plot the strips which have been hit in a given event, for a selected part of the apparatus. This plot is divided into blocks of 15 STARS each, corresponding to one chamber. A blow-up of one block shows the pulse height for each of its $32 \times 15 = 480$ channels. The integrated number of STARS which have had a signal and the number of hits per channel are collected into a hit map. The pulse height distribution is recorded into several histograms, corresponding to the different physical parts of the apparatus.

4. PERFORMANCE

4.1 Reliability and stability

Reliability of operation is very important for the system so described as once the chambers have been installed there is virtually no access to replace malfunctioning STAR modules during the course of a run. Typical run periods are two to four months of continuous operation. A problem can be particularly acute because of the bus structure of the readout, i.e. one STAR module can inhibit 90 modules. The bus interconnects between chambers have all been made accessible such that any chamber can be bypassed, if necessary, thus limiting the operational loss to 15 STAR modules in the case of malfunction. In practice, during the four month run at the end of 1985 this occurred once, leading to the loss of one chamber out of 108. The other chambers operated reliably throughout this period.

During a run period, calibration of the STAR modules is performed approximately once every two days. For each channel of readout a pedestal and a gain correction factor are evaluated. Over a period of two weeks the constants evaluated were compared in detail. In general, it was found that the pedestals were extremely stable: 97% had an r.m.s. variation from the average of less than 0.5 counts and only 1.5% of the channels exhibited changes of greater than 3 counts from the average value. The channel gains are more sensitive, though 80% showed variations of less than 1% and only 5% fluctuated by more than 5% in gain. These results are consistent with the statistical procedure used in evaluating the constants and are characteristic of a very stable system.

4.2 Data quality

All CPUs perform the readout in parallel, checking first the Data-Ready flag of each STAR ($\approx 20 \mu\text{s}$ per STAR). When the flag is set, the 32 channels are read and buffered, performing zero skipping, in about $300 \mu\text{s}$. Since the maximum number of STARS connected to a CPU is 420, and 1 to 2% of the modules set the Data-Ready Flag during normal data taking, the mean time to read out and bufferize an event is:

$$420 \times 20 \mu\text{s} + 8 \times 300 \mu\text{s} = 11 \text{ ms}.$$

This corresponds to about 700 bytes per event.

For a muon track traversing a chamber, typically two or three strips per coordinate are fired. These are called clusters and from the relative pulse heights on the strips within a cluster the position of the track is inferred. Figure 11 shows the maximum pulse height in a cluster obtained from a sample of cosmic-ray muons. From the dynamic range of the pulse heights observed it can be seen that the gain of the double integrator is well matched to the sensitivity of the ADC.

The intrinsic spatial resolution that can be achieved by the system has been evaluated in a test beam. Figure 12 shows a plot of residuals obtained for beam particles at normal incidence to the tubes and the strips. The algorithm used to extract the coordinate from the strip pulse height produces a systematic non-linearity across a strip which has been corrected for. The resolution of the beam spectrometer in defining the position of the beam was $150 \mu\text{m}$ and is not corrected for. The full width at half maximum of the residual distribution is seen to be $900 \mu\text{m}$. The r.m.s. is $580 \mu\text{m}$.

including the tails of the distribution. However, cutting the tails of the distribution at 3σ and re-evaluating the r.m.s. we obtain $420\ \mu\text{m}$. We conclude therefore that the design goal of a spatial resolution better than $0.5\ \text{mm}$ is met.

In order to estimate the global efficiency of the system to detect muons we have used a sample of cosmic-rays triggers. This data sample was collected without any requirement from the streamer-tube system. Muons with momenta, as measured by the central detector, bigger than $8\ \text{GeV}/c$ were selected; their track can easily be extrapolated to the detector planes. By asking how many clusters have been observed compared to those expected from the extrapolation we obtain an efficiency of $93 \pm 2\%$, consistent with what we expected from geometrical dead spaces.

Acknowledgments

We wish to thank R. Conte, S. Lazic and P.A. Marchesini for their collaboration in testing and building prototypes, J. Dorenbosch and D.J. Holthuisen for the help given for the first connection of the apparatus to the existing data-acquisition system. M. Demoulin, W. Haynes, W. Jank, E. Pietarinen and H. Von Der Schmitt made many important suggestions for the use of the VME acquisition system.

All the electronics modules were built and tested by Laben SPA, Milan. We wish to thank P. Casoli, from Laben, for his valuable collaboration during installation and testing.

REFERENCES

- [1] G. Battistoni et al., Nucl. Instrum. Methods **152** (1978), 423.
- [2] G. Battistoni et al., Nucl. Instrum. Methods **164** (1979), 57.
- [3] G. Battistoni et al., Nucl. Instrum. Methods **176** (1980), 297.
- [4] G. Battistoni et al., Plastic tubes calorimeter *in* Proc. 2nd Int. Conf. on Instrumentation for Colliding Beam Physics, Stanford, 1982 (SLAC, Stanford, 1982).
- [5] E. Iarocci, Nucl. Instrum. Methods **217** (1983), 30.
- [6] G. Bauer et al., Upgraded muon detection system for UA1 based on a limited streamer tubes, preprint CERN-EP/86-161, submitted to Nuclear Instrum. Methods (1986).
- [7] H. Von Der Schmitt, RT FORTRAN compiler manual.

Figure captions

- Fig. 1** Schematic view of general system organization.
- Fig. 2** Setup and communication buses of the VME crates.
- Fig. 3** Schematics of the front-end amplifier.
- Fig. 4** Block diagram of the STAR module.
- Fig. 5** Block diagram of the gain controlled ADC.
- Fig. 6** ADC response curves, ranges, and resolution
- Fig. 7** Block diagram of the STAR Controller.
- Fig. 8** State diagram of the Hardware Sequencer.
- Fig. 9** Memory map of the STAR Controller registers.
- Fig. 10** General organization of the control software.
- Fig. 11** Maximum pulse height in a cluster.
- Fig. 12** Plot of residuals for beam particles at normal incidence.
- Fig. 13** STAR module.
- Fig. 14** STAR modules connected to the chamber strips.

Fig. 1

Fig. 2

Fig. 3

Fig. 4

Fig. 5

Fig. 6

Fig. 7

STATUS	BUFF1	BUFF2
A	0	0
B	1	0
C	1	1
D	0	1
E	1	1

0 = Buffer Empty

1 = Buffer Full

EOR = End Of Read

EOC = End Of Conversion

Fig. 8

SC MEMORY MAP

0000	CHANNEL 0	CHANNEL 1	} Pedestal Memory
	CHANNEL 2	CHANNEL 3	
	⋮	⋮	
OFFE	CHANNEL 4094	CHANNEL 4095	
2000	STATUS	CONTROL	
3000	THRESHOLD	CHARGE	
4000	DATA REGISTER		
8000	CLR. B2 FLAG	CLR. B1 FLAG	
9000	CLR. SEQ.	CLRB	
A000		GENER. STOR	
B000	CLR. CONV. FL.	INCR. SEQ.	

Fig. 9

Commands for data acquisition are broadcast to all the ACQ CPUs by the SUP CPU.
 Commands for monitor/calibration are sent to the appropriate CPU(s) by the MON CPU.

Fig. 10

Fig. 11

Fig. 12

Fig. 13

Fig. 14