The KLOE DAQ system and DAQ issues for experiments in the year 2000

M.L. Ferrer and The KLOE Collaboration[a]

INFN, Laboratori Nazionali di Frascati, Via Enrico Fermi 40, 00044 Frascati, Italy

An overview of the KLOE data acquisition system is presented, covering trigger, readout, event builder system and software. The technical choices in developing the DAQ architecture and results of the tests already done are discussed. Finally some issues for the future are given in the conclusions.

1 Physics and requirements

The KLOE experiment[1,2] is scheduled to run in the fall of 1997, at the DAΦNE Φ-factory in Frascati. Its aim is the study of direct CP violation looking at the neutral kaon system produced in the Φ decay, in particular the final states involving 4 pions from $K_S K_L$ decays, in all possible charge combination.

As in all e^+e^- collider experiments background levels at KLOE are low, allowing for efficient selection of high purity $K_S - K_L$ event samples, but the aim is to reach an accuracy in the measurement of $\Re(\epsilon'/\epsilon)$ of $\sim \mathcal{O}(10^{-4})$. Such accuracies impose stringent requirements on the detector performance, the DAQ and the trigger systems, that can be achieved by insuring that the inefficiencies in triggering on different event configurations be kept at the level of few$\times 10^{-3}$. At the same time a redundant trigger must allow the measurement of these efficiencies to the required accuracy of 0.3×10^{-3}. Moreover, the DAQ must ensure that the trigger treats democratically the several channels involved in the measurement of the parameters describing CP violation.

2 The detector

The detector has two main components: an electromagnetic calorimeter (E.M.) made of lead and scintillating fibers, particularly suited to provide a fast and efficient trigger, and a tracking chamber (D.C.) for charged particles reconstruction that uses aluminum field wires, helium + hydrocarbons for the chamber gas and cell sizes of $2 \times 2\ cm^2$ and $3 \times 3\ cm^2$ with a maximum drift time of 2μs. The KLOE detector contains 5,000 photomultipliers (PMs) for the readout of the calorimeter and $\sim$13,000 wires for the tracking chamber. The PM signals are amplified in the PMs' bases and sent to analog boards of 30 channels each. The outputs of this board are connected to ADCs, TDCs and to the trigger processor. The chamber signals are preamplified at the wire feed-through and sent to boards of 48 channels each. The output of two of these boards are connected to a 96 channel TDC board for drift time measurements.

[a]The members of the KLOE Collaboration are listed in Ref. 2.

The FEE is now under test. Both major detector components are under construction, and completion is expected by the end of 1996. A prototype of the final data acquisition (DAQ) system will be used and tested with real cosmic rays data from these detector components and with MonteCarlo data, until the apparatus is finally installed in the accelerator ring.

3 The Trigger

In DAΦNE the time between bunch crossings is $\sim$ 2.7 ns. Since it is not possible to generate a trigger in this time, the KLOE trigger must operate in a continuous mode. At the DAΦNE target luminosity of $10^{33} cm^{-2} s^{-1}$ the ϕ production rate is $\sim$5 kHz. The main backgrounds are due to electromagnetic processes, Bhabha and radiactive Bhabha scattering from collisions of the two beams, as well as Coulomb scattering and bremsstrahlung in the residual vacuum chamber gas, and finally Touschek scattering in the single bunches. Cosmic rays are another important class of background events.

A two level scheme of trigger [3] has been adopted in order to produce an early trigger with good timing information to start the front end electronic (FEE) operation (E.M. TDC's), while making best use of the D.C. information.

The first level trigger is produced within 150 ns using local energy deposition in the E.M. sectors and early multiplicity information from the D.C. It can be vetoed if the event was identified as a Bhabha or a cosmic ray event.

The second level trigger uses additional information collected from the D.C. within $\sim$ 850 ns together with the E.M. information to confirm the former and start the DAQ system. It originates local trigger numbering on each ADC and TDC boards and copy its contents in the local buffers together with the related trigger number. Event integrity is assured by generating early busy signal in the FEE boards when only space for a new event is available. Readout controllers memorize the new trigger number in a local FIFO. Synch. cycles (4 μs) are periodically generated to check trigger number consistency.

The trigger system has been tested using GEANFI, a modified version of GEANT including new or better treatment of the KLOE physics. Trigger efficiencies and expected rates for physics and background events are presented in Table 1. A first level trigger rate of about 25 kHz, with a fixed deadtime of 2 μs imposed

Table 1: Trigger efficiencies and expected rates

Events	L1 eff. (%)	L2 eff. (%)	Events	L1 (kHz)	L2 (kHz)
$\pi^{\pm,\pm}$	99.8	99.6	Total ϕ	5	5
$\pi^{\pm,00}$	99.9	99.4	Bhabha	10	3
$\pi^{00,00}$	99.9	99.8	Machine	9	5
$\pi^{\pm,\pi\ell}$	99.9	99.7	Cosmics	1	1
			Total	25	12

by the FEE, will originate a loss of luminosity of ∼5%. The expected physics event size is 5 kByte and the required sustained throughput is ∼ 50 MByte/s.

4 The Readout system

The KLOE FEE and the Readout system[4] use VME hardware, 6U or 9U cards. Two levels of data concentration are implemented. At the first level, data in every FEE crate (L1-VME) are concentrated in a Readout Controller for KLOE (ROCK) using a custom bus implemented on the user lines of the VME/J2 connector. A sparse data scan cycle reveals for each trigger number the FEE cards having data to be collected. In a second step DMA transfers are started to read the contents of each module until an EOB line marks the end of the transfer.

The 2nd level of data concentration (L2-VME) is also implemented without the intervention of dedicated CPUs. A ROCK Manager (ROCKM) interconnects up to 8 ROCK's in a chain using a custom C-bus working at a data transfer rate of up to 10MHz over 10 m total cable length. By using a token passing mechanism through the C-bus, data related to a given trigger number are concentrated in the ROCKM local FIFO.

ROCK and ROCKM have local buffering of 16 and 32 kBytes respectively, work asynchronously to the trigger signal, produce a first derandomization of data and implement data integrity check with minimum software overhead and complexity.

For initial settings and monitoring, the crates in a chain are interconnected via VIC-bus, while the L2-VME crates are interconnected via VIC-bus for Data Flow control.

At 10 kHz trigger rate a chain can handle, according to the simulations, up to 15 MB/s bandwidth with a maximum occupancy at ROCK FIFO of 6 kByte. The KLOE readout is configured to sustain a 5 MB/s throughput per chain. So the required throughput is less than $\frac{1}{2}$ of the FDDI nominal bandwidth which is used to interconnect the 10 chains of the readout system to the event builder switch[5].

5 Data flow and Run controls

Sub-event data of 0.5 kB size in average are memorized in the ROCKMs for each trigger. Empty sub-events are also memorized in a two words record. CPUs in the L2-VME crates collect the sub-events in a local circular buffer, pack ∼ 100 of them (∼ 50 kB) and send every packet using TCP/IP to a defined processor in a farm connected to a switch port. The Data Flow Control (DFC) is in charge of the destination assignment for every packet of events, assuring event integrity, optimizing traffic in the switch (consecutive packets sent to processors connected to different switch ports), and preventing the reduction of the TCP/IP performance when the buffer at the receiver side become smaller and a new shorter window size need to be negotiated.

The DFC implementation is made by automatically creating a network map that contains a list of processors receiving data and the switch port where they are seen, by producing an ordered list of processors in an static table and by creating dynamic tables containing ordered list of processor flags according to the queue status. Each table has a validity defined by a reasonable future trigger number calculated by the DFC when a *'QueueAlmostFull'* message is received. Shared memory through VIC-bus is used to distribute tables to VME-CPUs. At 10 kHz trigger rate, 100 sub-events packet, and 10 farm chains a new event is assigned per port every 0.1 sec. This time is large enough to be managed in software.

The Simple Network Management Protocol (SNMP), widely used to manage network devices that are defined in standard Management Information Base (MIB), is used in KLOE to implement DFC and Run Control facilities[6]. Public domain software produced by the Carnegie Mellon University allows us to implement dedicated SNMP agents. Private MIBs are defined to distribute commands between processors, by setting MIB variables and to obtain acknowledgements by getting MIB variables.

6 The Event Builder

The KLOE event builder system has been studied having in mind the following requirements: a high rate, parallel and scalable system using physical channels that are standards in the telecommunications environment and also standard protocols for data transfer and network management. The FDDI GIGAswitch bridge from DEC, running in our laboratory since two years, was chosen to implement the interconnections of the readout system with a farm of very powerful processors where events are built, monitored, formatted and written on DLT devices. TCP/IP and SNMP were chosen as data transfer and control and management protocols.

New high speed network channels like ATM or FCS are still waiting for complete standardization so FDDI is nowadays the only widely supported and implemented standard where every software application can be tested and optimized now. Moreover the migration to ATM is possible and relatively easy. Table 2 presents the maximum performances obtained using TCP and UDP on ALPHA stations over FDDI and a Digital ATM switch. The Digital proprietary credits mechanism that eliminates cell retransmission can be used. The required CPU cost, that are similar for TCP and UDP protocols, are also given in percent and in number of SPECint92. DEC 3000-600 (114 SPECint92) stations were used on both sides.

According to Table 2 an estimation of the ALPHA station CPU cost for a TCP protocol running over 622 Mb/s ATM (70 MB/s of maximum data throughput)

Table 2: TCP and UDP over FDDI and ATM

	FDDI	ATM
Channel speed (Mb/s)	100	155
Phys. throughput (MB/s)	12.5 MB/s	17.5 MB/s
TCP (MB/s)	11.5	16.5
UDP (MB/s)	12.5	16.5
CPU %	50-60	70-80
SPECint92	$\sim$60	$\sim$85

is ~350 SPECint92. However optimization of the TCP/IP protocol is possible by increasing the window size beyond 64 kB. This was already proposed as standard [7] and is used in OSF/1 [8] and will also be implemented in HP-RT 2.1. HP-UX implements a 'single copy' TCP/IP with specialized hardware to improve performance[9]. By increasing both the memory bandwidth and the fraction of protocol implemented in the interface, the performances can be improved reducing at the same time the cost of the required CPU.

Independently of the definition of a standard reliable UDP, the TCP protocol is a good choice for data transfer and of general interest independently of the physical channel.

TCP/IP based event builder require multiple connections active in any processor. Two different methods to maintain multiple connections are used[10], both accessing TCP/IP through the Berkeley socket interface,

- SELECT Call: A single process does the IO-multiplexing by looking through looping over open connections and using the routine *select* which blocks until there is something to read or write in at least one of the active sockets. A timeout is available for error handling. In the KLOE event builder, incomplete events in the receiving farm will originate this kind of timeout, that can be propagated in the L2-VME crate to a *Trigger busy* signal asserted by the ROCKM.

- POSIX-Threads: Either one thread is used for each connection maintained independently or a defined number of threads are used to handle a bigger number of active connections using the first free active socket to execute the related read or write operation. A binary semaphore is used in accessing the socket list.

Tests with up to 60 open connections were made and the same performances were obtained using both methods. The total throughput achieved is independent of number of sockets/connections. The maximum throughput is also obtainable with small window size of 8 kByte.

The KLOE event builder software is being developed by writing UNIX portable code that run in a multi-platform set of computers mounting common NFS disks. So mechanisms are implemented to maintain multi-platform code in only ONE set of source files and platform (or flavor) specific directories.

No advantages are obtained by using real-time operating systems in KLOE, where real-time is implemented in hardware in both ROCKS and ROCKMs. Moreover, the porting of applications is more difficult and the communication drivers, obtainable from different vendors, are frequently more difficult to maintain.

7 CPU and FDDI interfaces on VME

CPU's from HP, DEC, and CETIA have been tested in the KLOE laboratories using different solutions for FDDI interfaces that are based on separated VME boards or use the PCI connector. Tests on CES boards, with expected good VME performances, will follow. Results about TCP, UDP and VME performances are shown in Table 3.

Table 3: Network interfaces and VME CPUs

	HP-743rt 100MHz	DEC-M224 224MHz	CETIA-Power 50MHz	CES-PC603 and PC604
SPECint92	100	80	50	63/155
SPECfp92	137	100	60	58/160
O.S.	HP-RT/ HP-UX	VxWorks/ OSF1 rt	LynxOS	LynxOS
FDDI	VME- Rockwell	PCI- DEC	VME- Interphase	PCI- Interphase
TCP (MB/s)	7 / 7.8	- / 7.5	6	-
UDP (MB/s)	8.1 / 11.4	- / 10	-	-
VME (MB/s)	15/12 D32 30/24 D64	16/12 D32 27/16 D64	-	[40]

These measurements show the improvement of performance obtained by using UNIX instead of real time operating systems in both HP and Digital platforms.

8 Farm

Three different configurations have been consided in the KLOE experiment and are still under study.

- VME or Futurebus+ crates with Single Board Computer (SBC) connected to the FDDI network, receiving events and using the backplane to transfer data over a dedicated IO-SBC,
- VME or Futurebus+ crates with only one SBC connected to the FDDI network. Two passes in the backplane are needed in this case: to distribute events and to collect data to be written on tape,
- Multi-CPU servers implementing both network connection and tape output using the high performance internal bus for data transfer.

Table 4: Measured and estimated CPU power needed in the KLOE online farm

Operation	SPECint92	Operation	SPECint92
Receive events	450	Record events(DLT/RBIO)	~300
Receive+order	500	Reformat events	~300
Receive+order+swap	750	Compress events	500 - 1000
		Monitor events	500 - 1000

The evaluation of the CPUs needed in KLOE to implement the minimum functions required in a farm for 10 kHz of events is estimated to be around 4000

SPECint92 (see Table 4). Taking into account the fact that the VME-CPU performances accessing the VME backplane are still low, and that the cost of the servers with the same aggregated CPU power are just 20-30% higher, but offering a much most general and comfortable development system and better overall performance, the first implementation of the online farm software in KLOE will be developed on servers. Events distribution will be implemented using circular buffer managers that could run through VME or Futurebus+ backplanes as well.

9 Conclusions and some issue for the future

A multi-vendor event builder system is feasible and is required in experiments running many years and having different requirements for different pieces of the system. Moreover it allows us to obtain lower price, better performance and easy migration. There are still problems in using multi-vendors VME-CPU boards, because standards for software access to VME are missing. Effort is needed in this direction.

For experiments implementing in hardware the real-time functionalities, diskless UNIX systems are a good solution and no benefit is obtained using real-time operating systems. Having only UNIX systems to be configured and maintained is a desirable goal.

Looking at an Event Builder system as a software telecommunication problem, TCP/IP for data transfer is still the best choice, even if the CPU cost is high, but improvement of the performances can be expected in the future. SNMP can be extended for DAQ control and management.

No good general purpose solutions are available now for making farms based on VME crates. Boards with reduced functionalities (no SCSI, no graphic) and better VME performances could be attractive in the near future for special tasks like event reconstruction.

1. The KLOE collaboration, *KLOE, a general purpose detector for DAΦNE*, **LNF**-92/019 (1992); *The KLOE detector, Technical Proposal*, **LNF**-93/002 (1993).
2. The KLOE collaboration, *The KLOE Data Acquisition System*, **LNF**-95/014 (1995).
3. The KLOE collaboration, *The KLOE Trigger System*, **LNF**-95 (1995).
4. A.Aloisio et al., *Level 1 DAQ System for the KLOE experiment*, CHEP95.
5. H. Beker et al., *Interprocess and interprocessor Data Flow in the KLOE Data Acquisition System*, CHEP95.
6. E. Pasqualucci et al., *The use of SNMP implementing Run and Data Flow controls in the KLOE experiment*, CHEP95.
7. RFC 1323, *TCP Extensions for High Performance.*
8. C.Chang et al., *High performance TCP/IP and UDP/IP Networking in DEC OSF/1 for Alpha AXP*, Digital Technical Journal 1(5), 44 (1993).
9. C.Dalton et al., *Afterburner: A network-independent card provides architectural support for high-performance protocols*, IEEE Network 7(4), 36 (1993).
10. W.Grandegger, H.Beker, M.L. Ferrer, *TCP performance test with I/O-multiplexing for the KLOE event builder*, KLOE-note 120/94, (1994).